Improving RGB-Thermal Semantic Scene Understanding With Synthetic Data Augmentation for Autonomous Driving

Haotian Li, *Graduate Student Member, IEEE*, Henry K. Chu, and Yuxiang Sun, *Member, IEEE*

***Abstract*—Semantic scene understanding is an important capability for autonomous vehicles. Despite recent advances in RGB-Thermal (RGB-T) semantic segmentation, existing methods often rely on parameter-heavy models, which are particularly constrained by the lack of precisely-labeled training data. To alleviate this limitation, we propose a data-driven method, SyntheticSeg, to enhance RGB-T semantic segmentation. Specifically, we utilize generative models to generate synthetic RGB-T images from the semantic layouts in real datasets and construct a large-scale, high-fidelity synthetic dataset to provide the segmentation models with sufficient training data. We also introduce a novel metric that measures both the scarcity and segmentation difficulty of semantic layouts, guiding sampling from the synthetic dataset to alleviate class imbalance and improve the overall segmentation performance. Experimental results on a public dataset demonstrate our superior performance over the state of the arts.**

***Index Terms*—Semantic scene understanding, RGB-T fusion, autonomous driving, synthetic image generation.**

I. Introduction

RGB-T semantic segmentation [1] enhances scene understanding for autonomous vehicles by combining RGB-T images to improve performance under challenging illumination conditions [2], leveraging both convolutional neural network (CNN) [3], [4] and transformer [5], [6]. It provides essential information for downstream tasks, such as vehicle localization [7], [8] and autonomous navigation [9]. In supervised RGB-T semantic segmentation, the hand-labeling process is laborious and costly, resulting in limited datasets. For example, the largest public dataset, MFNet dataset [2], contains only 1,568 pairs of RGB-T images (including 784 flipped pairs) for training. Existing methods [3], [4], [5], [6] focus on designing more advanced models, but it is challenging to enhance segmentation performance by upgrading the model when the training set is limited. This motivates us to explore whether the segmentation performance could be further improved by generating synthetic image pairs that mimic real scenes.

Received 26 December 2024; accepted 5 February 2025. Date of publication 5 March 2025; date of current version 26 March 2025. This article was recommended for publication by Associate Editor M. Ramezani and Editor C. Cadena upon evaluation of the reviewers' comments. This work was supported in part by Hong Kong Innovation and Technology Fund under Grant ITS/145/21 and in part by City University of Hong Kong under Grant 9610675. *(Corresponding author: Yuxiang Sun.)*

Haotian Li and Henry K. Chu are with the Department of Mechanical Engineering, The Hong Kong Polytechnic University, Kowloon, Hong Kong (e-mail: haotian.li@connect.polyu.hk; henry.chu@polyu.edu.hk).

Yuxiang Sun is with the Department of Mechanical Engineering, City University of Hong Kong, Kowloon, Hong Kong (e-mail: yx.sun@cityu.edu.hk).

Digital Object Identifier 10.1109/LRA.2025.3548399

To validate this idea in supervised RGB-T semantic segmentation, we need to generate high-fidelity synthetic RGB-T images from semantic layouts, which are pixel-level maps labeling each pixel into different classes. Recent techniques [10], [11] introduce new possibilities for image generation. Several layout-to-image generative models [12], [13] can generate synthetic RGB images from the semantic layouts. However, these methods are designed for RGB images and cannot be directly applied to RGB-T images. To generate high-fidelity synthetic RGB-T images, we adapt the generative model FreestyleNet [13] to the widely-used MFNet dataset [2]. Based on this method, we can generate diverse synthetic images to build a large-scale synthetic dataset (see Fig. 1).

The generation of a large-scale synthetic dataset offers a potential solution to the class imbalance problem, which is a significant factor affecting the accuracy of RGB-T semantic segmentation. Analysis of experimental results from existing studies reveals notable disparities in segmentation accuracy across different classes. In Fig. 2, we show the pixel ratio of each class (class pixel ratio) in the MFNet training set and the Intersection-over-Union (IoU) for each class, as evaluated using two state-of-the-art methods [5], [6]. The results show a clear correlation between the class pixel ratio and its segmentation performance: for instance, the `Car` class, with the highest pixel ratio, consistently achieves the highest IoU, whereas the `Guardrail` class, with the lowest pixel ratio, consistently records the lowest IoU. However, an anomaly is observed where the `Car Stop` class, despite having a higher pixel ratio, yields a lower IoU compared to the `Bump` class. This suggests that segmentation performance is influenced not only by the class pixel ratio but also by the difficulty of segmenting each class. Therefore, the class imbalance problem involves not only disparities in the number of pixels for each class, but also differences in the segmentation difficulty of each class.

In previous studies, the class imbalance problem in RGB-T semantic segmentation has not been discussed. To balance class distribution, object detection tasks typically adjust the number of samples for each class through resampling [14]. However, increasing the number of samples for uncommon classes may lead to overfitting, while decreasing the number of samples for common classes can result in the loss of important information. In addition, Mixup series methods [15], [16], [17] generate new training data by interpolating or mixing samples, but these can produce unrealistic sample combinations, potentially leading the model to learn incorrect information. Although some loss functions [18], [19] have been proposed for addressing class imbalance in semantic segmentation tasks, they often introduce hyperparameters that can complicate model tuning.

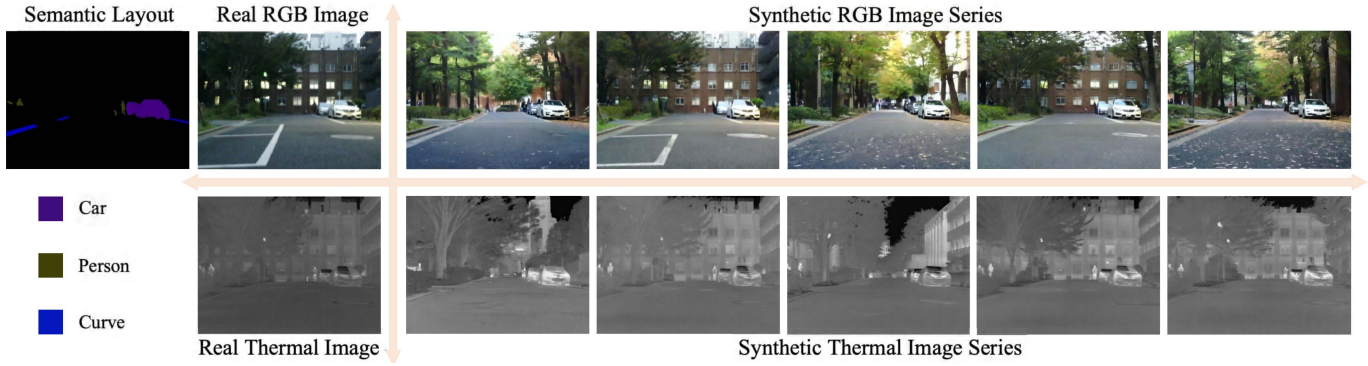

Fig. 1. The synthetic RGB-T dataset. Real RGB images, thermal images, and their corresponding semantic layouts are sampled from the MFNet dataset.

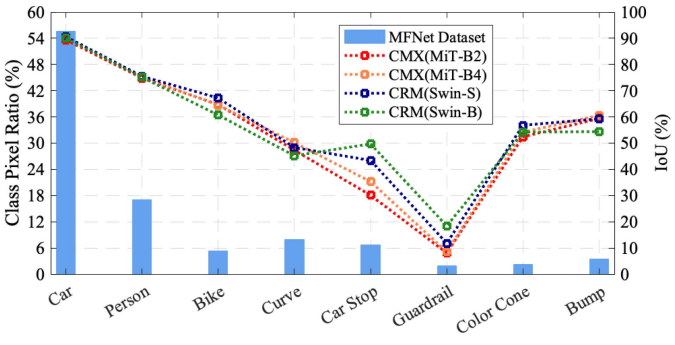

Fig. 2. The pixel ratio of each class in the training set of MFNet dataset [2] and the IoU for each class of CMX [5] and CRM [6]. We choose CMX using MiT-B2 and MiT-B4 as backbone, and CRM using Swin-S and Swin-B as backbone. This figure shows the correlation between the class pixel ratio in the training set and the IoU for that class.

The large-scale synthetic datasets generated in this study provide new ways to alleviate the class imbalance problem. To improve segmentation accuracy for rare and difficult-to-segment classes, we propose a novel metric that measures both the scarcity and segmentation difficulty of each semantic layout in the real dataset. This metric guides the sampling from the pre-generated synthetic dataset, ensuring that classes with fewer or more challenging examples are adequately sampled. The main contributions of this work are summarized as follows:

1) We propose a data-driven method, SyntheticSeg, to enhance RGB-T semantic segmentation by generating a large-scale, high-fidelity synthetic dataset. Both the code and the dataset are open-sourced.[1]
2) We design a novel metric to measure the scarcity and segmentation difficulty of each semantic layout, optimizing sample selection from the synthetic dataset to better alleviate class imbalance.
3) Our method achieves state-of-the-art performance on the MFNet dataset [2], demonstrating the effectiveness of our synthetic dataset and sampling strategy.

This paper is structured as follows. Section II reviews the related work. Section III describes our proposed method. Section IV discusses the experimental results. Conclusions and future work are drawn in the last section.

[1]Our code and dataset: https://github.com/lab-sun/SyntheticSeg

II. Related Work

A. RGB-T Segmentation Methods

Some methods [20], [21], [22] focus on designing novel multimodal feature fusion modules to enhance the fusion of RGB-T features. Li et al. [21] proposed IGFNet, which utilizes a weight mask from an Illumination Estimation Module (IEM) to selectively integrate RGB-T features. Huang et al. [23] proposed RoadFormer+ to extracts heterogeneous features from various modalities and merge the features across different scales and receptive fields. In addition, Li et al. [24] proposed temporal-consistent framework to improve the segmentation accuracy and consistency. Other methods [5], [6] improve RGB-T segmentation by using larger backbones in feature extraction modules. Shin et al. [6] introduced a complementary random masking strategy that boosts segmentation accuracy and robustness by reducing over-reliance on a single modality. To achieve better results, there is a trend towards designing models with more parameters and using larger backbones. These model-driven methods make them prone to overfitting, especially in tasks like RGB-T semantic segmentation with limited training data.

B. Generative Models

The introduction of Diffusion series networks [10], [11] has enabled the generation of high-quality images through a diffusion process that gradually transitions from noises to clear images. Denoising Diffusion Probabilistic Models (DDPMs) [10] establish a novel link between denoising score matching and Langevin dynamics. Dhariwal et al. [11] demonstrated that diffusion models can surpass Generative Adversarial Networks (GANs) in image synthesis. Building on this foundation, a series of works [12], [13] propose layout-to-image generative models to generate RGB images based on semantic layouts. Wang et al. [12] proposed a framework that processes semantic layouts and noisy images differently, enhancing the quality and diversity of generated images. Xue et al. [13] explored image generation with unseen semantics using pre-trained text-to-image diffusion models.

C. Class Imbalance Problem

To address the class imbalance in object detection, Saez et al. [15] enhanced SMOTE with an ensemble-based noise filter, addressing noisy and borderline examples in imbalanced

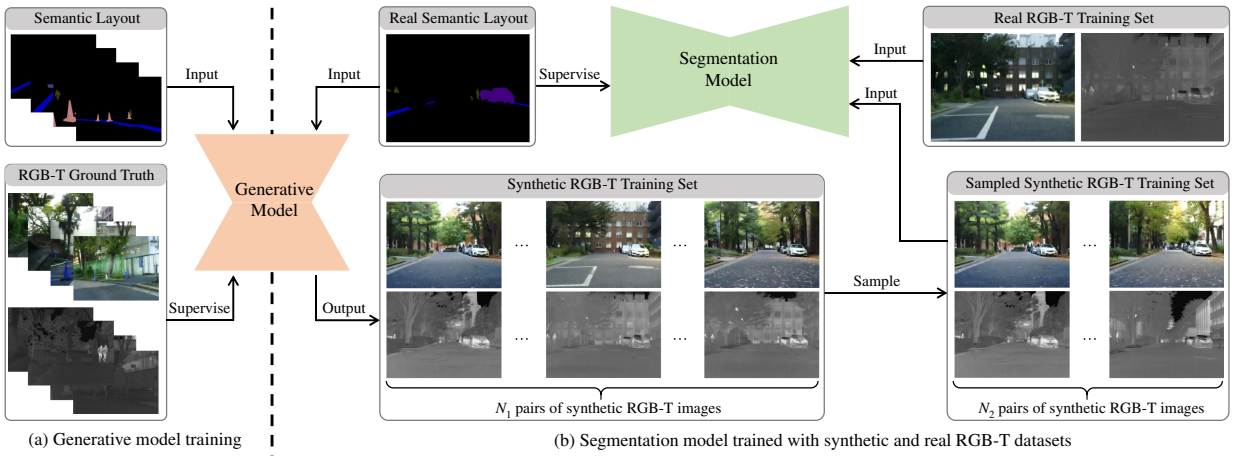

Fig. 3. The overall framework of our SyntheticSeg. (a) illustrates the training process of RGB-T generative model; (b) illustrates the pipeline for joint training of segmentation models using both synthetic and real RGB-T datasets.

datasets. Choi et al. [16] introduced a token-level data augmentation method for transformers that is efficient and guided by attention. Venkataramanan et al. [17] presented a data augmentation technique that interpolates aligned features, combining the geometry of one image with the texture of another to enhance representation learning. In addition, specific loss functions [25], [26] shift focus away from simple samples and emphasize the classification of more challenging ones. Some of them [18], [19] have been proposed to address the class imbalance in semantic segmentation. Tian et al. [18] introduced Recall Loss, a new loss function to balance precision and recall in semantic segmentation. Qiu et al. [19] proposed Subclassified Loss, which addresses class imbalance by focusing on subclasses.

D. Differences From Existing Methods

Our method differs from the above methods in two aspects: 1) Corresponding to the model-driven method, we propose to improve RGB-T segmentation through a data-driven method by creating a large-scale synthetic RGB-T dataset to expand the training data; 2) We use the high-fidelity synthetic data to alleviate the class imbalance problem in RGB-T semantic segmentation.

III. The Proposed Method

A. The Overall Framework

The MFNet dataset [2] contains only 2,353 fully annotated pairs (including 784 flipped pairs) of RGB-T images, with just 1,568 pairs in the training set. As networks used for RGB-T semantic segmentation grow more powerful, the limited scale of the dataset has become a critical factor affecting segmentation performance. With the advent of generative models like Diffusion [11], it is now feasible to expand the training data by creating high-fidelity synthetic images.

We adapt the layout-to-image FreestyleNet [13] as our generative model, and the framework of our SyntheticSeg is illustrated in Fig. 3. FreestyleNet is trained and validated on COCO-Stuff [27] and ADE20K [28] to generate RGB images from semantic layouts. To apply it to the RGB-T image generation task on the MFNet dataset [2], we separately feed the RGB and thermal images from the MFNet dataset into FreestyleNet. The thermal images are first normalized and then converted into 3-channel images to match the input dimensions expected by FreestyleNet. Then, we modify the class indices of the dataset to define the number of output classes of the generative model.

As illustrated in Fig. 3(b), we can feed the semantic layout from the real training set into the trained RGB-T generative model to obtain the corresponding RGB-T images. The semantic layout fed into the model and the generated RGB-T images form a pair of synthetic data suitable for training. In principle, for one semantic layout, we can generate an infinite number of corresponding RGB-T images by varying the random seed. So, if the real dataset contains N_0 pairs of training data, we can generate $N_0 \times N_1$ pairs of synthetic data, where N_1 is the number of selected seeds. This method can significantly expand the scale of fully annotated data available for training. To alleviate the class imbalance problem, we propose a metric to assess the scarcity and segmentation challenge of each semantic layout in the real dataset. Based on this metric, we determine N_2, the number of synthetic data to be sampled for each real semantic layout. In this way, we have one pair of real RGB-T image and N_2 pairs of synthetic RGB-T images for each semantic layout.

Then, we feed RGB-T images from both the real and sampled synthetic datasets into the existing segmentation model, perform feature extraction and fusion on the RGB and thermal images, and decode them to obtain the segmentation results.

B. Sampling Mechanism

As shown in Fig. 2, the segmentation performance of different classes in RGB-T semantic segmentation is related to their respective pixel ratio and segmentation difficulty. To improve the segmentation performance of classes with low pixel ratio and high segmentation difficulty, we design a metric to evaluate the scarcity and segmentation difficulty of each semantic layout in the training set. This metric determines the number of corresponding synthetic RGB-T images sampled from the generated synthetic dataset for that semantic layout. This sampling mechanism can improve the number and diversity of uncommon and challenging samples.

To evaluate the scarcity and segmentation challenge of each semantic layout, we first assess the pixel ratio and segmentation difficulty of the different classes. In the MFNet dataset, for each

class c ($c \in \{1, 2, \ldots, C\}$, where C is the total number of classes excluding the background class 0), the pixel ratio of the class P_c can be defined as:

$$P_c = \frac{\sum_{i=1}^{M} |\Omega_c|}{\sum_{c=1}^{C} \sum_{i=1}^{M} |\Omega_c|}, \tag{1}$$

where M is the total number of images in the training set and Ω_c is the set of pixels belonging to class c in the layout. So, the numerator represents the sum of pixels for class c across all images, while the denominator represents the sum of pixels for all classes except the background across all images.

Then, we define μ_c as the class-wise mean loss for class c, representing the corresponding segmentation challenge. The μ_c can be defined as:

$$\mu_c = \frac{\sum_{i=1}^{M} \sum_{j \in \Omega_c} L_{ij}}{\sum_{i=1}^{M} |\Omega_c| + \epsilon}, \tag{2}$$

where L_{ij} is the loss for pixel j in image i, calculated by a pre-trained RGB-T semantic segmentation model. ϵ is a small constant to ensure numerical stability. So, the numerator represents the sum of losses for class c across all images, while the denominator represents the sum of pixels for class c across all images.

For each semantic layout i ($i \in \{1, 2, \ldots, M\}$, where M is the total number of layouts in the training set), we can calculate the score for scarcity and segmentation challenge of the semantic layout based on the above obtained P_c and μ_c. The scarcer the semantic layout and the more difficult the segmentation, the higher the score. We can obtain the score β_i for layout i by:

$$\beta_i = \frac{\sum_{c=1}^{C} \sum_{j \in \Omega_c} \mu_c^{ij}}{\sum_{c=1}^{C} |\Omega_c| + \epsilon} \frac{1}{\min_{c \in C_i} P_c}, \tag{3}$$

where μ_c^{ij} is the class-wise mean loss for class c for pixel j in image i. C is the total number of classes excluding the background class 0. C_i denotes the set of classes present in semantic layout i, meaning the number of classes in C_i is less than or equal to C. So, we can use this formula as a metric to measure the scarcity and segmentation challenge of the semantic layout. This metric helps determine the number of synthetic images to sample from the high-fidelity synthetic dataset. By sorting all semantic layouts in the training set from small to large using this metric, the number of samples S_i corresponding to semantic layout i can be defined as:

$$S_i = \min\left(1 + \left\lfloor \frac{(i+1)}{M} \times R_{\max} \right\rceil, R_{\max}\right), \tag{4}$$

where $\lfloor \cdot \rceil$ indicates rounding to the nearest integer, M is the total number of layouts in the training set, and $R_{\max}$ is the maximum sampling number for a single semantic layout in the synthetic dataset. By leveraging this mechanism, we sample more synthetic images for semantic layouts that are both uncommon in the training set and present high segmentation challenges.

C. Data Distribution of Synthetic Dataset

Based on the sampling mechanism described above, we can create different scales of sampled synthetic datasets by setting different maximum sampling numbers $R_{\max}$ in Equ. (4) for the original large-scale synthetic dataset. We set $R_{\max}$ to values of 1, 2, 3, 4, and 5, which resulted in 1,568, 2,744, 3,660, 4,509, and 5,333 sampled synthetic images, respectively. When $R_{\max} = 1$, the pixel ratio of each class in the sampled synthetic dataset is the same as that in the real dataset. In principle, increasing $R_{\max}$ can raise the pixel ratios in the uncommon and difficult-to-segment classes. To more clearly illustrate the variation in pixel ratios as $R_{\max}$ increases, we normalize these ratios across different $R_{\max}$. The normalized ratios are presented in Fig. 4. Take the `Guardrail` class as an example: prior to normalization, the pixel ratios for $R_{\max}$ values of 1, 2, 3, 4, and 5 are 1.86%, 2.22%, 2.29%, 2.36%, and 2.39%, respectively. These ratios show a slight upward trend, but the change is subtle when viewed as absolute values. However, after normalization, the pixel ratios for $R_{\max}$ values of 1, 2, 3, 4, and 5 become 0%, 67.92%, 81.13%, 94.34%, and 100%, respectively. This normalization makes the incremental changes clearer and highlights the trend. Fig. 4 shows that as $R_{\max}$ increases, the pixel ratio of the car class, which is the most common and easiest to segment, decreases. Meanwhile, the pixel ratios of other classes for $R_{\max} > 1$ are higher than those for $R_{\max} = 1$ (i.e., the pixel ratios in the real dataset).

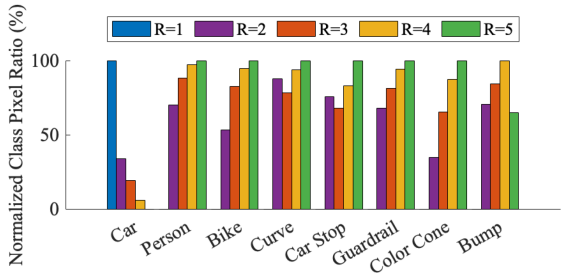

Fig. 4. The normalized pixel ratio of each class in the synthetic dataset with different $R_{\max}$.

IV. Experimental Results and Discussions

A. Implementation Details

We use the MFNet dataset [2] as the real dataset for our experiments. The dataset consists of 2,353 pairs (including 784 flipped pairs) of RGB-T images and it includes 9 classes: `Background`, `Car`, `Person`, `Bike`, `Curve`, `Car Stop`, `Guardrail`, `Color Cone`, and `Bump`. We follow the same split scheme as that in [2]: 1,568 pairs for training, 392 for validation, and 393 for testing. To construct a synthetic dataset, we feed 1,568 semantic layouts from the real training set into the trained layout-to-image generation model. We set the class indices of FreestyleNet [13] to 9 to adjust the number of output classes, aligning it with the MFNet dataset. Using an NVIDIA RTX 3090 graphics card, the generation speed for synthetic RGB and thermal images is 7.1 seconds per image. By setting 20 different random seeds, we obtain a large-scale synthetic dataset with 31,360 pairs of RGB-T synthetic images and corresponding semantic layouts. We set $R_{\max} = 2$ in Equ. (4) to create the sampled synthetic training set, which includes 2,744 pairs of synthetic RGB-T images.

We train the existing RGB-T semantic segmentation model using both the sampled synthetic training set and the original real training set. Since the scale of the sampled synthetic training set is larger than that of the original real training set, and the quality of the synthetic images is lower than that of the real images, we oversample the real training set to match the scale

TABLE I
THE RESULTS (%) OF THE ABLATION STUDY ON THE TRAINING DATA

Training Data		mAcc	mIoU
Real	Synthetic		
1×		67.3	58.2
	1×	65.6	52.8
1×	1×	**72.2**	**59.9**
2×		68.5	57.5

1× and 2× indicate using the real or synthetic training data once or twice for training, respectively. The best results are highlighted in bold.

of the sampled synthetic training set. Specifically, if two pairs of synthetic RGB-T images are sampled for a semantic layout during training, the corresponding real RGB-T images are also sampled twice to ensure same sizes of the real and synthetic training data. This prevents the low-quality synthetic images from dominating the training process.

B. Ablation Study

To balance performance and model complexity, we use CMX [5] with a MiT-B2 backbone as the RGB-T semantic segmentation model for ablation experiments.

1) Ablation on Training Data: To examine the effects of various training sets on the RGB-T semantic segmentation model, we first conduct an ablation study presented in Table I. The first row of Table I shows the segmentation results obtained by training with 1,568 pairs of real RGB-T images from the original MFNet dataset. The second row shows the results from training with 1,568 RGB-T images generated from 1,568 semantic layouts. Comparing the results, we observe that the mIoU of training with synthetic data is 5.4% lower than that of training with real data. This indicates that although the generated synthetic RGB-T images are visually similar to real images, there remains a quality gap between them and real images. The third row of Table I shows the segmentation results obtained by training with 1,568 pairs of real RGB-T images and the corresponding 1,568 pairs of synthetic RGB-T images. Training with a combination of real and synthetic images results in a higher mIoU compared to training with only real images. This indicates that despite synthetic images not being as high-quality as real images, joint training enhances the diversity of the training set. This allows the model to learn from a wider variety of samples, thereby improving its performance. To demonstrate that the mIoU improvement in the third row is not solely due to increased amount of training data, we conduct the fourth experiment using twice the original real training set, ensuring the same amount of training data as in the third experiment. The mIoU from the fourth experiment is decreased by 2.4% compared to the third and is even lower than the mIoU obtained with just the original real training set. This indicates that merely increasing the training data without enhancing its diversity does not effectively improve the segmentation performance.

2) Ablation on Sampling Mechanism: We set 20 different random seeds and generate a synthetic dataset 20 times the scale of the original dataset. Our goal is to enhance the segmentation performance of uncommon and challenging classes through the sampling mechanism, thereby improving overall performance. We design an ablation experiment, shown in Table II, to analyze the impact of the maximum number of samples $R_{\max}$ on the results. $R_{\max} = 1, 2, 3, 4, 5$ means that for each semantic layout in the original training set, a maximum of 1, 2, 3, 4, or 5 pairs of synthetic RGB-T images are sampled, respectively. The higher the $R_{\max}$, the larger the sampled synthetic training set. Table II indicates that the overall segmentation performance of the model improves when $R_{\max} > 1$ compared to $R_{\max} = 1$. The accuracy (Acc) and IoU for the `Car Stop` and `Guardrail` classes, which have the worst segmentation results, are significantly improved when $R_{\max} > 1$. The model achieves the highest mAcc and mIoU when $R_{\max} = 2$, but segmentation performance gradually deteriorates as $R_{\max}$ increases further. This indicates that while training with both synthetic and real images can enhance segmentation performance, an excessive number of synthetic images can lead to low quality issues and overfitting, thus degrading model performance. Specifically, although synthetic data increases the diversity of the training set, the quality of synthetic images is generally inferior to that of real images. As the number of synthetic data increases, the model may learn more noises and inaccurate features.

As shown in Table III, we conduct the experiments to analyze the effect of different sampling mechanisms for synthetic and real data on model performance. We set $R_{\max} = 2$ and sampled 2,744 synthetic RGB-T images for joint training. In Table III, the first row shows the segmentation results when the sampling mechanism for synthetic data considers only the pixel ratio of each class. The second row shows the results when it considers only the segmentation difficulty of each class. Both rows oversample the real training set, resulting in 2,744 pairs of real RGB-T images used for joint training, to ensure the training size of synthetic and real data are the same. The results in the first two rows are better than those obtained by training only with real datasets (first row of Table I). This demonstrates that using either the pixel ratio or segmentation difficulty for synthetic data improves the ability of the model. However, their segmentation performance is inferior to the fourth row, which considers both pixel ratio and segmentation difficulty in the synthetic data sampling mechanism. This proves that considering both factors can simultaneously maximize the segmentation ability of the model.

Compared to the fourth row, which achieves the best segmentation performance, the third row does not oversample the real data. It retains 1,568 pairs of real RGB-T images for joint training and, as a result, obtains worse mIoU, particularly for the `Car Stop` and `Guardrail` classes. These classes have scarce samples and are more difficult to segment. Sample scarcity means fewer training samples for the generative model, and harder segmentation indicates greater difficulty in extracting semantic features. Consequently, such classes yield poorer quality synthetic data from the generative model. When the scale of synthetic data outweighs that of real data in joint training, these classes further suppress segmentation accuracy compared to other classes. The `Car Stop` and `Guardrail` classes unexpectedly achieved the highest Acc while obtaining the lowest IoU. This suggests that false positives dominated their segmentation results, incorrectly judging negative cases as positive. This further validates the effect of lower-quality synthetic data on the segmentation results. The above results demonstrate the effectiveness of oversampling real data to match the scale of synthetic data.

C. Comparative Study

Based on the ablation experiments described above, we construct the synthetic dataset used in the comparative experiments

TABLE II
THE RESULTS (%) OF THE ABLATION STUDY ON THE MAXIMUM SAMPLING NUMBER $R_{\max}$

R_{max}	Car		Person		Bike		Curve		Car Stop		Guardrail		Color Cone		Bump		mAcc	mIoU
	Acc	IoU	Acc	IoU	Acc	IoU	Acc	IoU	Acc	IoU	Acc	IoU	Acc	IoU	Acc	IoU		
1	93.0	88.3	83.0	73.5	**74.2**	65.0	**66.1**	51.2	41.1	31.4	36.1	8.2	56.8	**52.9**	69.2	57.7	68.8	58.5
2	92.9	88.3	**84.2**	**74.9**	73.4	**65.2**	64.6	51.3	43.5	36.3	63.6	14.2	**57.9**	52.8	**74.1**	**62.1**	**72.6**	**60.4**
3	93.0	88.4	83.3	74.0	72.7	64.0	65.8	**51.7**	44.7	36.1	**68.2**	**14.3**	56.4	52.6	68.3	57.4	72.4	59.7
4	**93.7**	**88.9**	82.7	73.5	73.8	65.0	62.8	51.6	**50.6**	**38.6**	53.6	12.3	54.2	51.7	69.3	54.6	71.1	59.4
5	92.8	88.0	83.7	74.4	74.0	64.9	63.4	51.0	46.3	34.5	59.1	12.3	55.3	50.8	70.7	56.6	71.6	59.0

The best results are highlighted in bold.

TABLE III
THE RESULTS (%) OF THE ABLATION STUDY ON THE SAMPLING MECHANISM

Sampling			Car		Person		Bike		Curve		Car Stop		Guardrail		Color Cone		Bump		mAcc	mIoU
P_c	μ_c	OverSamp	Acc	IoU	Acc	IoU	Acc	IoU	Acc	IoU	Acc	IoU	Acc	IoU	Acc	IoU	Acc	IoU		
✓		✓	**93.1**	**88.3**	83.1	73.9	73.3	64.8	64.8	50.5	48.2	**37.5**	54.9	14.0	54.7	49.9	71.5	55.3	71.4	59.2
	✓	✓	92.7	87.9	82.6	73.7	71.2	63.3	62.9	51.0	44.0	36.4	**65.1**	**16.0**	56.4	51.7	64.9	57.1	71.0	59.5
✓	✓		92.3	87.9	83.1	74.0	**74.8**	64.7	**67.0**	51.1	**49.4**	34.0	53.4	12.0	55.9	51.9	70.9	59.2	71.8	59.2
✓	✓	✓	92.9	**88.3**	**84.2**	**74.9**	73.4	**65.2**	64.6	**51.3**	43.5	36.3	63.6	14.2	**57.9**	**52.8**	**74.1**	**62.1**	**72.6**	**60.4**

P_c is the pixel ratio of each class. μ_c is the mean loss of each class. A ✓ under P_c or μ_c indicates whether P_c or μ_c is included in the sampling mechanism for synthetic data. 'OverSamp' indicates whether the real training set is oversampled to ensure the training size of synthetic and real data are the same. The best results are highlighted in bold.

TABLE IV
THE PER-CLASS RESULTS (%) OF MODELS USING DIFFERENT METHOD TO ALLEVIATE CLASS IMBALANCE

Methods	Car		Person		Bike		Curve		Car Stop		Guardrail		Color Cone		Bump		mAcc	mIoU
	Acc	IoU	Acc	IoU	Acc	IoU	Acc	IoU	Acc	IoU	Acc	IoU	Acc	IoU	Acc	IoU		
CMX(MiT-B2) [5]	92.2	**89.4**	81.3	74.8	73.4	64.7	63.5	47.3	38.8	30.1	36.3	8.1	53.3	52.4	67.7	59.4	67.3	58.2
Focal Loss [25]	**93.9**	88.1	**84.8**	73.6	**74.5**	63.9	**68.8**	49.1	42.6	31.7	51.6	9.4	55.6	51.0	70.6	59.5	71.3	58.3
Resampling	93.4	87.7	**84.8**	74.1	73.5	64.1	63.1	47.7	**45.0**	27.0	**66.1**	11.0	**58.3**	52.4	68.3	54.7	72.4	57.4
SyntheticSeg (Ours)	92.9	88.3	84.2	**74.9**	73.4	**65.2**	64.6	**51.3**	43.5	**36.3**	63.6	**14.2**	57.9	**52.8**	**74.1**	**62.1**	**72.6**	**60.4**

The best results are highlighted in bold.

by setting $R_{\max} = 2$ and sampling from the generated large-scale synthetic dataset. We also oversample the real dataset to match the size of the synthetic dataset.

1) The Quantitative Results: In Table IV, we compare our SyntheticSeg with focal loss [25] and resampling. All the three methods use CMX [5] with MiT-B2 as the baseline. Focal loss is implemented with $\gamma = 2$, emphasizing hard-to-classify examples. The resampling method oversamples rare samples during training to emphasize classes with fewer instances. While focal loss achieves a higher IoU than the baseline for `Curve`, `Car Stop`, `Guardrail` and `Bump`, it performs poorly on other classes. Resampling only outperforms the baseline in `Curve` and `Guardrail`, resulting in a lower mIoU than the baseline. Our method, however, not only significantly improves segmentation accuracy for objects with few samples and high segmentation difficulty but also performs well across other classes.

In Table V, we select three RGB-T semantic segmentation models and compare them by training only with real data versus training with both synthetic data from our proposed method and real data. We train each model with multiple backbones to analyze the impact of different training data on various architectures and backbone models. The first row for each model and backbone uses only the real training set, while the second row uses both the real and synthetic training sets. After incorporating synthetic training data, the segmentation ability of RTFNet has been significantly improved. Specifically, the IoU of the `Bump` class in RTFNet with ResNet-152 is the only one that decreases, while other segmentation results show improvement. For the three models with different backbones, the mIoU has consistently been improved after training with both synthetic and real data. More importantly, these models (except for CRM with Swin-B) notably enhanced the segmentation IoU of the `Car Stop` and `Guardrail` classes, which have few samples and are challenging to segment. This aligns with the intended outcomes of our proposed sampling mechanism. The results show that our method consistently improves the overall segmentation performance.

2) The Qualitative Demonstrations: To better demonstrate the effectiveness of our SyntheticSeg, we select 4 samples taken at night and 4 samples taken during the day, which are displayed in Fig. 5. After training with synthetic and real data, the ability to segment uncommon and difficult-to-segment classes of the existing models has significantly been improved. For example, the models trained with synthetic and real data in the second cases can better segment the `Guardrail` class, which has the lowest pixel ratio. This result aligns with the conclusions from the quantitative experiments above. We also find that training with synthetic and real data improved the resistance to false detection of the models. For example, CMX trained with only real data in the sixth, seventh, and eighth cases detect the background as the `Curve` class. This issue is significantly

TABLE V
THE PER-CLASS RESULTS (%) OF MODELS USING DIFFERENT TRAINING SETS

Model	Backbone	Training Data		IoU								mIoU	Δ
		Real	Synthetic	Car	Person	Bike	Curve	Car Stop	Guardrail	Color Cone	Bump		
RTFNet [4]	ResNet-50	✓		86.3	67.8	58.2	43.7	24.3	3.6	26.0	57.2	51.7	↑ 4.4
		✓	✓	**86.5**	**69.9**	**60.8**	**49.5**	**27.8**	**8.7**	**46.0**	**58.0**	**56.1**	
	ResNet-152	✓		87.4	70.3	62.7	45.3	29.8	0.0	29.1	**55.7**	53.2	↑ 3.7
		✓	✓	**88.6**	**71.1**	**63.3**	**50.2**	**30.5**	**8.0**	**50.2**	51.7	**56.9**	
CMX [5]	MiT-B2	✓		**89.4**	74.8	64.7	47.3	30.1	8.1	52.4	59.4	58.2	↑ 2.2
		✓	✓	88.3	**74.9**	**65.2**	**51.3**	**36.3**	**14.2**	**52.8**	**62.1**	**60.4**	
	MiT-B4	✓		**90.1**	**75.2**	64.5	50.2	35.3	8.5	**54.2**	60.6	59.7	↑ 1.2
		✓	✓	89.4	74.9	**65.0**	**54.6**	**38.7**	**13.6**	53.1	**60.8**	**60.9**	
CRM [6]	Swin-T	✓		90.0	73.1	63.7	47.9	40.7	9.9	**54.4**	**54.2**	59.1	↑ 0.8
		✓	✓	**90.3**	**75.0**	**64.1**	**49.4**	**48.6**	**11.1**	50.6	52.1	**59.9**	
	Swin-S	✓		90.6	**75.5**	**67.2**	48.3	43.4	11.8	**56.8**	**59.3**	61.2	↑ 0.8
		✓	✓	**90.7**	**75.5**	65.2	**52.0**	**50.4**	**15.4**	54.8	55.2	**62.0**	
	Swin-B	✓		90.0	**75.1**	**67.0**	45.2	**49.7**	**18.4**	**54.2**	54.4	61.4	↑ 0.7
		✓	✓	**91.2**	74.4	65.2	**50.9**	46.4	15.8	**54.2**	**62.0**	**62.1**	

We used RTFNet [4], CMX [5], and CRM [6] with different backbones as the RGB-T segmentation models. Δ represents the mIoU improvement when training with both synthetic and real data compared to using only real data. The best results of each backbone are highlighted in bold.

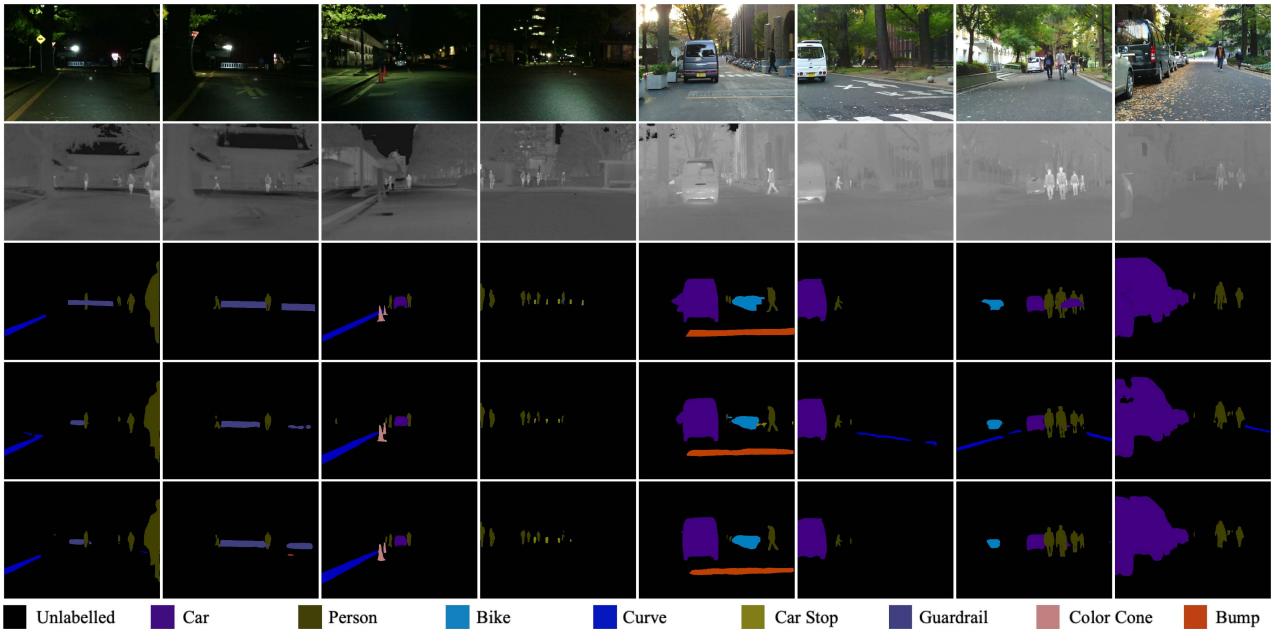

Fig. 5. Qualitative comparison for semantic segmentation of RGB-T images on MFNet [2] dataset. The rows from top to bottom are RGB images, thermal images, ground truth of semantic layouts, results obtained by training CMX [5] with MiT-B2 using real data, and results obtained by training CMX [5] with MiT-B2 using real and sampled synthetic data. The first four columns and the last four columns are the samples of nighttime and daytime, respectively.

reduced after adding synthetic data to the training process. This improvement is due to the increased number and diversity of training samples from the synthetic dataset.

3) Suboptimal Case Analysis: Although the above quantitative and qualitative analyses have proved the effectiveness of our method, there are still cases where the segmentation performance is not satisfactory, as shown in Fig. 6. Specifically, Fig. 6(f) illustrates that the prediction results are not accurate for both the `Person` class with a large sample size and the `Car Stop` class with a small sample size. The first reason is that the image quality of distant objects in real RGB images is poor. As seen in Fig. 6(a), the model struggles to extract rich semantic information and accurate contours of distant persons in dark environments. Besides, for classes with a small sample size, the quality of synthetic images generated by the model is suboptimal. The generated `Car Stop` and `Color Cone` classes in Fig. 6(d) and (e) and their corresponding semantic layouts in Fig. 6(e) lack spatial consistency, which means that if the synthetic images dominate the training, the segmentation accuracy will be suppressed. In addition, since the synthetic RGB and thermal images are obtained separately through the generative model, their background parts cannot guarantee spatial consistency. Despite our method has been proven to effectively improve the segmentation

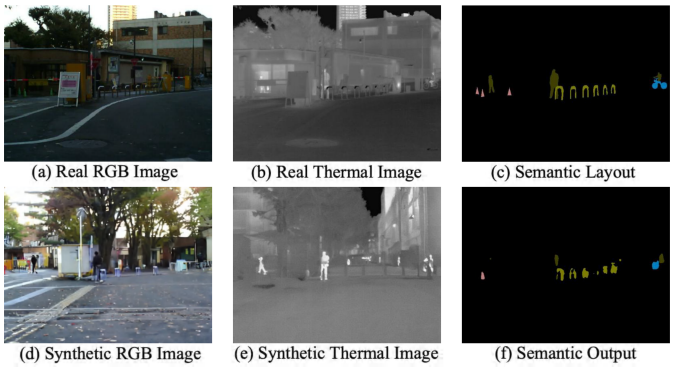

Fig. 6. Visualization of the segmentation suboptimal case. (a) and (b) are real RGB-T images from MFNet dataset; (c) is the corresponding semantic layout; (d) and (e) are synthetic RGB-T images generated from (c); (f) is the prediction of segmentation model.

performance, this spatial inconsistency may limit the further improvement.

V. Conclusions and Future Work

We proposed here a data-driven method, SyntheticSeg, to enhance RGB-T segmentation by using synthetic data augmentation. We created high-quality synthetic RGB-T images and built a large-scale dataset to diversify training samples. The new metric we introduced effectively guides sampling from synthetic datasets by considering the scarcity of semantic layouts and the difficulty of segmentation. This method not only alleviates the class imbalance problem but also improves the overall segmentation accuracy. The experimental results show that our method achieves state-of-the-art performance on the MFNet dataset.

However, our method has some limitations. First, the quality of synthetic images is still inferior to real images, particularly for scarce classes, leading to suppressed segmentation performance when synthetic images dominate the training. Second, the synthetic RGB and thermal images cannot be accurately aligned, especially in the background, and their physical consistency is uncertain. Our future work would focus on developing a generative model that generates synthetic RGB-T image pairs with improved physical and spatial consistency, potentially enhancing the segmentation performance.

References

[1] Z. Feng, Y. Guo, D. Navarro-Alarcon, Y. Lyu, and Y. Sun, "InconSeg: Residual-guided fusion with inconsistent multi-modal data for negative and positive road obstacles segmentation," *IEEE Robot. Autom. Lett.*, vol. 8, no. 8, pp. 4871–4878, Aug. 2023.

[2] Q. Ha, K. Watanabe, T. Karasawa, Y. Ushiku, and T. Harada, "MFNet: Towards real-time semantic segmentation for autonomous vehicles with multi-spectral scenes," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst*, IEEE, 2017, pp. 5108–5115.

[3] Z. Feng, Y. Guo, and Y. Sun, "CEKD: Cross-modal edge-privileged knowledge distillation for semantic scene understanding using only thermal images," *IEEE Robot. Autom. Lett.*, vol. 8, no. 4, pp. 2205–2212, Apr. 2023.

[4] Y. Sun, W. Zuo, and M. Liu, "RTFNet: RGB-thermal fusion network for semantic segmentation of urban scenes," *IEEE Robot. Autom. Lett.*, vol. 4, no. 3, pp. 2576–2583, Jul. 2019.

[5] J. Zhang, H. Liu, K. Yang, X. Hu, R. Liu, and R. Stiefelhagen, "CMX: Cross-modal fusion for RGB-X semantic segmentation with transformers," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 12, pp. 14679–14694, Dec. 2023.

[6] U. Shin, K. Lee, I. S. Kweon, and J. Oh, "Complementary random masking for RGB-thermal semantic segmentation," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2024, pp. 11110–11117.

[7] W. Ma, S. Huang, and Y. Sun, "Triplet-graph: Global metric localization based on semantic triplet graph for autonomous vehicles," *IEEE Robot. Autom. Lett.*, vol. 9, no. 4, pp. 3155–3162, Apr. 2024.

[8] H. Xu, H. Liu, S. Meng, and Y. Sun, "A novel place recognition network using visual sequences and LiDAR point clouds for autonomous vehicles," in *Proc. IEEE Int. Conf. Intell. Transp. Syst.*, IEEE, 2023, pp. 2862–2867.

[9] Y. Feng, W. Hua, and Y. Sun, "NLE-DM: Natural-language explanations for decision making of autonomous driving based on semantic scene understanding," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 9, pp. 9780–9791, Sep. 2023.

[10] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Adv. Neural Inf. Process. Syst.*, vol. 33, pp. 6840–6851, 2020.

[11] P. Dhariwal and A. Nichol, "Diffusion models beat GANs on image synthesis," *Adv. Neural Inf. Process. Syst.*, vol. 34, pp. 8780–8794, 2021.

[12] W. Wang et al., "Semantic image synthesis via diffusion models," 2022, arXiv: *2207.00050*.

[13] H. Xue, Z. Huang, Q. Sun, L. Song, and W. Zhang, "Freestyle layout-to-image synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 14256–14266.

[14] L. Abdi and S. Hashemi, "To combat multi-class imbalanced problems by means of over-sampling techniques," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 1, pp. 238–251, Jan. 2016.

[15] J. A. Sáez, J. Luengo, J. Stefanowski, and F. Herrera, "SMOTE–IPF: Addressing the noisy and borderline examples problem in imbalanced classification by a re-sampling method with filtering," *Inf. Sci.*, vol. 291, pp. 184–203, 2015.

[16] H. K. Choi, J. Choi, and H. J. Kim, "TokenMixup: Efficient attention-guided token-level data augmentation for transformers," *Adv. Neural Inf. Process. Syst.*, vol. 35, pp. 14224–14235, 2022.

[17] S. Venkataramanan, E. Kijak, L. Amsaleg, and Y. Avrithis, "AlignMixup: Improving representations by interpolating aligned features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 19174–19183.

[18] J. Tian, N. C. Mithun, Z. Seymour, H.-P. Chiu, and Z. Kira, "Striking the right balance: Recall loss for semantic segmentation," in *Proc. IEEE Int. Conf. Robot. Autom.*, IEEE, 2022, pp. 5063–5069.

[19] S. Qiu et al., "Subclassified loss: Rethinking data imbalance from subclass perspective for semantic segmentation," *IEEE Trans. Intell. Veh.*, vol. 9, no. 1, pp. 1547–1558, Jan. 2024.

[20] Y. Sun, W. Zuo, P. Yun, H. Wang, and M. Liu, "FuseSeg: Semantic segmentation of urban scenes based on RGB and thermal data fusion," *IEEE Trans. Autom. Sci. Eng.*, vol. 18, no. 3, pp. 1000–1011, Jul. 2021.

[21] H. Li and Y. Sun, "IGFNet: Illumination-guided fusion network for semantic scene understanding using RGB-thermal images," in *Proc. IEEE Int. Conf. Robot. Biomimetics*, IEEE, 2023, pp. 1–6.

[22] M. Liang, J. Hu, C. Bao, H. Feng, F. Deng, and T. L. Lam, "Explicit attention-enhanced fusion for RGB-thermal perception tasks," *IEEE Robot. Autom. Lett.*, vol. 8, no. 7, pp. 4060–4067, Jul. 2023.

[23] J. Huang et al., "RoadFormer+: Delivering RGB-X scene parsing through scale-aware information decoupling and advanced heterogeneous feature fusion," *IEEE Trans. Intell. Veh.*, early access, Aug. 22, 2024, doi: 10.1109/TIV.2024.3448251.

[24] H. Li, H. K. Chu, and Y. Sun, "Temporal consistency for RGB-thermal data-based semantic scene understanding," *IEEE Robot. Autom. Lett.*, vol. 9, no. 11, pp. 9757–9764, Nov. 2024.

[25] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2980–2988.

[26] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. Belongie, "Class-balanced loss based on effective number of samples," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 9268–9277.

[27] H. Caesar, J. Uijlings, and V. Ferrari, "COCO-stuff: Thing and stuff classes in context," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1209–1218.

[28] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba, "Scene parsing through ADE20 K dataset," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 633–641.