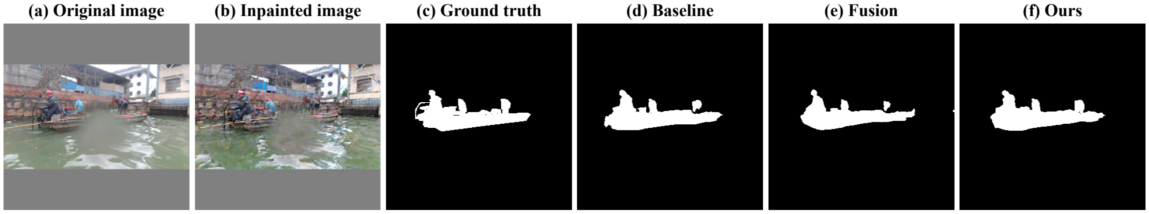
CaRaFFusion: Improving 2D Semantic Segmentation with Camera-Radar Point Cloud Fusion and Zero-Shot Image Inpainting

Huawei Sun[⋆1,2], Bora Kunter Sahin[⋆1,3], Georg Stettinger[1], Maximilian Bernhard[3],
Matthias Schubert[3], Robert Wille[2]



(a) Original image (b) Inpainted image (c) Ground truth (d) Baseline (e) Fusion (f) Ours

Fig. 1: Our method can segment the objects in very adverse conditions where other methods fail.

Abstract— **Segmenting objects in an environment is a crucial task for autonomous driving and robotics, as it enables a better understanding of the surroundings of each agent. Although camera sensors provide rich visual details, they are vulnerable to adverse weather conditions. In contrast, radar sensors remain robust under such conditions, but often produce sparse and noisy data. Therefore, a promising approach is to fuse information from both sensors. In this work, we propose a novel framework to enhance camera-only baselines by integrating a diffusion model into a camera-radar fusion architecture. We leverage radar point features to create pseudo-masks using the Segment-Anything model, treating the projected radar points as point prompts. Additionally, we propose a noise reduction unit to denoise these pseudo-masks, which are further used to generate inpainted images that complete the missing information in the original images. Our method improves the camera-only segmentation baseline by 2.63% in mIoU and enhances our camera-radar fusion architecture by 1.48% in mIoU on the Waterscenes dataset. This demonstrates the effectiveness of our approach for semantic segmentation using camera-radar fusion under adverse weather conditions.**

I. Introduction

Robust semantic segmentation is vital for applications like autonomous driving [1], robotics [2], and medical imaging [3]. Although camera-based models excel in ideal conditions, their performance deteriorates under adverse weather, leading to inconsistent boundaries and incomplete segmentations [4]. Multi-modal integration, particularly radar-camera fusion, has emerged as a promising solution to enhance segmentation robustness [5].

Radars are resilient to challenging conditions, making them ideal for tasks like object detection [6, 7], depth estimation [8, 9], and BEV segmentation [10, 11]. However, radar data is sparse and lacks fine-grained details, complicating its use for image-plane segmentation. Integrating radar's spatial information with RGB's rich visual detail poses challenges, including differences in data structure, resolution, and noise from reflections on surfaces like water.

Thus, this paper proposes CaRaFFusion, a three-stage framework for robust radar-camera fusion-based segmentation. In the first stage, spatial and visual features from the radar and camera inputs are fused via cross-attention [12] to generate initial segmentation masks. The second stage uses MobileSAM [13] to refine these masks with radar point prompts, improving robustness to weather-induced noise. To handle radar's inherent noise, we introduce a Noise Reduction Unit (NRU) that filters unwanted reflections, producing cleaner masks. Finally, a diffusion model [14] inpaints missing information, enhancing segmentation performance. Dual Segformer encoders [1] process the inpainted images and original inputs to deliver high-quality segmentation outputs.

In summary, our key contributions are as follows:

- We propose CaRaFFusion, a three-stage radar-camera fusion framework designed for robust segmentation under adverse conditions.
- A Noise Reduction Unit (NRU) is invented to filter radar noise and improve segmentation mask accuracy.
- We utilize the Image Inpainting strategy to restore information lost in adverse weather, further enhancing segmentation performance.

We train and evaluate CaRaFFusion on the WaterScenes dataset [15], which includes radar point cloud data for 2D semantic segmentation. Results show that our approach outperforms both camera-only and existing fusion-based methods, particularly in challenging scenarios.

II. Related Work

This section reviews semantic segmentation algorithms across image-only, radar-only, and image-radar fusion methods, and discusses generative approaches for segmentation.

A. Semantic Segmentation

We categorize semantic segmentation techniques into image-only, radar-only, and fusion-based methods.

[1] Infineon Technologies AG, Neubiberg, Germany, {huawei.sun, georg.stettinger}@infineon.com
[2] Technical University of Munich, Munich, Germany
[3] Ludwig Maximilian University of Munich, Munich, Germany, {b.sahin}@campus.lmu.de

[*] These authors contributed equally to this work

1) Image-Based Methods: Image-based semantic segmentation assigns classes to image pixels, with boundary information playing a critical role in accuracy. Early approaches like Fully Convolutional Networks (FCNs) [16] laid the foundation for CNN-based segmentation. Subsequent improved segmentation accuracy through advanced decoder modules [17, 18] and lightweight architectures for real-time performance [19, 20]. Transformer-based architectures [12] further advanced segmentation [1, 21] by expanding perceptual fields, yielding greater accuracy. However, image-only methods often falter in adverse conditions, where domain adaptation [22, 23] is used to transfer knowledge between weather domains. Integrating radar with camera data could address these limitations.

2) Radar-based Semantic Segmentation: Radar-only segmentation methods are classified into tensor-based and point-cloud-based approaches. Tensor-based methods [24, 25] process and segment radar Range-Angle-Doppler (RAD) tensor, similar to the image segmentation task, whereas point cloud segmentation assigns labels to individual points. Existing algorithms [26]–[28] focus on extracting reliable radar-specific features and mitigating the sparsity and noisy issues.

3) Camera-Radar Fusion-based Semantic Segmentation: Fusion-based methods integrate complementary information from radar and camera for segmentation tasks. CMG-GAN [29] fuses radar-generated images with RGB data, while SO-NET [30] and MCAF-Net [31] project radar points onto the image plane for detection and segmentation. BEV-based methods [10, 11] jointly process radar and camera data. Achelous [32] integrates radar and monocular camera data for object detection and segmentation but primarily utilizes radar for detection, leaving image features for segmentation. Effectively leveraging radar point clouds for dense image-plane segmentation remains an unresolved challenge.

To the best of our knowledge, no existing research has yet explored how radar point clouds can support semantic segmentation in the image plane. Given the inherent sparsity of radar data and the lack of shape information in point cloud data, effectively leveraging radar point clouds for image-plane semantic segmentation remains an open challenge.

B. Generative Methods in Segmentation

Generative models are increasingly applied to improve semantic segmentation. GMMSeg [33] combines generative and discriminative approaches, leveraging Gaussian Mixture Models to enhance robustness to out-of-distribution data. FissGAN [34] employs dual GANs to handle foggy conditions, generating edge information and semantic features.

Inpainting methods reconstruct missing image regions using segmentation masks. SGE-Net [35] uses segmentation maps for structural refinement, ensuring coherent inpainted results. Inpaint Anything [36] integrates Segment-Anything [37] and Stable Diffusion [14] to refine specific regions. While promising, these methods are yet to realize their full potential in enhancing semantic segmentation performance for camera-radar fusion.

III. Approach

This section begins by presenting the overall architecture of the proposed three-stage CaRaFFusion model. We then describe each stage in detail: the Camera-Radar Fusion stage, the Pseudo-Mask Generation and Mask Denoising stage, and the Image Inpainting and Final Mask Prediction stage. Lastly, we introduce the loss function used to train the networks.

A. Model Architecture

The proposed framework consists of three main stages, designed to process radar points and camera images to produce accurate segmented masks.

In the first stage, an initial camera-radar fusion model is introduced for the semantic segmentation task. This model takes the radar point cloud and RGB image as inputs, which are processed by separate encoders. The extracted features are fused through a cross-attention strategy [12] and passed to the Segformer decoder, producing initial segmentation masks M_{init}. Simultaneously, the radar features are processed by a PointNet [38] decoder for point cloud classification.

The second stage focuses on generating more robust masks for the third stage. Here, MobileSAM [13] is employed to generate additional SAM masks M_{sam}, using the image as input and radar points as prompts. This leverages the radar sensor's robustness to detect unseen objects, which may be obscured by blurriness or water droplets in the image. However, radar noise presents challenges, as object-bound radar points projected onto the image plane may lie on water surfaces, creating noisy SAM masks. To address this, we propose a Noise Reduction Unit (NRU) designed to filter out radar projection noise and refine the MobileSAM-generated masks, yielding M_{nr} masks for the next stage.

The third stage takes the refined masks M_{nr}, the RGB image, and predicted object types from the radar point cloud classifier as inputs to generate an inpainted image. This process utilizes the power of a diffusion model to fill regions of the image blurred or obscured by water droplets. The inpainted image serves as an additional modality, processed by a separate Segformer encoder. Features encoded from both the original and inpainted images are concatenated and sent to the decoder, outputting the final segmentation masks.

B. Stage 1: Camera-Radar Fusion

This stage takes radar and camera data as inputs, utilizing the multi-task learning strategy to predict segmentation masks and classify the radar point cloud simultaneously. **Segmentation:** The radar data, represented as points with shape $(N_p, 5)$, are processed through a PointNet [38] encoder to extract spatial features. Here, N_p represents the number of radar points, and 5 corresponds to the five input features: the 3D positions of the points, Radar Cross Section, and Doppler velocities. The resulting radar features F^i_{radar} have a shape of (N, C^i_r), where N is the number of radar point features, and C^i_r is the number of feature channels.

Simultaneously, the RGB image I_{rgb} is processed by a Segformer encoder [1], which extracts rich visual features F^i_{img} with a shape of (H^i, W^i, C^i_I), where H^i and W^i denote

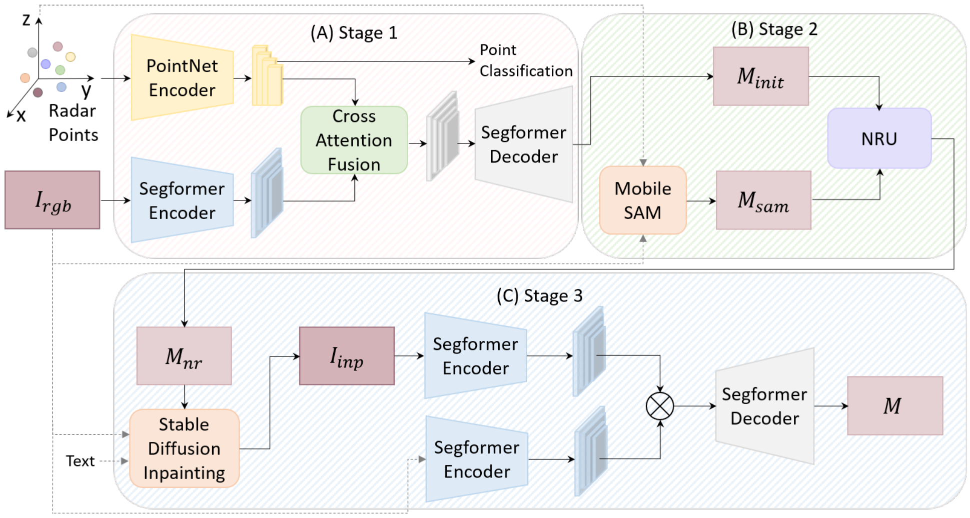


Fig. 2: Model Architecture: Our three-stage framework combines radar and camera data for robust segmentation, especially in challenging conditions. First, radar and image features are extracted and fused through cross-attention to produce an initial segmentation mask. To improve resilience, MobileSAM generates an additional mask using radar points as prompts, and the Noise Reduction Unit (NRU) refines this by filtering radar-related noise. Finally, dual Segformer Encoders further enhance the refined mask, producing a high-quality segmentation output.

the height and width of the image features, and C_I^i is the number of image feature channels, with $C_r^i = C_I^i$. Here, $i \in \{1, 2, 3, 4\}$ refers to the layer index. To enhance the feature representations, we applied a cross-attention mechanism [12] from the image features to the radar features. In our setup, we use the image features as query Q_{img} and the radar features as the key K_{radar} and value V_{radar} matrices. The Cross-Attention Fusion (CAF) process is represented as follows:

$$Q_{\text{img}}^i = F_{\text{img}}^i W_Q^i, \tag{1}$$

$$K_{\text{radar}}^i = F_{\text{radar}}^i W_K^i, \tag{2}$$

$$V_{\text{radar}}^i = F_{\text{radar}}^i W_V^i, \tag{3}$$

$$\text{Attention}(Q_{\text{img}}^i, K_{\text{radar}}^i, V_{\text{radar}}^i) = \text{softmax}\left(\frac{Q_{\text{img}}^i (K_{\text{radar}}^i)^T}{\sqrt{C_I^i}}\right) V_{\text{radar}}^i, \tag{4}$$

where $Q_{\text{img}}^i \in \mathbb{R}^{C_I^i \times d}$ and $K_{\text{radar}}^i, V_{\text{radar}}^i \in \mathbb{R}^{C_r^i \times d}$. Q_{img}^i are set to the extracted intermediate Segformer encoder features and K_{radar}^i, V_{radar}^i are set to the extracted intermediate PointNet encoder features. Then we get the final fused feature F^i:

$$F^i = Q_{\text{img}}^i + \text{Attention}(Q_{\text{img}}^i, K_{\text{radar}}^i, V_{\text{radar}}^i) \tag{5}$$

The fused features $F^i, i \in \{1, 2, 3, 4\}$ generated by the CAF module are subsequently passed through a Segformer decoder, which produces the initial segmentation mask M_{init}. This mask integrates spatial information from radar data with visual context from the camera image, providing a solid foundation for further refinement in subsequent stages.

Radar Point Cloud Segmentation: We train the radar point cloud segmentation task alongside the segmentation task for two main reasons. First, as shown in previous studies [11, 31], a multitask learning strategy enables the model to extract more meaningful features, thus improving the performance of the primary task. Second, point cloud segmentation prepares for the second stage, where the MobileSAM model uses radar points as prompts to generate additional masks. Since MobileSAM is an instance segmentation model, it produces a binary mask for each prompt without identifying the object's category. By using the predicted class of a given radar point, we can assign a class label to each segmented binary mask.

The point cloud segmentation model functions as follows: encoded radar point features are first extracted by the PointNet encoder, which captures relevant spatial and contextual patterns. These features are then fed into a classification head consisting of multiple Multi-Layer Perceptrons (MLPs) followed by a softmax layer to classify the points.

C. Stage 2: Pseudo-Mask Generation and Mask Denoising

In stage 2, we improve M_{init} by addressing limitations that arise under challenging conditions. Although radar data remain robust in adverse weather, they lack the fine detail needed to accurately outline object shapes, which is essential for segmentation. Consequently, despite the fusion of both modalities in stage 1, the predicted masks may still miss parts of objects. For example, in poor weather conditions, such as rain or water droplets in the lens, M_{init} may struggle to maintain accuracy due to interference in visual data.

To supplement M_{init}, we employ MobileSAM [13], a lightweight Segment Anything Model, to generate additional masks M_{sam} using radar points as prompts. This approach leverages the radar's robustness under adverse conditions, helping to produce masks that are less affected by weather-related interference. However, radar data can introduce noise, as radar points projected onto a 2D plane may inaccurately outline object boundaries, with some points positioned on water surfaces or other extraneous areas. This can create artifacts in M_{sam}, introducing excessive noise that impacts the third stage.

Since our first stage model segments the background $M_{\text{background}}$ and M_{water}, which are extracted from the first and last channel of M_{init}, with high accuracy, we can use M_{init} to remove noisy artifacts from M_{sam}. To accomplish this, we introduce a simple Noise Reduction Unit (NRU). This module removes the background $M_{\text{background}}$ and waterline M_{water} from M_{sam} on a channel-wise basis, then adds the segmented object parts from M_{init} back into M_{sam} to create the final pseudo-mask M_{nr}. This refined mask M_{nr} is used by the inpainting model in the next stage. The NRU process can be formally defined as shown in the following:

1) **Step (1):** We define the noise mask M_{noise} as the sum of two components, $M_{\text{background}}$ and M_{water}:

$$M_{\text{noise}} := M_{\text{background}} + M_{\text{water}}.$$

This step identifies the noise sources within the mask by combining background and water areas, which are often sources of irrelevant or misleading information in segmentation.

2) **Step (2):** The denoised mask M_{denoised} is computed by subtracting the noise mask M_{noise} from the initial mask M_{sam} generated by MobileSAM:

$$M_{\text{denoised}} := M_{\text{sam}} - M_{\text{noise}}.$$

This operation removes unwanted background and water information, retaining only relevant regions that better represent the object.

3) **Step (3):** A ReLU activation function is applied to M_{denoised}, resulting in the final denoised mask M_{sam}:

$$M_{\text{sam}} := \text{ReLU}(M_{\text{denoised}}).$$

ReLU effectively removes any negative values that may have appeared from the subtraction, ensuring that only non-negative values are kept.

4) **Step (4):** The merged mask M_{merged} is then created by combining the refined mask M_{sam} with the initial segmentation mask M_{init} from Stage 1:

$$M_{\text{merged}} := M_{\text{sam}} + M_{\text{init}}.$$

This merging step leverages the strengths of both the refined MobileSAM output and the initial mask to create a more comprehensive representation.

5) **Step (5):** Finally, the noise-reduced mask M_{nr} is obtained by applying a clamping operation to M_{merged}, restricting its values within the range $[0, 1]$:

$$M_{\text{nr}} := \text{clamp}(M_{\text{merged}}, 0, 1).$$

This step ensures that the mask values remain within valid bounds, enhancing stability of the final output.

Each of these steps is applied channel-wise, allowing the model to handle multi-channel data independently for more precise noise reduction and mask refinement.

D. Stage 3: Image Inpainting and Final Mask Prediction

The final stage aims to generate robust segmentation masks by leveraging the capabilities of a zero-shot generative model. First, the Stable Diffusion model [14] takes the mask output M_{nr} from stage 2, the related text prompt of the mask and the original input image I_{img} as inputs, producing an inpainted image I_{inp} of the same size as the original image. This process serves multiple purposes: during adverse weather conditions, such as rain, images often become blurry and are affected by water droplets. The robust masks M_{nr} generated in stage 2 mitigate these effects by focusing on reliable object regions, allowing the inpainting model to generate plausible object details in the masked areas. The inpainting process is summarized in Algorithm 1.

Subsequently, the inpainted image I_{inp} and the original image I_{img} are processed separately by two distinct Segformer encoders. This parallel encoding enables the model to capture information from both the original image and the enhanced inpainted image, which is especially beneficial under challenging weather conditions. The outputs from these encoders are concatenated to integrate spatial and semantic features from both sources and then fed into a Segformer decoder, which generates the final segmentation mask M for each class.

This final decoding step synthesizes the combined features, producing an accurate and consistent mask with reduced noise and enhanced detail. By leveraging multimodal refinement, where the inpainted and original images complement each other, the model achieves a high-quality segmentation output even in adverse conditions.

Algorithm 1 Iterative Inpainting

```
Require: Image $I_{\text{img}}$, Set of text prompts $\{P_1, P_2, \ldots, P_N\}$,
    Set of masks $\{M_1, M_2, \ldots, M_N\}$, Stable Diffusion model SD
Initialize inpainted image $I \leftarrow I_{\text{img}}$
for each mask $M_i$ in $\{M_1, M_2, \ldots, M_N\}$ do
    $I \leftarrow \text{SD}(I, M_i, P_i)$
end for
return $I$
```

E. Loss Functions

Two loss functions are used to train our network. For radar point cloud segmentation in stage 1, we apply Focal Loss [39], defined as follows:

$$L_{\text{cls}} = -\sum_{c=1}^{C} \alpha_c (1 - p_c)^{\gamma} \log(p_c), \tag{6}$$

where p_c represents the predicted probability of class c, and C denotes the number of classes. We choose the parameters α_c according to the relative class frequency of class c and set $\gamma = 2$ to balance class distribution and emphasize harder samples over easier ones, respectively [39].

For 2D semantic segmentation, we use the Dice loss [40], defined as follows:

$$L_{\text{seg}} = 1 - \frac{2 \sum_{i=1}^{N} p_i g_i}{\sum_{i=1}^{N} p_i + \sum_{i=1}^{N} g_i}, \tag{7}$$

where p_i and g_i denote the predicted probability and the ground truth label for the i^{th} pixel, respectively, and N is the total number of pixels in the image. The Dice loss is particularly effective for addressing class imbalance, as it emphasizes the overlap between the predicted and ground truth segmentation masks.

IV. Experiments

This section begins with an explanation of the dataset and implementation details. Next, we evaluate our results both quantitatively and qualitatively. Finally, we perform ablation studies to further demonstrate the effectiveness of the proposed methods.

A. Dataset and the Implementation Details

Dataset Details: We train and test our method on the Waterscenes dataset [15], which provides per-pixel segmentation masks for seven object classes, as well as a waterline class. Additionally, we include a background class to aid in mask denoising for the Noise Reduction Unit (NRU). Furthermore, we evaluate our method on a subset of the dataset, selecting images based on specific weather conditions to test performance under the most challenging scenarios, such as *foggy*, *strong light exposure*, and *rainy* conditions, as well as instances where the camera sensor is distorted, e.g., by *water drop hit*. The dataset comprises a total of 37,884 training, 10,824 validation, and 1,596 test images.

Implementation Details: We use the Segformer-B0 architecture for both the encoder and decoder and the Stable Diffusion Inpainting models from the Huggingface platform [41]. During training, if the number of radar points exceeds 1000, we randomly sample 1000 points; otherwise, we apply zero-padding to the input. The images are downscaled to 320x320. For efficient and stable training, we utilize the PyTorch Lightning library [42]. Our models are trained on an Nvidia® Tesla A30 GPU with a batch size of 32.

To optimize computational efficiency, we first train the stage 1 model and use the NRU to generate masks. Stage 3 is then trained separately using these extracted masks. We employ the AdamW optimizer [43] with an initial learning rate of 5×10^{-4}, which is linearly decayed to a minimum learning rate of 1×10^{-6}. In line with the training strategy from Achelous [32], no additional data augmentation techniques are applied during training.

B. Quantitative Results

Method	**mIoU (%)**	Params (M)
Achelous-EV-GDF-PN-S2 [32]	-	8.28
Segformer(camera-only)	75.47	3.7
Segformer Fusion	76.62	4.3
Segformer Fusion + Inpainting	**78.10**	4.3 + 9.66(MobileSAM) + 1290(Stable Diffusion)

TABLE I: Comparison of Segmentation Methods with mIoU under Adverse Weather Conditions.

To evaluate our method, we use the mean Intersection over Union (mIoU) metric [44], a widely recognized measure for assessing segmentation quality. We first evaluate the models on the adverse weather sub-testset, with results presented in Table I. There is a clear improvement in mIoU from the camera-only model to the fusion model, and further to the fusion model with inpainting. This trend indicates that incorporating multiple modalities enhances segmentation accuracy by providing richer information about the scene, especially during the hard sample segmentation scenarios. Additionally, inpainting contributes to improved results by filling in missing or obstructed regions, especially in cases where visual data is compromised due to adverse weather conditions. It is important to note that the Achelous models did not include evaluation results on the adverse weather subset. Therefore, our comparison with the best-performing model from [32] is limited to model parameter counts only.

Additionally, we analyze the models on the total test set, which contains 5120 images, against the models provided in the Achelous [32] as shown in the Table II. Our fusion-based methods for semantic segmentation of targets achieve 8.11% improvement compared to the best Achelous model. However, due to the inherent noise in radar data, some point segmentations are incorrectly predicted, for example, due to water reflections. This results in a slight performance drop in mIoU$_d$ when compared to the Achelous models.

TABLE II: Comparison of Segmentation Methods with mIoU on the Total Testset.

Method	*mIoU$_t$ (%)	*mIoU$_d$ (%)
Achelous-MV-GDF-PN-S0† [32]	70.60	99.5
Achelous-MV-GDF-PN-S1† [32]	73.20	99.5
Achelous-EV-GDF-PN-S2† [32]	74.10	99.5
Segformer(camera-only)	81.12	98.64
Segformer Fusion	82.39	98.74
Segformer Fusion + Inpainting	82.21	98.75

*mIoU$_t$: mIoU of targets, *mIoU$_d$: mIoU of drivable area
† These results come from the original paper.

Here, we use 'Segformer Fusion' and 'Segformer Fusion + Inpainting' to denote the models from the first and third stages, respectively.

C. Qualitative Results

Inpainted Images: The generated inpainted images exhibit a pixel distribution closely matching the original images, with the primary differences occurring in the inpainted areas. These regions, however, do not seamlessly integrate the original texture of the masked areas, which can result in a noticeable difference in appearance. Additionally, the inpainting process introduces a degree of noise, attributed to the inherent variability of the sampling process. Despite these challenges, such as the distinct textures and noise, the system demonstrates a remarkable ability to learn and fuse features from both the original and inpainted images effectively. This suggests that the model is robust in aligning and integrating visual information, enabling it to synthesize cohesive images even when texture inconsistencies are present. To demonstrate qualitatively that our method is effective in adverse conditions, some predicted images can be found in the second column of Figure 3.

Segmentation: Building on previous analyses, our model demonstrates notable performance improvements in semantic segmentation tasks, as illustrated in Figure 3. We present qualitative comparisons between the baseline prediction, the fusion prediction from the first stage, and the final prediction incorporating inpainted images. The results show that the proposed CaRaFFusion model detects finer details, producing more accurate masks. Despite challenges such as texture

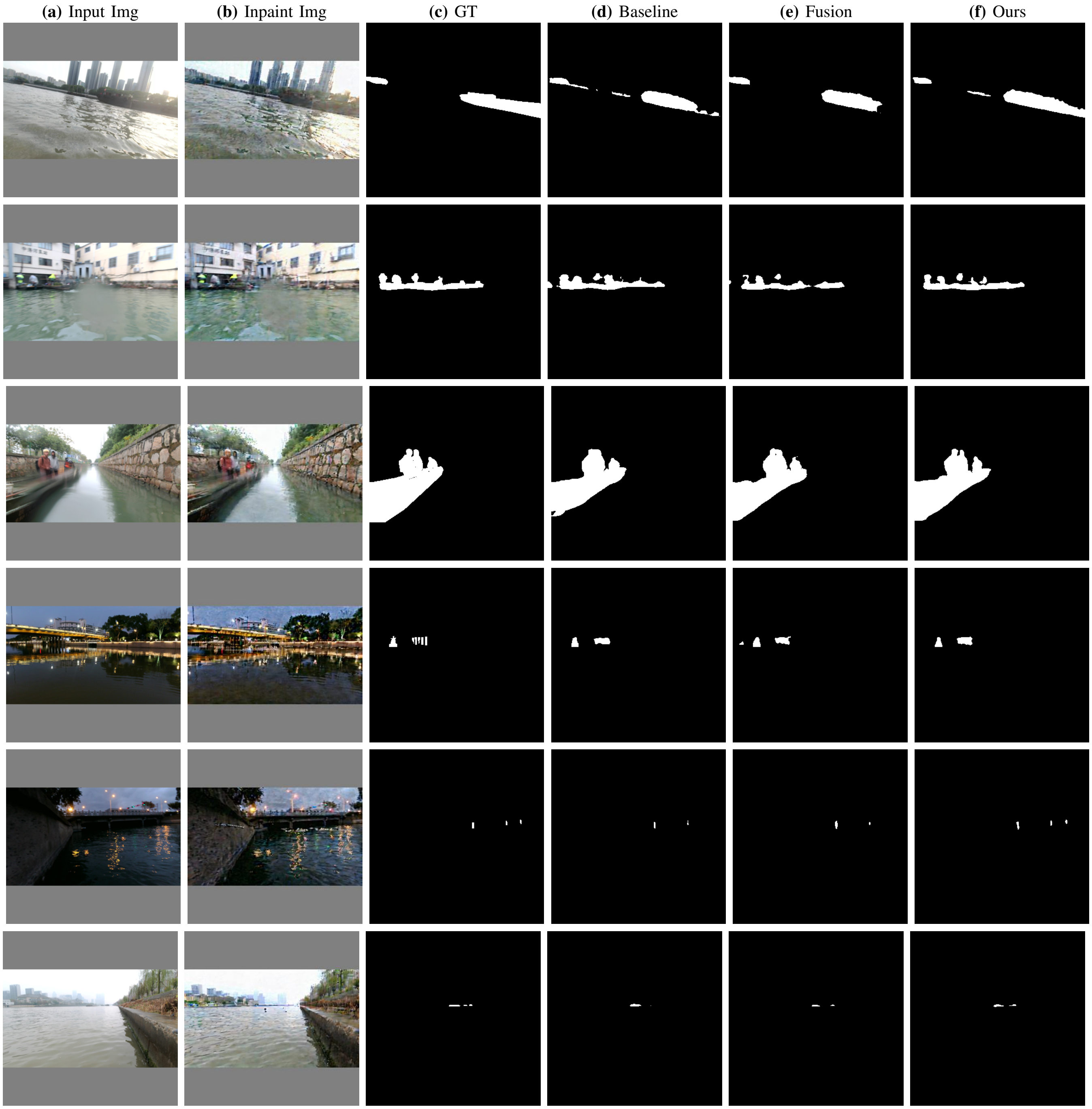


Fig. 3: Qualitative results show that our inpainting technique is likely addressing missing or occluded regions in the data. It helps to fill in parts of the objects or scenes that might otherwise go undetected, thereby boosting the model's ability to achieve higher segmentation accuracy. Columns (a), (b), and (c) visualize the original image, the inpainted image, and the Ground Truth (GT) mask of the given image. Columns (d), (e), and (f) illustrate the segmentation from the image-only baseline, the fusion model from the first stage, and the final model from the third stage, respectively.

inconsistencies and noise in inpainted regions, the model effectively learns to integrate features from both original and inpainted areas. This capacity to reconcile pixel distribution differences and fuse diverse textures enables the generation of more precise segmentations. Consequently, these improvements yield a significant boost over the baseline, highlighting the model's effectiveness in enhancing semantic segmentation performance.

D. Ablation Studies

This section first evaluates the sampling strategy for the radar point input. Then, we also evaluate the fusion approach at the third stage.

Input Point Sampling: Given that the number of radar points varies between frames, it is necessary to sample a subset of radar points during training. As shown in Table III, increasing the number of sampled radar points improves the effective utilization of radar point cloud data, leading to enhanced performance.

Number of sampled points during training	mIoU(%)
100	75.10
200	75.39
1000	76.72

TABLE III: Radar Points Sampling Comparison.

Different Fusion Methods: We evaluate the effectiveness of different image fusion methods in the third stage. Different fusion methods were implemented to assess their impact on segmentation performance. Table IV summarizes the performance of each fusion method.

Fusion Method	mIoU(%)
Addition	77.13
Gated Fusion [8]	77.43
Concatenation	**78.10**

TABLE IV: Results of Different Fusion Methods in Stage 3.

Importance of Inpainting Fusion: To demonstrate the effectiveness of image fusion in the third stage, we trained a model without inpainting fusion by directly using the inpainted images as input and evaluated it on the same test set. Table V shows the performance with and without fusion in the third stage. This indicates the importance of the fusion of original and inpainted images.

	mIoU(%)
Fusion	78.10
No Fusion	65.13

TABLE V: Results with and without inpainting fusion in Stage 3.

V. Discussion and Limitations

CaRaffusion is designed to operate with high-precision 3D radar point clouds that include height values. The rationale behind this choice is that accurate 3D positions are essential for correctly deriving 2D pixel coordinates when projecting 3D points onto the 2D image plane. This, in turn, enables the MobileSAM module to effectively utilize this information to generate inpainted images. Consequently, WaterScenes [15] is used for experiments instead of datasets like nuScenes [45], where radar point clouds lack height information.

Integrating an image inpainting framework into our 2D semantic segmentation architecture creates an efficiency bottleneck. Advancements in diffusion model inference could resolve real-time constraints and we leave this as future work. In our work, we rather focused on a general concept, how to incorporate a diffusion model into a camera-radar fusion pipeline.

Radar data, while invaluable in adverse weather conditions, often contain significant noise that can degrade perception performance. Sources of noise include reflections from water surfaces, environmental clutter, and overlapping object signals. Future work could explore more advanced noise reduction techniques to generate more noise-free pseudo-masks for a better inpainting framework.

VI. Conclusion

In this work, we presented a novel three-stage framework, CaRaFFusion, that effectively integrates radar and camera data for robust segmentation by integrating a generative image inpainting model, particularly suited for adverse weather conditions where traditional segmentation methods may struggle. By combining radar and camera data, our model leverages the unique strengths of each modality to compensate for visual limitations and environmental noise, which are prevalent challenges in outdoor settings. The framework approximately recovers essential shape features by fusing radar and image data and implementing the inpainting pipeline to enhance 2D pixel-wise semantic segmentation performance in adverse weather conditions. While our results show promise, future work should explore optimizing the framework's efficiency and reducing GPU capacity requirements to enhance scalability and make it more feasible for real-time applications. Addressing these performance concerns will be crucial for deploying this model in resource-constrained environments.

VII. Acknowledgement

Research leading to these results has received funding from the EU ECSEL Joint Undertaking under grant agreement n° 101007326 (project AI4CSM) and from the partner national funding authorities the German Ministry of Education and Research (BMBF).

References

[1] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "Segformer: Simple and efficient design for semantic segmentation with transformers," *Advances in neural information processing systems*, vol. 34, pp. 12 077–12 090, 2021.

[2] A. Milioto and C. Stachniss, "Bonnet: An open-source training and deployment framework for semantic segmentation in robotics using cnns," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 7094–7100.

[3] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*. Springer, 2015, pp. 234–241.

[4] S. Yao, R. Guan, X. Huang, Z. Li, X. Sha, Y. Yue, E. G. Lim, H. Seo, K. L. Man, X. Zhu *et al.*, "Radar-camera fusion for object detection and semantic segmentation in autonomous driving: A comprehensive review," *IEEE Transactions on Intelligent Vehicles*, 2023.

[5] A. Pfeuffer and K. Dietmayer, "Robust semantic segmentation in adverse weather conditions by means of sensor data fusion," in *2019 22th International Conference on Information Fusion (FUSION)*. IEEE, 2019, pp. 1–8.

[6] Y. Kim, J. Shin, S. Kim, I.-J. Lee, J. W. Choi, and D. Kum, "Crn: Camera radar net for accurate, robust, efficient 3d perception," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2023, pp. 17 615–17 626.

[7] Z. Lin, Z. Liu, Z. Xia, X. Wang, Y. Wang, S. Qi, Y. Dong, N. Dong, L. Zhang, and C. Zhu, "Rcbevdet: Radar-camera fusion in bird's eye view for 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 14 928–14 937.

[8] H. Sun, H. Feng, J. Ott, L. Servadei, and R. Wille, "Cafnet: A confidence-driven framework for radar camera depth estimation," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2024, pp. 2734–2740.

[9] H. Sun, Z. Wang, H. Feng, J. Ott, L. Servadei, and R. Wille, "Get-up: Geometric-aware depth estimation with radar points upsampling," in *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2025, pp. 1850–1860.

[10] K. Bansal, K. Rungta, and D. Bharadia, “Radsegnet: A reliable approach to radar camera fusion,” *arXiv preprint arXiv:2208.03849*, 2022.

[11] J. Schramm, N. Vödisch, K. Petek, B. R. Kiran, S. Yogamani, W. Burgard, and A. Valada, “Bevcar: Camera-radar fusion for bev map and object segmentation,” *arXiv preprint arXiv:2403.11761*, 2024.

[12] A. Vaswani, “Attention is all you need,” *Advances in Neural Information Processing Systems*, 2017.

[13] C. Zhang, D. Han, Y. Qiao, J. U. Kim, S.-H. Bae, S. Lee, and C. S. Hong, “Faster segment anything: Towards lightweight sam for mobile applications,” *arXiv preprint arXiv:2306.14289*, 2023.

[14] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “High-resolution image synthesis with latent diffusion models,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 10 684–10 695.

[15] S. Yao, R. Guan, Z. Wu, Y. Ni, Z. Huang, R. W. Liu, Y. Yue, W. Ding, E. G. Lim, H. Seo *et al.*, “Waterscenes: A multi-task 4d radar-camera fusion dataset and benchmarks for autonomous driving on water surfaces,” *IEEE Transactions on Intelligent Transportation Systems*, 2024.

[16] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[17] V. Badrinarayanan, A. Kendall, and R. Cipolla, “Segnet: A deep convolutional encoder-decoder architecture for image segmentation,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[18] X. Li, H. Zhao, L. Han, Y. Tong, S. Tan, and K. Yang, “Gated fully fusion for semantic segmentation,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 07, 2020, pp. 11 418–11 425.

[19] J. Liu, F. Zhang, Z. Zhou, and J. Wang, “Bfmnet: Bilateral feature fusion network with multi-scale context aggregation for real-time semantic segmentation,” *Neurocomputing*, vol. 521, pp. 27–40, 2023.

[20] H. Pan, Y. Hong, W. Sun, and Y. Jia, “Deep dual-resolution networks for real-time and accurate semantic segmentation of traffic scenes,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 3, pp. 3448–3460, 2022.

[21] R. Strudel, R. Garcia, I. Laptev, and C. Schmid, “Segmenter: Transformer for semantic segmentation,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 7262–7272.

[22] X. Yang, W. Yan, Y. Yuan, M. B. Mi, and R. T. Tan, “Semantic segmentation in multiple adverse weather conditions with domain knowledge retention,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 7, 2024, pp. 6558–6566.

[23] A. Kerim, F. Chamone, W. Ramos, L. S. Marcolino, E. R. Nascimento, and R. Jiang, “Semantic segmentation under adverse conditions: a weather and nighttime-aware synthetic data-based approach,” *arXiv preprint arXiv:2210.05626*, 2022.

[24] A. Ouaknine, A. Newson, P. Pérez, F. Tupin, and J. Rebut, “Multi-view radar semantic segmentation,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 15 671–15 680.

[25] Y. Dalbah, J. Lahoud, and H. Cholakkal, “Transradar: Adaptive-directional transformer for real-time multi-view radar semantic segmentation,” in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 353–362.

[26] M. Zeller, J. Behley, M. Heidingsfeld, and C. Stachniss, “Gaussian radar transformer for semantic segmentation in noisy radar data,” *IEEE Robotics and Automation Letters*, vol. 8, no. 1, pp. 344–351, 2022.

[27] G. Lu, Z. He, S. Zhang, Y. Huang, Y. Zhong, Z. Li, and Y. Han, “A novel method for improving point cloud accuracy in automotive radar object recognition,” *IEEE Access*, vol. 11, pp. 78 538–78 548, 2023.

[28] M. Zeller, V. S. Sandhu, B. Mersch, J. Behley, M. Heidingsfeld, and C. Stachniss, “Radar instance transformer: Reliable moving instance segmentation in sparse radar point clouds,” *IEEE Transactions on Robotics*, 2023.

[29] V. Lekic and Z. Babic, “Automotive radar and camera fusion using generative adversarial networks,” *Computer Vision and Image Understanding*, vol. 184, pp. 1–8, 2019.

[30] V. John, M. Nithilan, S. Mita, H. Tehrani, R. Sudheesh, and P. Lalu, “So-net: Joint semantic segmentation and obstacle detection using deep fusion of monocular camera and radar,” in *Image and Video Technology: PSIVT 2019 International Workshops, Sydney, NSW, Australia, November 18–22, 2019, Revised Selected Papers 9*. Springer, 2020, pp. 138–148.

[31] H. Sun, H. Feng, G. Stettinger, L. Servadei, and R. Wille, “Multi-task cross-modality attention-fusion for 2d object detection,” in *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2023, pp. 3619–3626.

[32] R. Guan, S. Yao, X. Zhu, K. L. Man, E. G. Lim, J. Smith, Y. Yue, and Y. Yue, “Achelous: A fast unified water-surface panoptic perception framework based on fusion of monocular camera and 4d mmwave radar,” in *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2023, pp. 182–188.

[33] C. Liang, W. Wang, J. Miao, and Y. Yang, “Gmmseg: Gaussian mixture based generative semantic segmentation models,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 31 360–31 375, 2022.

[34] K. Liu, Z. Ye, H. Guo, D. Cao, L. Chen, and F.-Y. Wang, “Fiss gan: A generative adversarial network for foggy image semantic segmentation,” *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 8, pp. 1428–1439, 2021.

[35] L. Liao, J. Xiao, Z. Wang, C.-W. Lin, and S. Satoh, “Guidance and evaluation: Semantic-aware image inpainting for mixed scenes,” in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16*. Springer, 2020, pp. 683–700.

[36] T. Yu, R. Feng, R. Feng, J. Liu, X. Jin, W. Zeng, and Z. Chen, “Inpaint anything: Segment anything meets image inpainting,” *arXiv preprint arXiv:2304.06790*, 2023.

[37] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, “Segment anything,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 4015–4026.

[38] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, “Pointnet: Deep learning on point sets for 3d classification and segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 652–660.

[39] T.-Y. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, pp. 318–327, 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:206771220

[40] C. H. Sudre, W. Li, T. K. M. Vercauteren, S. Ourselin, and M. J. Cardoso, “Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations,” *Deep learning in medical image analysis and multimodal learning for clinical decision support : Third International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, held in conjunction with MICCAI 2017 Quebec City, QC,...*, vol. 2017, pp. 240–248, 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:21957663

[41] T. Wolf, “Huggingface’s transformers: State-of-the-art natural language processing,” *arXiv preprint arXiv:1910.03771*, 2019.

[42] W. Falcon and T. P. L. team, “Pytorch lightning,” *GitHub. Note: https://github.com/PyTorchLightning/pytorch-lightning*, 2019.

[43] I. Loshchilov, “Decoupled weight decay regularization,” *arXiv preprint arXiv:1711.05101*, 2017.

[44] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, “The pascal visual object classes (voc) challenge,” *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, Jun. 2010.

[45] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, “nuscenes: A multimodal dataset for autonomous driving,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 621–11 631.