Robust Few-Shot Semantic Segmentation for Blurred and Occluded Objects in Construction Environments

Maghsood Salimi[1], Mohammad Loni[2], Antonio Cicchetti[1], Marjan Sirjani[1]
[1]School of Innovation, Design and Engineering, Mälardalen University, Västerås, Sweden
[2]RNP Lab of the AASS Research Centre, Örebro University, Örebro, Sweden

***Abstract*—The increasing demand for autonomous machines in construction environments necessitates the development of robust object detection algorithms that can perform effectively across various weather and environmental conditions. However, challenging conditions at construction sites, such as mud splashes and vibrations, can degrade object detection performance by causing sensor occlusions and image blurriness. Traditional adversarial training methods, which enhance model robustness by using perturbed data, are limited in construction environments due to the scarcity of diverse real-world adversarial data and the dynamic nature of construction environments. To overcome these challenges, this paper explores utilizing few-shot learning (FSL) to improve the generalization performance and robustness of object detection models. FSL enables models to adapt quickly using minimal data, reducing the need for large datasets. In addition, we identify an often-overlooked issue: the hyperparameters used in FSL training are typically not optimized for this unique paradigm. To address this, we combine FSL with hyperparameter optimization to enhance model performance across multiple small-scale datasets. Experimental results demonstrate that our approach improves model performance on the ConstScene dataset over the default training paradigm. The code for this study is available at here.**

***Index Terms*—Adversarial Attacks, Semantic Segmentation, Meta-Learning, Hyperparameter Optimization, Construction**

I. Introduction

The market for autonomous construction machines is projected to grow at a compound annual growth rate of 7.80%, increasing from USD 8.73 billion in 2023 to USD 15.92 billion by 2032 [1]. This growth is driven by the diverse advantages of autonomous construction machinery, such as reliable performance, environmental sustainability, enhanced safety, and cost efficiency. Object detection is a widely used computer vision task that plays an essential role in the safety of autonomous construction sites [2]. Nevertheless, environmental conditions on construction sites, such as mud splashes and dirt, can occlude sensor lenses, making it difficult for the object detection models to perceive accurately [3], [4]. Furthermore, construction machines often operate on uneven and rough terrains, leading to substantial vibrations that can result in blurry camera images [5]. As a result, leveraging object detection models in construction sites demands additional considerations for robust and reliable predictions.

Adversarial training methods [6], [7] are popular for defending against adversarial examples and noisy inputs. These approaches involve using a large amount of intentionally perturbed data to train the model to resist such manipulations. The perturbed image data for adversarial training can be obtained either from (i) the real environment, or (ii) using generative models with lower costs and fewer privacy limits such as generative adversarial networks (GANs) [8] or diffusion models [9]. However, each of these approaches comes with its own challenges.

Collecting data from construction sites is both time-consuming and expensive. Even with making significant effort, the data still lacks diversity, as construction machines typically perform repetitive tasks. Consequently, we suffer from small-scale datasets when it comes to object detection in construction sites [10], [11]. On the other hand, generative models depend on strong assumptions about how data is distributed [12], which is often ineffective in real-world situations. Plus, construction sites are constantly changing, with factors such as moving gravel piles, new equipment, and changing machine layouts, creating unusual data that generative models struggle to handle. Lastly, adversarial training methods can sometimes overfit, focusing too much on specific examples [6].

To improve the generalization performance and robustness of object detection in highly dynamic and complex construction sites, a potential solution is to utilize multiple small-scale datasets collected from various construction sites. Each site presents unique adverse conditions and focuses on certain tasks. Few-shot learning (FSL) has recently gained attention since it allows models to generalize well with only a few training examples [13], [14]. Instead of requiring large datasets to learn patterns and make accurate predictions, few-shot learning is designed to work in scenarios where we have access to small-scale datasets. The objective is to teach the model to learn effectively from minimal data by leveraging prior knowledge from related datasets.

In this paper, we leverage the idea of FSL for improving the generalization and robustness of object detection models in construction environments. FSL has the capacity to facilitate the rapid adaptation of models to new data while reducing training time, making FSL particularly well-suited for a wide range of construction applications. Finally, unlike synthetic data generation methods, FSL is independent of prior assumptions about the data distribution.

While significant blame for the poor performance of object detection in construction sites goes to the lack of large-

scale datasets, which could be potentially resolved by FSL, we identified an overlooked factor: hyperparameters of the model trained with the FSL paradigm are usually kept the same as the regular training paradigm. A natural step, thus, is to propose leveraging a hyperparameter optimization (HPO) regime combined with FSL to train a meta model that can effectively learn from multiple small-scale datasets.

Contributions. Our paper's contributions are:

1) We demonstrate that utilizing the few-shot learning approach enhances the robustness of object detection models in handling blurriness and image occlusions.
2) We apply data augmentation techniques for motion blur and dirty lens perturbations to simulate real-world adverse conditions.
3) We propose a novel approach to tweak FSL hyperparameters, moving away from default training protocols of non-robust counterparts.

Our experiments on the ConstScene dataset [3] demonstrated accuracy improvements of at least 7.9% for motion blur and 14.6% for dirty lens perturbations in a 1-shot learning scenario, averaged across all folds, compared to the state-of-the-art few-shot segmentation models.

II. Preliminaries

A. Few-Shot Learning

Few-shot learning (FSL) is a machine learning paradigm that aims to develop models capable of solving new tasks using only a small number of training data, mimicking human ability to generalize from limited examples. Pioneering work by Fei-Fei et al. [15] laid the groundwork for one-shot learning with their innovative approach of harnessing knowledge from previously learned categories. A Bayesian framework was introduced that modeled represented object categories as probabilistic models, showcasing the efficacy of knowledge transfer across diverse categories in enhancing one-shot learning outcomes. Finn et al. [14] made a significant contribution to FSL by introducing Model-Agnostic Meta-Learning (MAML). This approach learns an initial model state that can be rapidly adapted to new tasks with a few gradient steps, showing remarkable performance on diverse benchmarks. Investigating the synergy between generative models and FSL, Antoniou et al. [16] introduced Data Augmentation Generative Adversarial Networks (DAGANs), which employ GANs to create synthetic training samples for few-shot scenarios. This approach aims to boost model robustness and precision by expanding the limited training dataset. Additionally, Oreshkin et al. [17] proposed TADAM, a method that dynamically modifies the metric space based on the specific task. By tailoring this adaptive strategy to the characteristics of each task, TADAM has significantly shown to enhance performance in FSL scenarios. Although FSL, primarily focused on classification tasks, has been extensively studied, its adaptation to segmentation tasks remains challenging due to the dense prediction problem. [13], [18]–[20] tries to address few-shot semantic segmentation methods. Despite their success, there has been no significant effort to optimize the hyperparameters of the FSL paradigm.

B. Hyper-parameter Optimization

Hyperparameter optimization (HPO) [21], [22] is critical for enhancing model performance, as it involves selecting the best set of hyperparameters to guide the training process. Unlike model parameters, which are learned during training, hyperparameters are typically fixed and control various aspects of the model, such as the learning rate and regularization strength. Consider a model $f(x;\theta)$, where θ represents the model parameters and x is the input data. The objective of HPO is to find the set of hyperparameters λ that minimizes the validation loss $\mathcal{L}(f(x;\theta(\lambda)),y)$, where y is the ground truth label. This can be expressed as: $\lambda^* = \arg\min_\lambda \mathcal{L}(f(x;\theta(\lambda)),y)$, where λ^* represents the optimal hyperparameters, and $\mathcal{L}$ is the loss function used to evaluate the model's performance.

C. Motion Blur Perturbation

Motion blur, a common artifact in photographs, occurs when the camera or the object shifts during the exposure period. This prevalent image distortion has garnered attention across various domains such as computer vision and image processing. Operating on uneven construction terrains with medium- to large-sized gravel is the primary cause of blurriness in recorded images (Figure 1). Mathematically, motion blur can be described as a convolution of the original image with a kernel that represents the motion. Let $I(x,y)$ be the original image, $I_{blurred}(x,y)$ represent the blurred image, and the motion blur kernel as $h(x,y)$, then the motion blur can be expressed as: $I_{blurred}(x,y) = I(x,y) * h(x,y)$, where $*$ denotes the convolution operation. For instance, kernel for linear motion blur can be formulated as:

$$h(x,y) = \begin{cases} \frac{1}{L} & \text{if } \sqrt{x^2+y^2} \leq \frac{L}{2} \text{ and } y = x \cdot \tan(\theta) \\ 0 & \text{otherwise} \end{cases}$$

In this context, L denotes the blur length, and θ represents the angle of the motion blur relative to the horizontal axis.

Fig. 1: Illustrative examples of linear motion blur with varying kernel sizes: (a) $L = 3$, (b) $L = 5$, (c) $L = 7$, and (d) $L = 11$.

D. Dirty Lens Perturbation

The dirty lens effect is a photographic technique that emulates the visual distortions that occur when a camera lens is contaminated with dirt, smudges, or other imperfections (Figure 2). This effect can impart a sense of realism and imperfection to images, making them appear more artistic and

(a) **(b)**

Fig. 2: Illustrative examples of (a) a dirty camera lens covered by e.g., snow, and (b) a recorded image with dirt-induced occlusion at a construction quarry.

evocative. It is particularly popular in creative photography and digital art, where artists seek to create an emotional impact through the use of imperfection.

Simulating the dirty lens effect involves blending the original image with a blurred version of itself, modulated by a texture that represents the imperfections of a dirty lens. The texture typically includes an alpha channel, which controls the transparency of the smudges, allowing for blending between the sharp and blurred components of the image. Let $I(x, y)$ be the original image, $I_{blurred}(x, y)$ represent the blurred image, and $M(x, y)$ be the mask derived from the texture's alpha channel. The dirty lens effect can be expressed as:

$$I_{dirty}(x, y) = I(x, y) \cdot (1 - M(x, y)) + I_{blurred}(x, y) \cdot M(x, y)$$

where $I_{dirty}(x, y)$ is the resulting image simulating the dirty lens effect. The blurred image $I_{blurred}(x, y)$ is obtained using a Gaussian blur operation defined as follows:

$$I_{blurred}(x, y) = I(x, y) * h(x, y)$$

where $h(x, y)$ is the Gaussian blur kernel, which can be formulated as: $h(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$. In this context, σ determines the extent of the blur, influencing the amount of distortion applied to the original image. The mask $M(x, y)$ derived from the texture's alpha channel can be defined as: $M(x, y) = \frac{\alpha(x,y)}{255}$, where $\alpha(x, y)$ is the alpha channel value from the texture image. This formulation describes how the original image and the blurred image are blended based on the mask, effectively simulating the appearance of a photograph taken through a dirty lens.

III. Related Work

To the best of our knowledge, our method is the first automated framework that applies the FSL paradigm to improve the robustness of object detection models. Our study also sheds light on the fact that tweaking the hyperparameters plays a crucial role in improving the accuracy of object detection models faced with adversarial inputs. The robustness of object detection models against motion blur and occluded objects has been studied in the past. In this section, we briefly review these methods and compare them with our contributions.

A. Object Detection in Construction Industry

Vision-based sensors are prevalent at construction sites [23], generating vast amounts of image and video data for various purposes such as detecting objects (e.g., workers, materials, and equipment), progress tracking, productivity measurement, and safety monitoring. While traditional computer vision methods have been employed for this purpose [24], [25], their accuracy has been limited by manual feature extraction processes and insufficient training data. To overcome these challenges, an improved Faster R-CNN approach [26] was developed by [27], which significantly enhances real-time detection accuracy. [28] introduced a sophisticated three-stage framework for tracking multiple individuals concurrently in construction sites. Their approach initiates with a detection phase, leveraging both 2D and 3D Mask-RCNN models to locate human figures and determine their poses within images. The researchers compared the two approaches based on detection and tracking capabilities.

Construction object detection models face challenges in adverse conditions due to the limited training data and lack of robustness. To address the challenges of manually interpreting data, researchers [10] are exploring soft computing methods. These methods utilize convolutional models, which are emerging as a promising approach for fast construction object detection. The authors of [29] developed the UIA-YOLOv5 model as a technique to enhance performance in adverse environments such as low light, fog, and rain. At construction sites, moving obstacles often obstruct views, which compromises the quality of captured images. The research proposed by [30] adapts a U-Net-based deep learning [31] model to remove these occlusions and restore the missing background, improving image analysis at construction sites. [3] addressed the challenges of adverse weather and environmental conditions at mining construction sites by leveraging adversarial training to improve the performance of object detection models.

Summary. Although previous studies have been successful, they have largely overlooked adverse environmental issues such as image blurring caused by vibrations and dirty lenses, commonly resulting from mud or water splashes. Additionally, previous studies heavily depend on the availability of adversarial examples, which are not only scarce but also time-consuming to gather. Furthermore, earlier research has overlooked the critical role that hyperparameter tweaking plays in enhancing the robustness of object detection models.

B. Robustness Against Motion Blur

Image de-blurring methods boost computer vision model resilience by sharpening images and refining feature extraction, especially in challenging, high-motion scenarios. A study by [32] proposed a piecewise-linear model to accurately estimate the blur kernel, providing a balanced trade-off, offering robustness to noise while maintaining flexibility for different blur types. Sayed et al. [33] studied five classes of remedies, each of which addresses specific causes of the performance gap between object detection in sharp and blurred images. [34] utilized class-specific augmentation techniques to randomly apply motion blur to input images, thereby enhancing model robustness to motion blur artifacts. Recent research has explored methods to improve the resilience of semantic

segmentation models against motion blur. The study discovered that custom label generation outperforms methods such as de-blurring, multi-scale texture analysis, out-of-distribution testing, and conditioning based on blur type.

C. Robustness Against Dirty Lens

Dirty lenses frequently compromise image quality in practical imaging applications, posing a significant challenge for various vision tasks. Eigen et al. [35] proposed the use of convolutional neural networks (CNNs) to reconstruct clean images from image data degraded by dirt, marking an early effort in this domain. Uricar et al. [36] employed GAN-based data augmentation to generate diverse lens soiling patterns, improving the robustness of autonomous driving systems.

D. Robustness Against Occluded Objects

Despite the remarkable benefits in utilizing CNNs for computer vision tasks, they often struggle with occlusions. DeVries et al. [37] introduced Mask R-CNN to address this weakness by combining object detection and instance segmentation, enabling a more detailed understanding of object boundaries and occluded areas. To enhance model robustness, data augmentation techniques have also been studied. DeVries and Taylor [37] proposed cutout, a straightforward yet effective method of randomly obscuring square areas of training data, which can simulate occlusions and improve generalization. Recent studies have investigated attention mechanisms to enhance robustness against occlusions. Woo et al. [38] introduced the Convolutional Block Attention Module, which improves the network's ability to prioritize important features, thereby reducing the impact of occlusions.

IV. Method

A. Model Architecture and Training

In this section, we present the architecture of the semantic segmentation model and the training procedure. Inspired by [18], the proposed method improves the feature embedding of few-shot segmentation by resolving feature undermining and prototype bias issues. To address feature undermining, a mining branch is utilized that identifies latent novel classes from the background using transferable sub-clusters and pseudo-labeling. Prototype bias is mitigated through a rectification technique that stabilizes prototypes for both the foreground and background categories.

The backbone network, based on ResNet-50/101 (Figure 3), extracts features from support and query images. Episodic training generates support prototypes using masked average pooling (MAP), which computes the prototype p^c for class c in image i as:

$$p_i^c = \frac{\sum_{x,y} F_i(x,y) \cdot \mathbb{1}[M_i(x,y) = c]}{\sum_{x,y} \mathbb{1}[M_i(x,y) = c]},$$

where $F_i(x,y)$ is the feature vector at pixel (x,y), and $M_i(x,y)$ is the mask. These prototypes are compared with query features for classification using cosine similarity [39].

To enhance feature embeddings, the auxiliary mining branch annotates pseudo-labels by classifying each feature $F(x,y)$ based on its nearest neighbor among $K+1$ representative prototypes $\{p_k\}_{k=1}^{K+1}$: $M_p(x,y) = \arg\max_k \cos(F(x,y), p_k)$, where $M_p(x,y)$ is the pseudo-label and $\cos(\cdot,\cdot)$ is the cosine similarity function. The method incorporates prototype rectification for stability. The background prototype is refined using a global background prototype $p^{\text{bg}}_{\text{global}}$, updated during training as: $p^{\text{bg}}_{\text{global}} \leftarrow m p^{\text{bg}}_{\text{global}} + (1-m) p^{\text{bg}}_{\text{current}}$, where m is the momentum coefficient. During inference, the final background prototype $p^{\text{bg}}_{\text{final}}$ combines global and current prototypes: $p^{\text{bg}}_{\text{final}} = w p^{\text{bg}}_{\text{global}} + (1-w) p^{\text{bg}}_{\text{current}}$.

Foreground prototypes are refined during inference using region-level embeddings from the most relevant regions in similar images. The method also leverages unlabeled data, enhancing feature discrimination and enabling better generalization to unseen classes.

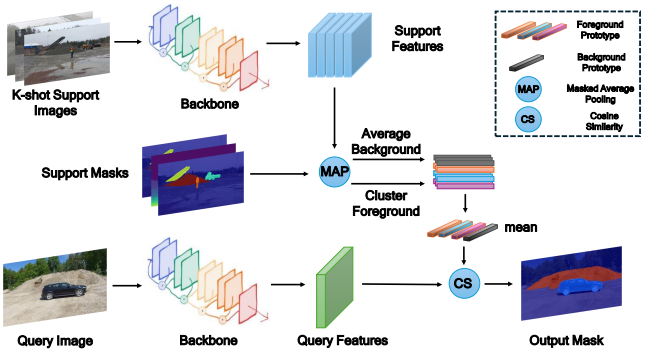

Fig. 3: The overview of the architecture of the few-show semantic segmentation model.

B. Hyperparameter Optimization of Few-shot Learning

The goal of our proposed method is to find an optimal hyperparameter configuration for a few-shot segmentation model, with a particular emphasis on meta-learning to enhance robustness and adaptability in challenging environments. Given the HPO setup described in Section V, we explain how to formulate the FSL problem and what is needed to solve it.

Let the task $\mathcal{T}_i = \{\mathcal{D}^{(i)}_{\text{train}}, \mathcal{D}^{(i)}_{\text{test}}\}$ consist of a small training set $\mathcal{D}^{(i)}_{\text{train}} = \{(x_j, y_j)\}_{j=1}^{N_{\text{train}}}$, where N_{train} is the number of training examples (shots), and a test set $\mathcal{D}^{(i)}_{\text{test}}$. The objective of FSL is to minimize the loss on the test set $\mathcal{D}^{(i)}_{\text{test}}$, after learning from $\mathcal{D}^{(i)}_{\text{train}}$. The overall goal of FSL can be written as:

$$\min_{\theta} \mathbb{E}_{\mathcal{T}\sim p(\mathcal{T})} \left[\mathcal{L}_{\text{test}}(f_\theta(x^{(i)}_{\text{test}}), y^{(i)}_{\text{test}} | \mathcal{D}^{(i)}_{\text{train}}) \right]$$

where:

- $\mathbb{E}_{\mathcal{T}\sim p(\mathcal{T})}$ denotes the expectation over the distribution of tasks $p(\mathcal{T})$.
- $\mathcal{L}_{\text{test}}$ is the loss function computed on the test set.
- $f_\theta(x)$ is the object detection model parameterized by θ, which is learned by observing $\mathcal{D}^{(i)}_{\text{train}}$.

The objective of HPO in FSL is to find the optimal hyperparameters λ^* that minimize the loss across the task distribution $p(\mathcal{T})$. Therefore, the meta-learning objective is formulated as:

$$\min_{\theta} \mathbb{E}_{\mathcal{T}_i \sim p(\mathcal{T})} \left[\mathcal{L}_{\text{test}}(f_{\theta'}(x_{\text{test}}^{(i)}), y_{\text{test}}^{(i)}) \right]$$

where $\theta' = \theta - \alpha \nabla_\theta \mathcal{L}_{\text{train}}(f_\theta(x_{\text{train}}^{(i)}), y_{\text{train}}^{(i)})$ represents the updated model parameters after one or a few gradient steps on the training data $\mathcal{D}_{\text{train}}^{(i)}$. Incorporating HPO into this meta-learning framework further enhances the ability of the model to generalize to new tasks, even under difficult conditions such as motion blur, occlusion, or limited data.

V. Experimental Setup

A. Dataset Preparation

In this study, we use the ConstScene dataset [3], designed to capture the challenges posed by adverse weather and environmental conditions in quarry-focused construction sites. This dataset contains annotated images captured in diverse weather conditions, including sunny, rainy, foggy, and low-light environments. ConstScene consists of images containing car, wheel-loader, crusher, human, pile, and road classes. The ConstScene dataset contains 3,470 images.

In this paper, we augment the dataset by introducing motion blur and dirty lens perturbations, mimicking the actual conditions found on construction sites. To achieve this, for 60% of the images in the original dataset, we randomly selected a perturbation strength from the available options and applied it to the image, resulting in a new perturbed version. We divided the dataset into five distinct folds, ensuring that each fold contained images from five classes. To evaluate model performance, we used four folds for training and reserved the remaining fold for evaluation. In the following, we outline the procedure for applying dirty lens and motion blur perturbations.

Generating dirty lens perturbation. Our method defines four scenarios, including three levels of dirty lens perturbation and one baseline without any perturbations. The perturbations are based on different groups of filters that are randomly applied to the original images. The severity of the perturbations is measured on a per-pixel basis, where 255 represents the strongest possible perturbation and 0 indicates no perturbation. We categorize the average severity into three levels: Perturbation #1, Perturbation #2, and Perturbation #3. Perturbation #1, with an average severity of 14.61%, introduces mild visual obstructions, slightly reducing image clarity. At the medium level, Perturbation #2 increases interference, leading to noticeable degradation with an average severity of 28.93%. The highest level, Perturbation #3, results in significant visual impairment, severely affecting clarity with an average severity of 38.99%. Figure 6.(c) shows an image sample with a dirty lens perturbation at level #2.

Generating motion blur perturbation. To generate the motion blur effect, the process involves creating a smooth streaking effect, utilizing a Gaussian blur kernel. We use kernel sizes of 3, 5, 7, and 11, where larger values result in a higher motion blur effect. Accordingly, we define five perturbation levels, including four levels with motion blur perturbation and one baseline without any perturbation. Figure 6.(d) shows an image with a motion blur perturbation at level #4.

B. Training and Hyperparameter Configuration

Table I specifies the training configuration and the list of hyperparameters. We use ResNet-50 and ResNet-101 [40] as segmentation backbone architectures in all experiments.

TABLE I: Configuration parameters of the training and HPO procedures.

Parameter	Value
Few-shot training episodes	1200
Few-shot test episodes	300
Shots	{1, 10}
Folds	{1, 2, 3, 4, 5}
Optimizer	{SGD, Adam, RMSProp}
Learning rate	[1e-4, 1e-1]
Batch size	{100, 250, 500}
Global training time	1.1 GPU-hours (Tesla M10)

VI. Results

This section presents the experimental results, demonstrating the effectiveness of our proposed method under various perturbation levels, followed by a detailed comparison of its performance with baseline models. In our experiments, "Base" refers to the baseline model's performance with default training protocols without utilizing any robustification method, "Ours" refers to the results of our method under each fold, and "Aug" refers to the results of our approach when incorporating augmented motion blur- and dirty lens-perturbed training data. In the following, we briefly explain the baselines of our experiments:

- **MiningFSS [18]:** MiningFSS is a few-shot segmentation method that enhances segmentation accuracy by mining and leveraging feature relationships based on conventional episodic training on support and query images.
- **PFENet [19]:** PFENet is a few-shot semantic segmentation framework that leverages prior information and feature enrichment to achieve accurate segmentation with limited labeled data.
- **GFS-Seg [13]:** By proposing context-aware prototype learning, GFS-Seg enables simultaneous segmentation of both well-trained base classes and novel classes with limited labeled data.
- **HSNet [20]:** HSNet used multi-level feature correlations, called hypercorrelations, by extracting features from various intermediate convolutional layers of a CNN.

A. Robustness Results Against the Dirty Lens Perturbation

Table II presents the results for 1-shot and 10-shot learning, evaluated across five distinct folds with and without data augmentations, under dirty lens perturbation. Figure 4 shows the average results of perturbation levels across different folds for the 1-shot learning scenario. In the "Without Perturbation"

scenario, our method achieves 7.9% higher accuracy compared to the best baseline (GFS-Seg [13]). This improvement increases to 15.6% when using an augmented training dataset. Furthermore, under the strongest perturbation conditions, our model outperforms the best baseline (GFS-Seg) by providing 12.4% higher accuracy. With augmented training data, our method demonstrates 21.3% higher accuracy compared to the best results (GFS-Seg).

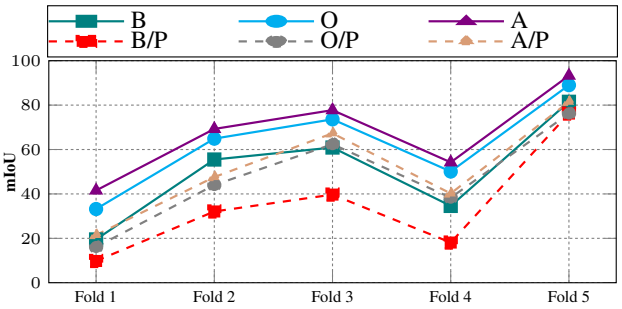

Fig. 4: Results of the 1-shot learning scenario against dirty lens perturbation. 'B' represents the average mIoU of all baseline models, 'O' refers to our model trained on original images, 'A' indicates our model trained on augmented images, and 'P' evaluates the models using the test set under the maximum perturbation level (#3).

B. Robustness Results Against the Motion Blur Perturbation

Table III presents the results for 1-shot and 10-shot learning, evaluated across five distinct folds with and without data augmentations, under the motion blur perturbation. Figure 5 shows the average results of perturbation levels across different folds for the 1-shot learning scenario. In the "Without Perturbation" scenario, our method achieved a 14.6% accuracy improvement over the best baseline (GFS-Seg [13]). With the augmented training dataset, this improvement increased to 23.9%. Under the strongest perturbation condition, our model achieved a 32% accuracy gain, which further rose to 44% with the augmented dataset, demonstrating significant advancements over the baseline approach (GFS-Seg [13]). All in all, the results show that our method significantly improves robustness performance across all folds and perturbation ratios.

C. Qualitative Results

Figure 6 illustrates the qualitative prediction results for various test scenarios against dirty lens and blurry image attacks. As can be seen, FSL+HPO method shows a notable accuracy improvement over the default training procedure.

VII. Conclusion, Limitations, and Broader Impact

Conclusion. Improving the robustness of object detection in the construction industry requires innovative solutions, as the demands of adversarial training approaches are incompatible with the dynamic and challenging conditions of construction sites. Few-shot learning (FSL) is a promising approach that enables models to generalize effectively from minimal data, addressing the limitations of small, repetitive datasets. However, as demonstrated in this paper, using FSL with default training protocols has proven to be inefficient. In this paper, we studied the impact of hyperparameters on training few-shot semantic segmentation models. To address this, we proposed a novel approach that jointly optimizes hyperparameters and model parameters within a few-shot learning paradigm. Our proposed solution offers remarkable improvement over the default training protocols. Crucially, since our method is independent of the model architecture, it can optimize any segmentation network.

Limitations. The authors have determined that this research poses no potential harm to society or the environment, as it does not address any concrete application.

Broader Impact. The proposed method contributes to increasing the safety of operations in harsh construction environments. We will also increase production efficiency by reducing down-times that currently occur when visibility is low in adverse weather conditions. We believe this opens up new avenues of research into methods that can improve the robustness of segmentation models.

References

[1] S. Singh, "Global autonomous construction equipment market overview," https://www.marketresearchfuture.com/reports/autonomous-construction-equipment-market-12648/, 2024, [Online; August 2024].

[2] B. Xiao and S.-C. Kang, "Development of an image data set of construction machines for deep learning object detection," *Journal of Computing in Civil Engineering*, vol. 35, no. 2, p. 05020005, 2021.

[3] M. Salimi, M. Loni, S. Afshar, M. Sirjani, and A. Cicchetti, "Constscene: Dataset and model for advancing robust semantic segmentation in construction environments," *arXiv preprint arXiv:2312.16516*, 2023.

[4] N. Vikas, G. Pahwa, and S. Mohanty, "Camera blockage detection in autonomous driving using deep neural networks," in *2022 Second International Conference on Computer Science, Engineering and Applications (ICCSEA)*. IEEE, 2022, pp. 1–6.

[5] C. Morikawa, M. Kobayashi, M. Satoh, Y. Kuroda, T. Inomata, H. Matsuo, T. Miura, and M. Hilaga, "Image and video processing on mobile devices: a survey," *The Visual Computer*, vol. 37, no. 12, 2021.

[6] T. Bai, J. Luo, J. Zhao, B. Wen, and Q. Wang, "Recent advances in adversarial training for adversarial robustness," *arXiv preprint arXiv:2102.01356*, 2021.

[7] Z. Qian, K. Huang, Q.-F. Wang, and X.-Y. Zhang, "A survey of robust adversarial training in pattern recognition: Fundamental, theory, and methodologies," *Pattern Recognition*, vol. 131, p. 108889, 2022.

[8] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, "Generative adversarial networks: An overview," *IEEE signal processing magazine*, vol. 35, no. 1, pp. 53–65, 2018.

[9] F.-A. Croitoru, V. Hondru, R. T. Ionescu, and M. Shah, "Diffusion models in vision: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 9, pp. 10 850–10 869, 2023.

[10] N. D. Nath and A. H. Behzadan, "Deep convolutional networks for construction object detection under different visual conditions," *Frontiers in Built Environment*, vol. 6, p. 97, 2020.

[11] J. Kim, I. Wang, and J. Yu, "Experimental study on using synthetic images as a portion of training dataset for object recognition in construction site," *Buildings*, vol. 14, no. 5, p. 1454, 2024.

[12] H. Lee, J. Lu, and Y. Tan, "Convergence of score-based generative modeling for general data distributions," in *International Conference on Algorithmic Learning Theory*. PMLR, 2023, pp. 946–985.

[13] Z. Tian, X. Lai, L. Jiang, S. Liu, M. Shu, H. Zhao, and J. Jia, "Generalized few-shot semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11 563–11 572.

[14] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *International conference on machine learning*. PMLR, 2017, pp. 1126–1135.

TABLE II: A comparison of the mIoU results (%) between our proposed method (Ours), our method combined with data augmentation (Aug), and various baseline models under dirty lens perturbation, with results analyzed across different levels of perturbation strength. Darker colors indicate better performance.

Perturbation	Model	1-shot					10-shot				
		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Without Perturb.	MiningFSS [18]	17.1	54.0	60.9	35.9	81.1	26.5	64.3	66.1	40.3	83.5
	PFENet [19]	15.8	55.1	58.4	31.0	77.1	21.4	68.4	69.2	44.8	84.3
	GFS-Seg [13]	25.1	59.7	64.1	37.2	85.0	36.7	68.5	74.4	40.7	86.6
	HSNet [20]	20.0	53.3	60.4	34.4	82.7	28.1	58.4	72.4	41.3	79.6
	Ours	33.2	64.9	73.6	50.1	88.9	38.7	75.0	74.3	53.7	91.1
	Ours+Aug	41.6	69.3	77.7	54.2	93.3	51.3	80.0	78.4	58.1	95.5
Perturb. #1 (14%)	MiningFSS [18]	14.9	52.1	58.1	28.9	76.6	20.3	59.1	61.8	37.9	77.8
	PFENet [19]	12.3	41.7	50.3	25.9	67.7	14.2	46.8	54.8	28.0	80.0
	GFS-Seg [13]	21.2	56.1	59.2	31.9	80.4	23.7	62.9	71.9	37.7	74.6
	HSNet [20]	16.6	38.1	42.8	24.3	76.0	18.3	41.8	60.8	29.6	77.9
	Ours	23.4	55.8	70.4	45.9	82.1	35.7	70.1	71.6	50.5	83.8
	Ours+Aug	34.7	60.7	76.5	49.7	86.5	46.1	73.0	74.6	53.3	88.0
Perturb. #2 (29%)	MiningFSS [18]	12.4	42.2	54.1	27.9	72.2	20.0	52.5	59.3	35.1	74.8
	PFENet [19]	11.1	42.5	45.8	22.8	58.0	13.7	45.0	49.7	26.7	75.3
	GFS-Seg [13]	19.8	44.2	48.6	29.1	61.4	17.0	48.7	66.5	35.2	71.8
	HSNet [20]	15.5	40.5	45.3	27.1	64.9	16.1	40.8	60.4	25.2	72.5
	Ours	19.2	54.1	69.1	41.1	81.1	32.7	64.3	69.2	46.3	81.9
	Ours+Aug	25.8	58.2	75.0	44.8	85.9	40.2	69.0	71.7	49.5	87.4
Perturb. #3 (39%)	MiningFSS [18]	6.1	40.7	50.4	13.7	68.2	12.8	52.6	56.9	21.5	69.8
	PFENet [19]	7.5	29.9	24.4	15.1	28.8	9.3	45.6	26.5	16.9	46.1
	GFS-Seg [13]	13.6	28.0	42.1	24.9	38.9	16.6	40.1	55.9	33.3	59.4
	HSNet [20]	12.2	29.9	42.0	18.9	53.1	17.8	35.6	58.3	24.9	66.5
	Ours	16.1	44.0	62.4	38.3	76.2	29.0	61.8	62.9	41.6	78.7
	Ours+Aug	21.5	47.6	67.3	40.2	81.5	35.9	65.0	65.9	45.7	83.9

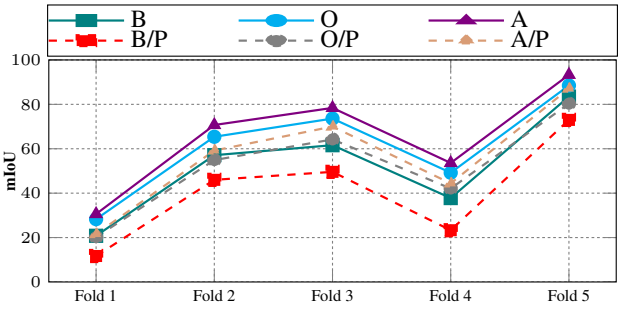

Fig. 5: Results of the 1-shot learning scenario against motion blur perturbation. 'B' represents the average mIoU of all baseline models, 'O' refers to our model trained on original images, 'A' indicates our model trained on augmented images, and 'P' evaluates the models using the test set under the maximum perturbation level (#4).

[15] L. Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE transactions on pattern analysis and machine intelligence*, vol. 28, no. 4, pp. 594–611, 2006.

[16] A. Antoniou, A. Storkey, and H. Edwards, "Data augmentation generative adversarial networks," *arXiv preprint arXiv:1711.04340*, 2017.

[17] B. Oreshkin, P. Rodríguez López, and A. Lacoste, "Tadam: Task dependent adaptive metric for improved few-shot learning," *Advances in neural information processing systems*, vol. 31, 2018.

[18] L. Yang, W. Zhuo, L. Qi, Y. Shi, and Y. Gao, "Mining latent classes for few-shot segmentation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 8721–8730.

[19] Z. Tian, H. Zhao, M. Shu, Z. Yang, R. Li, and J. Jia, "Prior guided feature enrichment network for few-shot segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 2, 2020.

[20] J. Min, D. Kang, and M. Cho, "Hypercorrelation squeeze for few-shot segmentation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 6941–6952.

[21] B. Bischl, M. Binder, M. Lang, T. Pielok, J. Richter, S. Coors, J. Thomas, T. Ullmann, M. Becker, A.-L. Boulesteix *et al.*, "Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 13, no. 2, p. e1484, 2023.

[22] M. Lindauer, K. Eggensperger, M. Feurer, A. Biedenkapp, D. Deng, C. Benjamins, T. Ruhkopf, R. Sass, and F. Hutter, "Smac3: A versatile bayesian optimization package for hyperparameter optimization," *Journal of Machine Learning Research*, vol. 23, no. 54, pp. 1–9, 2022.

[23] S. Xu, J. Wang, W. Shou, T. Ngo, A.-M. Sadick, and X. Wang, "Computer vision techniques in construction: a critical review," *Archives of Computational Methods in Engineering*, vol. 28, 2021.

[24] S. Chi and C. H. Caldas, "Automated object identification using optical video cameras on construction sites," *Computer-Aided Civil and Infrastructure Engineering*, vol. 26, no. 5, pp. 368–380, 2011.

[25] M.-W. Park and I. Brilakis, "Construction worker detection in video frames for initializing vision trackers," *Automation in Construction*, vol. 28, pp. 15–25, 2012.

[26] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, 2015.

[27] W. Fang, L. Ding, B. Zhong, P. E. Love, and H. Luo, "Automated detection of workers and heavy equipment on construction sites: A convolutional neural network approach," *Advanced Engineering Informatics*, vol. 37, pp. 139–149, 2018.

[28] O. Angah and A. Y. Chen, "Tracking multiple construction workers through deep learning and the gradient based method with re-matching based on multi-object tracking accuracy," *Automation in Construction*, vol. 119, p. 103308, 2020.

[29] Y. Ding, M. Zhang, J. Pan, J. Hu, and X. Luo, "Robust object detection in extreme construction conditions," *Automation in Construction*, vol. 165, p. 105487, 2024.

[30] O. Angah and A. Y. Chen, "Removal of occluding construction workers in job site image data using u-net based context encoders," *Automation in Construction*, vol. 119, p. 103332, 2020.

[31] 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.

[32] S. Oh and G. Kim, "Robust estimation of motion blur kernel using a piecewise-linear model," *IEEE transactions on image processing*, vol. 23, no. 3, pp. 1394–1407, 2014.

[33] M. Sayed and G. Brostow, "Improved handling of motion blur in online object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 1706–1716.

[34] A. Rajagopalan *et al.*, "Improving robustness of semantic segmentation

TABLE III: A comparison of the mIoU results (%) between our proposed method (Ours), our method combined with data augmentation (Aug), and various baseline models under the motion blur perturbation, with results analyzed across different kernel sizes. Darker colors indicate better results.

Perturbation	Model	1-shot					10-shot				
		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Without Perturb.	MiningFSS [18]	16.9	58.0	61.6	41.6	85.9	25.8	64.5	67.3	44.5	86.4
	PFENet [19]	17.3	54.8	58.5	32.9	77.5	22.0	67.8	71.9	44.9	84.6
	GFS-Seg [13]	28.1	60.1	66.7	39.7	87.9	36.1	71.4	75.7	40.3	88.3
	HSNet [20]	21.0	55.4	59.5	37.1	82.8	30.1	60.7	74.7	42.4	79.5
	Ours	28.3	65.4	73.6	49.2	88.5	40.8	75.7	75.1	54.8	91.9
	Ours+Aug	30.6	70.7	78.4	53.6	93.3	46.4	78.0	79.1	59.4	93.1
Perturb. #1 (L=3)	MiningFSS [18]	18.3	55.9	59.7	38.1	83.8	24.3	62.4	66.5	44.9	84.3
	PFENet [19]	12.0	41.4	52.5	23.4	70.4	17.9	46.3	54.3	30.5	83.7
	GFS-Seg [13]	20.7	58.1	62.2	29.5	79.5	22.7	62.0	71.6	37.2	77.6
	HSNet [20]	14.7	45.2	41.5	23.0	77.6	19.6	44.4	58.5	32.3	77.7
	Ours	26.0	64.2	71.6	48.7	87.2	38.7	74.5	74.0	51.3	89.0
	Ours+Aug	27.7	69.0	64.5	52.0	92.3	42.2	79.7	75.3	54.1	92.2
Perturb. #2 (L=5)	MiningFSS [18]	13.8	55.1	59.3	37.9	83.3	21.0	62.3	64.5	41.5	83.0
	PFENet [19]	10.0	40.1	51.0	22.2	68.0	15.9	43.7	52.6	28.3	82.6
	GFS-Seg [13]	18.7	56.9	59.6	27.5	78.2	20.8	60.4	70.2	35.0	75.5
	HSNet [20]	13.5	44.1	39.3	21.6	75.4	17.7	42.6	57.0	29.8	75.9
	Ours	24.9	62.4	70.5	46.3	85.5	35.7	73.0	72.5	51.9	89.2
	Ours+Aug	27.1	67.1	75.3	50.1	90.5	40.4	79.0	76.4	62.0	92.1
Perturb. #3 (L=7)	MiningFSS [18]	15.2	52.9	56.7	34.7	80.4	20.4	60.4	63.4	42.1	81.3
	PFENet [19]	9.7	37.8	50.7	21.1	67.6	15.7	43.7	51.6	29.2	82.9
	GFS-Seg [13]	19.8	54.7	59.0	25.1	75.8	19.9	58.3	71.2	34.4	76.5
	HSNet [20]	12.3	44.1	38.6	22.5	76.4	18.3	42.5	55.6	28.1	75.9
	Ours	22.7	60.6	68.0	46.3	84.1	32.2	72.0	71.4	48.7	86.4
	Ours+Aug	24.6	64.9	71.9	50.6	89.7	38.5	78.0	75.2	51.6	91.6
Perturb. #4 (L=11)	MiningFSS [18]	9.2	49.1	51.8	29.3	76.0	14.3	60.6	57.9	37.2	79.0
	PFENet [19]	7.2	36.9	50.2	18.9	65.6	14.1	41.0	50.1	26.4	80.5
	GFS-Seg [13]	20.2	54.5	57.1	24.3	76.6	19.2	59.1	70.0	35.3	77.1
	HSNet [20]	10.1	43.6	39.3	20.6	74.0	17.9	41.4	54.4	26.7	76.3
	Ours	20.2	54.9	64.2	42.1	80.3	31.7	66.5	68.8	45.7	82.3
	Ours+Aug	21.6	59.2	70.0	44.4	87.1	36.4	70.9	71.4	50.0	84.4

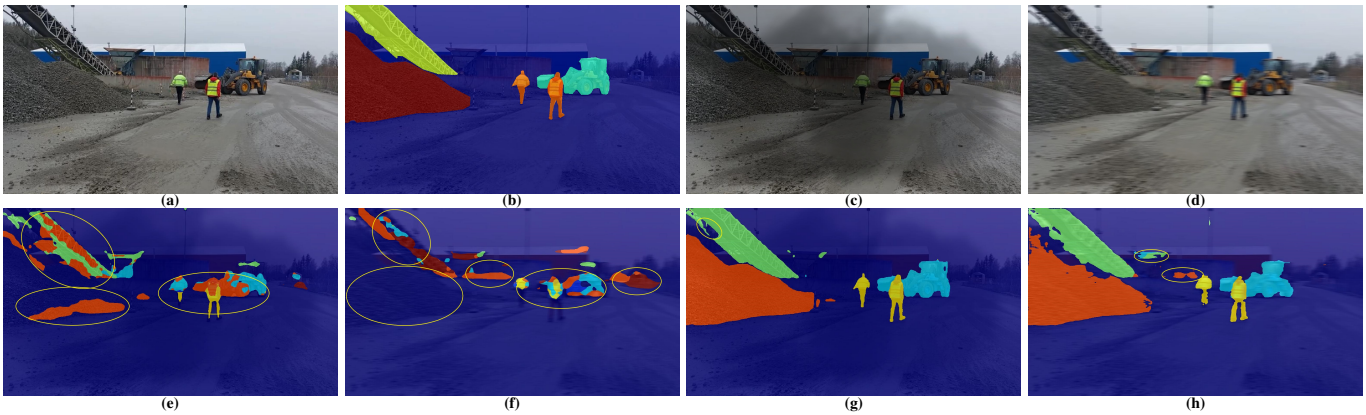

Fig. 6: Illustration of sample inputs of the prepared dataset alongside qualitative prediction results. (a) Original image. (b) Semantic label for the original image (ground truth). (c) Dirty image. (d) Blurred image. (e) Prediction result of the model with the default training setup on the dirty lens image. (f) Prediction result of the model with the default training setup on the blurry image. (g) Prediction result of the FSL+HPO model on the dirty lens image. (h) Prediction result of the the proposed FSL+HPO model on the blurry image.

to motion-blur using class-centric augmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 10 470–10 479.

[35] D. Eigen, D. Krishnan, and R. Fergus, "Restoring an image taken through a window covered with dirt or rain," in *Proceedings of the IEEE international conference on computer vision*, 2013, pp. 633–640.

[36] M. Uricar, G. Sistu, H. Rashed, A. Vobecky, V. R. Kumar, P. Krizek, F. Burger, and S. Yogamani, "Let's get dirty: Gan based data augmentation for camera lens soiling detection in autonomous driving," in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2021, pp. 766–775.

[37] T. DeVries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout," *arXiv preprint arXiv:1708.04552*, 2017.

[38] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 3–19.

[39] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu, "Cosface: Large margin cosine loss for deep face recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5265–5274.

[40] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.