IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

How to Relieve Distribution Shifts in Semantic Segmentation for Off-road Environments

Ji-Hoon Hwang, Daeyoung Kim, Hyung-Suk Yoon, Dong-Wook Kim and Seung-Woo Seo

***Abstract*— Semantic segmentation is crucial for autonomous navigation in off-road environments, enabling precise classification of surroundings to identify traversable regions. However, distinctive factors inherent to off-road conditions, such as source-target domain discrepancies and sensor corruption from rough terrain, can result in distribution shifts that alter the data differently from the trained conditions. This often leads to inaccurate semantic label predictions and subsequent failures in navigation tasks. To address this, we propose ST-Seg, a novel framework that expands the source distribution through style expansion (SE) and texture regularization (TR). Unlike prior methods that implicitly apply generalization within a fixed source distribution, ST-Seg offers an intuitive approach for distribution shift. Specifically, SE broadens domain coverage by generating diverse realistic styles, augmenting the limited style information of the source domain. TR stabilizes local texture representation affected by style-augmented learning through a deep texture manifold. Experiments across various distribution-shifted target domains demonstrate the effectiveness of ST-Seg, with substantial improvements over existing methods. These results highlight the robustness of ST-Seg, enhancing the real-world applicability of semantic segmentation for off-road navigation.**

I. INTRODUCTION

Research on off-road mobile robot navigation has gained significant attention in recent years due to its diverse applications, including transportation [1], disaster detection [2], and agricultural robotics [3]. A key component of these applications is semantic segmentation [4], a representative task in robot perception area that assigns a semantic label to each pixel, offering critical navigation information not provided by geometric sensor data. However, due to the unstructured and highly variable nature of off-road environments, previous semantic segmentation method often fail to maintain performance, leading to incorrect predictions in real-world scenarios [5]–[7] (Fig. 1). Such inaccurate perception results can lead to critical damage to the robot, particularly in off-road environments where rough terrain and unpredictable obstacles are prevalent [1].

In this paper, we observe that this issue arises from a phenomenon known as *distribution shift*, which occurs in deep learning when the environmental conditions encountered during deployment differ from those of the training dataset [8]. We conceptualize this problem from two perspectives: *external distribution shift* and *internal distribution shift*. The *external distribution shift* stems from discrepancies between the pre-acquired source domain and the target domain encountered during robot operation. This is prominent in off-road environments, where unstructured objects like natural elements lack fixed shapes and vary across locations and times (Fig. 1-(a)). The *internal distribution shift* results from corrupted sensor data as the robot navigates unstructured off-road environments. Unlike urban settings, where robots traverse smooth roads, off-road environments force robots to move over rough terrain, significantly impacting sensors with noise and blur (Fig. 1-(b)). Thus, internal shift can occur even in the source domain that has already been learned.

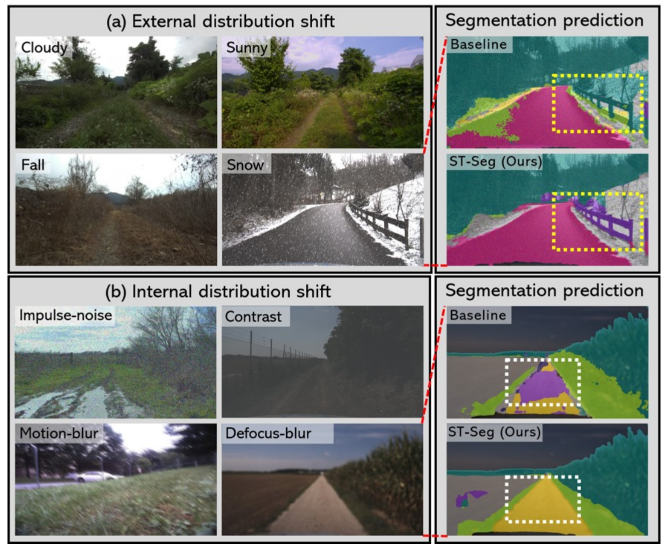

Fig. 1: **Performance of ST-Seg on various distribution-shifted target domains.** Examples of (a) external and (b) internal distribution shifts during off-road navigation are shown. The yellow box marks cases where the baseline fails to detect obstacles, while the white box shows misclassification of traversable terrain as obstacles.

In the context of supervised learning, the *distribution shift* arises due to the limited *style* information in the source domain used for training. The *style* [9] refers to the feature-level statistics that capture the global appearance of an image, a key characteristic of a domain. Since the model cannot access any information about the target domains during training, [7] removed style information to learn domain-agnostic features. While this approach reduces domain-specific biases, it also weakens feature representation power. To address this, [10] introduced random style adjustment and applied style-augmented learning by training with style-augmented features. However, random style adjustment generates unrealistic styles, leading to inconsistencies with real-world styles, and since this random approach cannot be said to effectively expand distribution coverage, it cannot be considered an explicit solution to distribution shift. We also observe that altering style information disrupts *texture* [11], which represents the feature embedding values capturing an

All authors are with Department of Electrical and Communication Engineering, Seoul National University, Korea hoons21@snu.ac.kr

image's local patterns and is partially influenced by style [12].

Due to these issues, previous semantic segmentation methods still fail to maintain their performance in distribution-shifted target domains.

In this paper, we propose ST-Seg based on our observations to relieve distribution shifts by incorporating Style Expansion (SE) and Texture Regularization (TR) methods. To address inconsistencies caused by non-realistic, randomly generated styles, SE improves adaptability to distribution-shifted target domains by performing style-augmented learning with realistic and diverse generated styles, leading to broader domain coverage. Specifically, the style generator utilizes the extensive style information from the ImageNet [13] to learn a distribution model of realistic styles. The style sampler subsequently employs stratified sampling [14] to extract diverse, unbiased styles, enabling the generation of extensive style-augmented features. Additionally, to mitigate the instability of local texture information that inevitably arises when adjusting global style information, TR regularizes texture information using a deep texture manifold obtained from natural texture data [15], thereby maintaining a consistent representation of natural textures and improving the robustness of style-augmented learning.

Contributions of this paper are summarized as follows:

- We propose an explicit solution to mitigate distribution shifts by expanding the source distribution itself, in contrast to previous methods that implicitly apply generalization techniques within a fixed source distribution.
- We propose style expansion (SE), which broadens domain coverage through style augmentation learning, and texture regularization (TR), which ensures consistent texture feature representation. Together, these methods enhance the robustness of semantic segmentation models under distribution shifts.
- We demonstrate that ST-Seg consistently maintains robust performance under both internal and external distribution shifts, validated through diverse off-road datasets, including challenging real-world scenarios collected from our UGV platform.

II. Related Works

A. Semantic Segmentation in Off-road Environments

It is important to understand the navigation characteristics and traversability of the terrain for off-road navigation [1]. The advancement of deep learning networks and the emergence of datasets from various off-road environments have led to extensive research in semantic segmentation for off-road environments [16]–[21]. From early CNNs [22]–[24] to transformer-based networks [5], [25], various architectures have been employed to off-road environments, with focusing on efficient designs to enable real-time operation. However, it is crucial to develop specialized training strategies designed for distribution shift of unstructured nature in off-road environments, rather than depending solely on existing validated segmentation networks.

Recent studies have applied additional modules to extract robust features, allowing the model to implicitly address the distribution shift problem during the training phase [5], [26]. However, this implicit approach still fails to maintain performance on distribution-shifted target domains, as it provides guidance to the model using only source data that is not exposed to distribution shifts. We propose ST-Seg, which changes the approach by explicitly utilizing domain-specific information, termed style, to expand the known domain coverage of the model and enable robust predictions in the distribution-shifted target domain.

B. Style Information in Semantic Segmentation

The channel-wise mean and standard deviation of the feature map, known as style [9], is a fundamental element representing a specific domain. Numerous studies have been utilized style information to enhance the generalization performance of semantic segmentation. [27] adjusts the style of all batch normalization (BN) layers in the network to match the style of target domain, preventing performance degradation in the target domain. While this works fine when target domain is accessible, however in off-road navigation settings, access to the target domain is not feasible.

To address a more practical problem where the target domain cannot be accessed, [7] have adopted instance normalization (IN) based methods, which separates style information during the training process to learn domain-agnostic features. While using IN to transform all samples into a standard normal distribution during training phase can improve the generalization performance of the model, it weakens the feature representation power because the style information is removed. To overcome the limitations of IN based methods, instead of removing the styles during the training phase, [10] diversified the distribution of style through random style adjustments; however, this random style may introduce disparities with real-world style. Moreover, style-augmented learning approaches that alter style may introduce variations in texture information [12], resulting in unstable learning of feature representations. To address these issue, we propose ST-Seg which can expands the limited style distribution of the source domain in a realistic and unbiased manner and stabilize the learning of feature representation.

III. Methodology

A. Problem Formulation

Task Settings The source domain dataset $D_s = \{(X_s, Y_s)\}$ refers to the training data for which we have class labeled ground truth, while the target domain $D_t = \{(X_t)\}$ refers to the dataset that has distribution shift and is inaccessible during the training process. $X_s, X_t \in \mathbb{R}^{H\times W\times 3}$ is an image, $Y_s \in \mathbb{R}^{H\times W\times C}$ is pixel-level semantic label, C is the number of semantic classes, H and W are the height and width of the image. Our goal is to train the source encoder ϕ_{source} and decoder ψ_s to better predict pixel-level class probability maps $P_s \in \mathbb{R}^{H\times W\times C}$ across multiple distribution-shifted target domains D_t while learning solely on the source domain D_s.

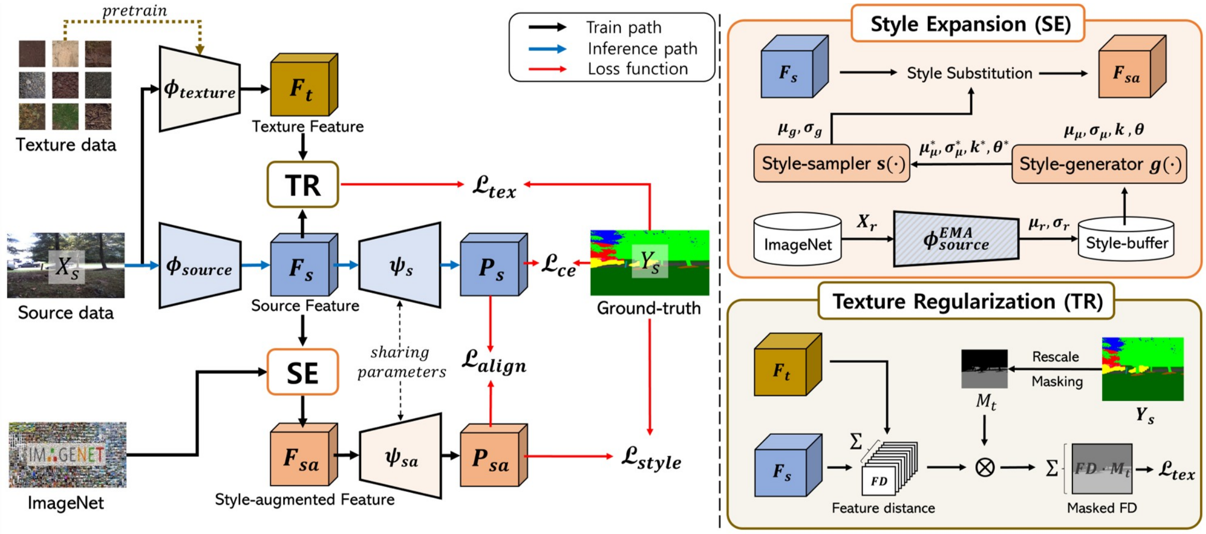

Fig. 2: **The overall framework of ST-Seg.** The left diagram illustrates how SE and TR are applied throughout the entire architecture. The right diagram is a block representation to facilitate understanding of the SE and TR methods individually. A detailed explanation of the figure can be found in the subsection III-B.

Style and Texture The channel-wise mean and standard deviation of the feature map, known as style information, is a global information representing a specific domain [9]. Following this works, let $F_s^l \in \mathbb{R}^{d \times H_l \times W_l}$ be the l-th layer feature of X_s from the source encoder network ϕ_{source}, where l denotes the number of layer in encoder and d denotes the dimension of channels. The style statistics $\mu_s^l \in \mathbb{R}^d$, $\sigma_s^l \in \mathbb{R}^d$ of the feature F_s^l can be calculated as follows:

$$\mu_s^l = \frac{1}{H_s W_s} \sum_{h=1}^{H_l} \sum_{w=1}^{W_l} F_s^l, \tag{1}$$

$$\sigma_s^l = \sqrt{\frac{1}{H_l W_l} \sum_{h=1}^{H_l} \sum_{w=1}^{W_l} (F_s^l - \mu_s^l)^2}. \tag{2}$$

The pixel-wise feature embedding values, known as texture information [11], represent the detailed local content of the scene, and are captured in the intermediate feature map F_s^l of an encoder ϕ_{source}. The pixel-wise texture value can be formulated as follows:

$$F_s^l(u, v) \in \mathbb{R}^d. \tag{3}$$

The layer notation l has been omitted to simplify the explanation of the equations introduced below.

B. Overview

The overall framework of ST-Seg is illustrated in Fig. 2. SE method guides our network to accurately predict ground truth labels, even when the source domain's style is substituted with a new one, ensuring robust performance in distribution-shifted target domains. To create a variety of new styles absent in the source domain, the style generator $g(\cdot)$ learns a realistic distribution of styles from unlabeled ImageNet data $X_r \in \mathbb{R}^{H \times W \times 3}$, and the style sampler $s(\cdot)$ extracts diverse styles in an unbiased manner. These newly generated styles μ_g and σ_g are then substituted into the source features F_s, resulting in style-augmented features F_{sa}. These features propagate through the network, producing style-augmented predictions P_{sa}, which are compared with the ground truth Y_s to compute the style expansion loss $\mathcal{L}_{style}$. The network parameters learned from these style-augmented predictions are shared with the source network ϕ_{source}, enabling the network to adapt to a wide range of distribution-shifted target domains.

The TR method is inspired by the fact that most materials in off-road environments consist of natural elements. We pre-train a deep texture manifold $\phi_{texture}$ that effectively discriminates the texture information of these natural elements and regulates the texture information shared from the style-augmented predictions by constraining it with the feature distance FD.

C. Style Expansion

To overcome the limitations of previous research that relied on removing [7] or randomizing [10] style information, we propose a style expansion method that produces realistic and unbiased styles by expanding the distribution of style in a realistic and diverse manner.

Realistic Style Extraction Inspired by the demonstrated effectiveness of utilizing ImageNet to borrow various feature statistics [28], we extract realistic features by feeding ImageNet samples X_r into the encoder. In this process, to mitigate the instability of feature extraction during training and to capture significant data trends, we employ the exponential moving average (EMA) of the source encoder ϕ_{source} for style statistics. Using Eqs. (1) and (2), we can calculate the channel-wise realistic style statistics μ_r^i and σ_r^i of the feature $F_r^i = [\phi_{source}^{EMA}]_i(X_r^i)$ at the current training step i.

If we simply replace the source style μ_s, σ_s with the extracted realistic style μ_r, σ_r, the expanded style distribution will be heavily affected by the randomness of sampling X_r^i from X_r, as the new style is determined by which samples are extracted from ImageNet. Consequently, this can lead to performance fluctuations with each training seed and restrict the style distribution, as the estimated distribution relies on randomness and is constrained by ImageNet features. To address this issue, we designed a style-generator $\mathbf{g}(\cdot)$ and a style-sampler $\mathbf{s}(\cdot)$ to learn a realistic style distribution model without relying on ImageNet data, ensuring a balanced experience of both common and rare styles.

Style-generator Generally, deep learning features follow a Gaussian distribution [29]. Given the IID assumption of ImageNet samples X_r, the feature F_r^i obtained in each iteration follows an independent Gaussian distribution relative to features from other samples F_r^j, as

$$F_r^i \sim \mathcal{N}(\mu_r^i, (\sigma_r^i)^2),\ F_r^i \perp F_r^j\ for\ all\ i \neq j. \tag{4}$$

This indicates that each feature can be modeled as an independent Gaussian distribution, uncorrelated with features from other samples. Since the means of multiple independent Gaussian distributions follow a Gaussian distribution and their standard deviation follow a Gamma distribution [30], the realistic style μ_r, σ_r can be formulated as follows:

$$\{\mu_r\}_{i=1}^n \sim \mathcal{N}(\mu_\mu, \sigma_\mu^2), \quad \{\sigma_r\}_{i=1}^n \sim \Gamma(k, \theta). \tag{5}$$

We temporarily store the extracted realistic style at each iteration i in a style-buffer. Using the accumulated μ_r and σ_r as observed samples, we update the parameters of each distribution $\mu_\mu, \sigma_\mu, k, \theta$ with Bayesian update [31] manner whenever a certain number of samples in style-buffer are gathered. The style-generator $\mathbf{g}(\cdot)$ formulated as follows:

$$(\mu_\mu^*, \sigma_\mu^*, k^*, \theta^*) \leftarrow \boldsymbol{g}(\{\mu_r\}_{i=1}^n, \{\sigma_r\}_{i=1}^n;\ \mu_\mu, \sigma_\mu, k, \theta). \tag{6}$$

Style-sampler To address the issue of randomness, we designed the style-sampler $\mathbf{s}(\cdot)$ to sample in a balanced way across all ranges using stratified sampling [14], a variance reduction method that divides the population Q into distinct subgroups and samples from each to accurately reflect the original standard deviation. Specifically, in each training batch, subgroups are predefined based on the number of samples in the batch, and generated samples are evenly assigned to these subgroups. We define each subgroup boundaries s_b^μ, s_b^σ for the generated mean and standard deviation distribution as:

$$s_b^\mu = \mu_\mu^* + \sigma_\mu^* \cdot \mathcal{N}^{-1}(b/B),\ \ s_b^\sigma = \Gamma^{-1}(b/B; k^*, \theta^*) \tag{7}$$

where b is subgroup index, B represents the number of subgroups, which equals the training batch size, and $\mathcal{N}^{-1}, \Gamma^{-1}$ are the inverse CDFs of the Gaussian and Gamma distributions, respectively. Using the computed subgroup boundaries, each batch at every training step can be assigned unbiased samples of generated style statistics.

$$(\{\mu_g^b\}_{b=1}^B, \{\sigma_g^b\}_{b=1}^B) \leftarrow \boldsymbol{s}(\mu_\mu^*, \sigma_\mu^*, k^*, \theta^*) \tag{8}$$

Through the above derivation, the style-sampler $\mathbf{s}(\cdot)$ samples diverse samples, effectively balancing the representation of styles and ensuring access to both common and rare styles within the modeled distribution.

Style Substitution Now, we can simply substitute the style of the source feature F_s with output of the style-sampler for each train-batch.

$$F_{sa} = \sigma_g \cdot \frac{F_s - \mu_s}{\sigma_s} + \mu_g \tag{9}$$

Note that, the distribution of F_s is re-normalized based on the generated statistics, which means that the spatial information is preserved in F_{sa}. The l-th style-augmented feature F_{sa}^l is then fed into next layer $l+1$. In this process, the generated style information propagates to the subsequent layers.

Style Expansion Loss We predict two class probability maps: one from the source path, where the source feature flows, and another from the additional path, where the style-augmented feature flows. The segmentation network is trained to predict the ground truth even when style augmentation is applied to the data. To achieve this, the cross-entropy loss is computed between the style-augmented probability map and the ground truth Y as follows:

$$\mathcal{L}_{style} = -\tfrac{1}{HW}\sum_{h=1}^{H}\sum_{w=1}^{W}\sum_{c=1}^{C} Y_s \log P_{sa} \tag{10}$$

Style loss encourages the network to learn style-invariant feature representations by introducing controlled perturbations in the feature space. It acts as a regularization mechanism, reducing overfitting to specific styles and enhancing robustness to unseen style variations. To ensure consistency between the source prediction P_s and the style-augmented prediction P_{sa}, we use the Kullback-Leibler (KL) divergence loss as the align loss:

$$\mathcal{L}_{align} = -\tfrac{1}{HW}\sum_{h=1}^{H}\sum_{w=1}^{W}\sum_{c=1}^{C} P_s \log \tfrac{P_s}{P_{sa}} \tag{11}$$

D. Texture Regularization

Since style information affects the global scene, it also affects local textures that constitute it [12]. Thus, altering the style can lead to changes in the texture as well. In off-road environments, which primarily consist of natural elements, textures exhibit limited variation, even though they appear in diverse forms across different locations and times. Regularizing the texture information of these natural elements stabilizes feature representation, mitigating the instability caused by fluctuating texture information.

Deep Texture Manifold We hypothesize that if an auxiliary encoder effectively extracts the deep texture manifold, it can guide the model in preserving texture knowledge, similar to the knowledge distillation method [32]. To achieve this, we pre-trained a texture encoder, $\phi_{texture}$, using a patch-level classification task by sampling natural materials from the GTOS-mobile dataset [15], which is publicly released in a study focused on the deep texture manifold. This frozen pre-trained texture encoder can be utilized as the teacher network during the training phase by extracting the deep texture manifold of a training sample, formulated as $F_t = \phi_{texture}(X_s)$, where $F_t \in \mathbb{R}^{d\times H\times W}$.

Feature Distance We designed a regularization term at the latent feature level to train the segmentation encoder to extract features that approximate those of the deep texture manifold. Since latent features are not probabilities like softmax outputs, we use L2 distance instead of cross-entropy, differing from traditional knowledge distillation:

$$FD(i,j) = \|F_t^l(i,j) - F_s^l(i,j)\|_2 \tag{12}$$

where, i and j are the pixel coordinates. However, enforcing all pixels to approximate the deep texture manifold could compromise semantic information. To address this, we selectively applied regularization to the natural element class $\mathcal{C}_{natural}$ (e.g., vegetation, soil, rock) by calculating the feature distance only for regions defined by the binary mask M_t, which corresponds to the natural element class in the ground-truth labels:

$$M_t = \sum_{c=1}^{C} Y_s \cdot [c \in \mathcal{C}_{natural}]. \tag{13}$$

Texture Regularization Loss We designed the texture regularization loss to selectively target the texture-only regions using a binary mask, defined as:

$$\mathcal{L}_{tex} = \frac{\sum_{h=1}^{H^l} \sum_{w=1}^{W^l} \gamma^l FD(i,j) \cdot M_t}{\sum_{h=1}^{H^l} \sum_{w=1}^{W^l} M_t} \tag{14}$$

where γ^l is layer-wise weight, inspired by previous deep learning studies suggesting that local texture information is predominantly processed in shallow layers [33]. Detailed information about the hyperparameters is provided in subsection IV-B.

E. Training and Inference

Training phase As illustrated in Fig. 2, the training phase optimizes the source encoder and decoder to minimize the proposed style loss, align loss, texture loss and the standard cross-entropy loss $\mathcal{L}_{ce}$ [34].

$$\mathcal{L} = \mathcal{L}_{ce} + \mathcal{L}_{style} + \mathcal{L}_{align} + \mathcal{L}_{tex} \tag{15}$$

Inference phase During training, the encoder and decoder are exposed to a variety of realistic styles through additional data and modules. Through the trained encoder-decoder, robust predictions can be made in real-time without the need for additional components or computations.

IV. Results and Analysis

A. Experimental Setup

We designed experiments to validate the effectiveness of ST-Seg in addressing the internal and external distribution shifts introduced earlier. To simulate internal distribution shifts, which arise from sensor corruption during navigation in unstructured environments, we adopted well-established image corruption methods [35] to construct the validation set. For external distribution shifts, which occur due to discrepancies between the source and target domains encountered during the operational phase, we split the training and validation data at the dataset level to evaluate performance. We benchmark our models on the standard segmentation metrics: mean intersection over Union (mIoU), which evaluates the overlap between predicted and ground truth regions, and mean pixel accuracy (mAcc), which measures the average proportion of correctly classified pixels per class.

$$IoU_c = \frac{\sum_{x,y} \mathbb{1}(P(x,y)=c \text{ and } Y(x,y)=c)}{\sum x,y \mathbb{1}(P(x,y)=c \text{ or } Y(x,y)=c)}, \quad mIoU = \frac{\sum_c IoU_c}{\sum_C 1} \tag{16}$$

$$mAcc = \frac{\sum_{c \in C}(\sum_{x,y,Y(x,y)=c} \mathbb{1}(P(x,y)=Y(x,y)))}{\sum_C 1} \tag{17}$$

For an image I, let $P(x,y)$ and $Y(x,y)$ be the prediction and ground truth labels at pixel (x,y), $\mathbb{1}(\cdot)$ be the indicator function, c be the class index, and C as the set of class labels.

We also qualitatively validated ST-Seg on challenging real-world data collected from both our Clearpath Husky UGV and the small mobile UGV, Frodobots [36], demonstrating its effectiveness in highly unstructured environments. The latency of the models used for performance comparison was measured on the NVIDIA Jetson AGX Orin.

B. Implementation Details

Backbone The backbone architecture of ST-Seg is the smallest version of the mixed vision transformer (MiT-B0) [25], which efficiently extracts multi-scale features, ensuring fast inference while achieving notable performance in semantic segmentation benchmarks.

Off-road Dataset RUGD [16], RELLIS [17], and GOOSE [21] datasets offer detailed class categorizations and high-quality labels, while the TAS [18], DeepScene [19], and YCOR [20] datasets provide simpler categorizations with lower-quality labels. Thus, the first three datasets were combined to form the source training domain, termed RGR. To validate internal distribution shifts, we generated RGR-C as target domain by applying image corruption methods to RGR. The applied corruptions include brightness, contrast, defocus blur, impulse noise, Gaussian noise, snow noise, and frost lens effects. For external distribution shift validation, RGR used as the source domain, while TDY served as the target domain. Official splits were employed for all datasets. To standardize class categories across different datasets, we remapped based on traversability levels relevant to off-road navigation {background, smooth, rough, bumpy, soft vegetation, hard vegetation, puddle, and obstacle.}.

Details of hyper-parameter setting All models were trained using the hyperparameters specified by their respective authors, with a batch size of 8. Optimization was performed using the AdamW optimizer with a learning rate of 0.00006, betas (0.9, 0.999), and a weight decay of 0.01. A polynomial learning rate policy with a power of 1.0 was applied. The EMA model was updated with a smoothing factor of $\alpha =$ 0.7, and the style-sampler's population size Q was set to 10,000. Style-augmented features through the SE method were applied only to the first two layers. The texture regularization loss weights γ^l were set to {0.05, 0.025, 0.01, 0.005} for each layer l. Our implementation is based on MMSeg [37]. All comparison models were initialized with ImageNet pre-trained weights, and performance was averaged over five training seeds.

C. Results and Compraisons

Baselines To ensure real-time operation on the mobile robot, lightweight segmentation models with strong performance were used as baselines [22]–[25]. These models represent the state of the art in real-time semantic segmentation. Additionally, GaNav [5], which demonstrated the best performance in off-road environments, and the method proposed in

TABLE I: Performance Comparisons for Internal Distribution-shifted Target Data (RGR-C).

Method (mIoU / mAcc)	Memory / Latency	Brightness	Contrast	Defocus-blur	Motion-blur	Impulse-noise	Gaussian-noise	Snow-noise	Frost-lens	*Avg.*
BiseNetv2 [22]	14.78 M / 108.7 ms	71.56 / 80.88	38.78 / 50.63	53.90 / 67.68	64.39 / 77.56	24.94 / 37.66	26.87 / 41.21	32.82 / 43.16	38.72 / 47.67	43.99 / 55.80
DeepLabv3+ [23]	14.47 M / 149.3 ms	77.55 / 86.01	48.56 / 62.86	52.80 / 66.30	66.48 / 80.06	26.66 / 40.43	29.07 / 42.80	37.58 / 48.98	45.42 / 57.01	48.02 / 60.56
MobileNetv3 [24]	**3.28 M** / 84 ms	73.68 / 83.48	47.73 / 62.67	51.12 / 71.38	64.36 / 78.54	33.59 / 45.74	35.10 / 47.26	38.70 / 51.92	44.63 / 57.11	48.61 / 62.26
SegFormer [25]	3.72 M / **74.6 ms**	80.13 / **88.27**	67.04 / 79.06	60.75 / 70.93	72.79 / 82.89	36.28 / 50.63	39.23 / 52.46	43.39 / 60.57	50.92 / 62.03	56.32 / 68.36
GaNav [5]	6.10 M / 102 ms	76.19 / 85.09	55.18 / 65.97	69.56 / 80.44	74.37 / 84.18	42.39 / 53.08	45.75 / 57.53	38.48 / 49.56	42.11 / 52.08	55.50 / 65.99
Lin et al. [7]	11.00 M / 112.4 ms	79.08 / 87.84	62.79 / 75.91	58.22 / 69.77	69.32 / 81.09	36.79 / 51.79	44.82 / 57.75	42.41 / 57.55	51.68 / 54.32	55.64 / 68.25
ST-Seg (Ours)	3.72 M / **74.6 ms**	**80.29** / 88.01	**70.46 / 81.51**	**76.45 / 85.34**	**78.79 / 87.06**	**62.26 / 70.60**	**61.22 / 69.50**	**67.44 / 76.81**	**59.87 / 70.24**	**69.60 / 78.63**

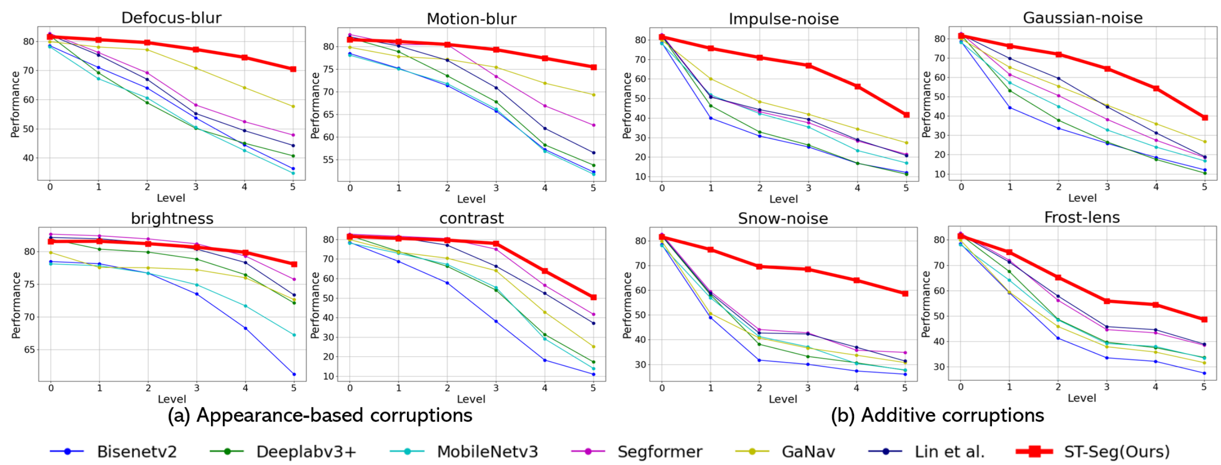

Fig. 3: **Results and Comparisons across Corruption Levels for Internal Distribution Shift.** We illustrated the performance trends for each type of corruption using line charts. In each chart, the x-axis represents the corruption level, while the y-axis indicates performance. The bold red line represents the performance of our proposed ST-Seg, which demonstrates the best ability to maintain its original performance even as the corruption levels increase.

[7], which applies instance normalization to extract general information, were also used as baselines.

Results on Internal Distribution Shift To verify the robustness in internal distribution shift, we conducted performance validation on RGR-C and presented in TABLE I. All models were trained on the training split of the RGR dataset and evaluated on the corrupted data from the RGR-C validation split. With our proposed learning strategies, ST-Seg achieved an average mIoU of 69.60% across all types and levels of corruption, showing a +13.28% improvement over the second-best baseline [25]. The main improvements, observed for defocus-blur (+15.7%), impulse-noise (+25.98%), gaussian-noise (+21.99%), and snow-noise (+24.05%), demonstrate the effectiveness of our model's ability to learn realistic styles from ImageNet, enabling robust performance on corruption types that closely mirror real-world scenarios. Performance trends across corruption levels are shown in Fig. 3, highlighting the ability to maintain performance of ST-Seg as corruption level increases. Qualitative results in Fig. 4 demonstrate that ST-Seg produces robust predictions, closely matching the ground truth even on corrupted data.

Results on External Distribution Shift To verify robustness under external distribution shift, we conducted experiments on unseen target data and presented in TABLE II. All models were trained on the training split of the RGR and evaluated on both the validation split of the source domain (RGR) and various unseen target domains (TDY).

Our ST-Seg model shows an improvement in mIoU compared to the SegFormer [25] with the same backbone: TAS (+16.23%), Deepscene (+19.86%), and YCOR (+5.74%), despite slight performance decrease in the source domain (-1.13%). These results indicate that the slight decline in the source domain RGR is a reasonable trade-off, given the significant performance gains achieved across various external distribution shifted target domains (TDY), including internal distribution shifted target domain (RGR-C).

TABLE II: Performance Comparisons for External Distribution-shifted Target Data. (TDY)

Method (mIoU / mAcc)	RGR	TAS	Deepscene	YCOR
BiseNetv2 [22]	78.48 / 86.32	43.38 / 59.28	34.60 / 36.80	35.97 / 47.82
DeepLabv3+ [23]	81.93 / 89.38	53.59 / 68.40	48.04 / 56.23	36.89 / 45.86
MobileNetv3 [24]	78.12 / 86.44	50.02 / 63.41	40.08 / 43.69	31.07 / 45.38
SegFormer [25]	**82.68 / 89.56**	39.99 / 55.62	40.10 / 46.05	34.39 / 48.11
GaNav [5]	79.86 / 87.87	45.80 / 62.89	52.62 / 58.62	34.57 / 47.52
Lin et al. [7]	82.17 / 89.21	50.62 / 65.76	54.78 / 62.61	36.99 / 48.06
ST-Seg (Ours)	81.55 / 88.64	**56.22 / 69.37**	**59.96 / 67.43**	**40.13 / 50.57**

Navigation Safety and Efficiency. To validate safety and efficiency from a navigation perspective, we conducted experiments on precision and recall on RGR-C. Precision, which measures the proportion of correctly predicted positives, indicates that high precision for the traversable class reduces the risk of misclassifying non-traversable areas as traversable, thereby lowering collision risks. Conversely, high precision for the non-traversable class means fewer occurrences of misclassifying traversable areas as non-traversable, which contributes to generating more efficient paths. Recall, the proportion of correctly predicted positives out of all actual

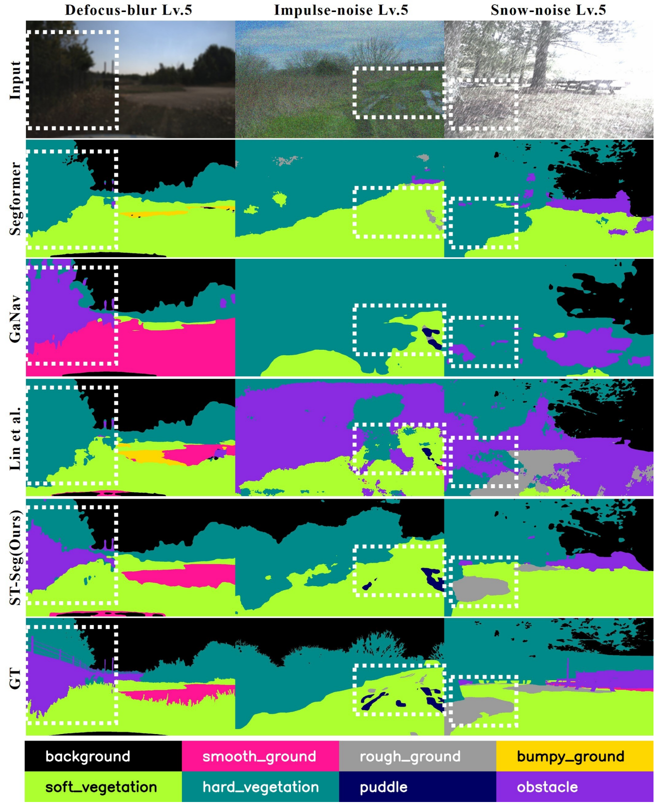

Fig. 4: **Qualitative results on RGR-C.** Our method achieves better segment of different pixel-wise classes, as shown in the white box, closely matching the ground truth.

positives, suggests that high recall for the traversable class reflects high navigation efficiency, similar to precision. On the other hand, recall for the non-traversable class is directly related to navigation safety.

$$precision = \frac{TP}{TP+FP}, \quad recall = \frac{TP}{TP+FN} \tag{18}$$

As shown in Fig. 6, our ST-Seg outperforms the baseline model [25] in all cases, highlighting its advantages in both driving safety and efficiency.

Effectiveness of SE, TR We conducted an ablation study to evaluate our approach on the corrupted seen domain RGR-C and the unseen domain TDY, as shown in TABLE III. As a baseline, we used SegFormer [25] with the same backbone network. To demonstrate the advantage of using realistic style statistics, we first experimented with two approaches: applying random style adjustment [10] to the baseline and training with a naive substitution of realistic style statistics from ImageNet.

TABLE III: Ablation Study of Proposed Method.

Method (mIoU / mAcc)	RGR-C	TDY	Avg.
Baseline [25]	56.32 / 68.36	38.16 / 49.93	47.24 / 59.15
Baseline + Rand style [10]	62.20 / 71.90	43.63 / 53.41	52.92 / 62.66
Baseline + Real style	64.12 / 74.37	45.66 / 56.39	54.89 / 65.38
Baseline + SE	66.67 / 75.87	48.02 / 58.35	57.35 / 67.11
Baseline + TR	58.78 / 69.31	46.70 / 58.59	52.24 / 63.95
Baseline + SE + TR	**69.60 / 78.63**	**52.10 / 62.46**	**60.85** / 70.55

Each approach showed an increase in average mIoU of +5.68% and +7.65%, respectively, compared to the baseline. The SE method, which utilizes style-generator and style-sampler for realistic and unbiased style generation, demonstrated a +2.46% improvement compared to the naive realistic style approach.

Furthermore, when combined with the TR method to compensate for texture discrimination power, it showed an overall increase of +5.96%. This demonstrates that the texture loss $\mathcal{L}_tex$ from the TR texture feature F_t contributes to consistent texture representation, mitigating the instability of local textures caused by the SE module.

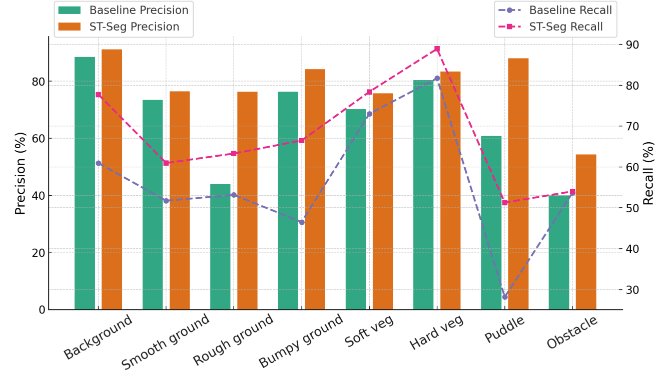

Fig. 5: **Precision and Recall Comparisons on RGR-C.**

D. Real world evaluation with mobile robot

We also qualitatively validated the proposed learning framework in challenging real-world scenes using our Clearpath Husky and Frodobots [36]. With the Husky, experiments were carried out on rough terrain under low-light mountain conditions. For the Frodobots, we experiments under challenging factors such as debris on the lens and intense backlighting from sunlight. As shown in Fig. 6, ST-Seg produces more robust predictions than the baseline [25] on distribution-shifted target data. It not only delivers more accurate semantic predictions but also significantly reduces segmentation noise, resulting in more consistent outputs across surrounding areas.

V. CONCLUSIONS

In this paper, we introduced ST-Seg, a novel learning framework to enhance the performance of semantic segmentation models in off-road environments under distribution shifts. By expanding the limited styles of the source domain through SE and stabilizing feature representation via TR, we achieved significant improvements in model robustness. Our experiments demonstrated that ST-Seg maintains performance effectively under both internal and external distribution-shifted target domains, enhancing the real-world applicability of off-road semantic segmentation. Future work will focus on enabling the model to incrementally adapt to distribution-shifted target domains during operation, making the navigation framework more flexible and responsive to real-time environmental changes.

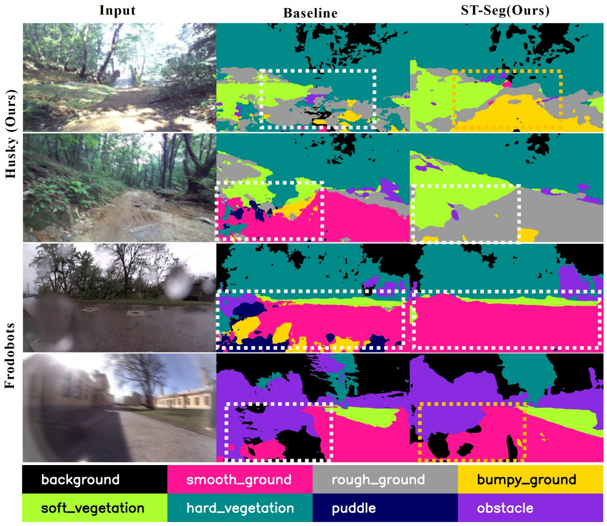

Fig. 6: **Qualitative results on challenging real-world data.** Compared to the baselines, which produce noisy and incorrect predictions, our method achieves clearer and more accurate segmentation, as shown in the white box. The areas highlighted with yellow boxes show that, in cases of severe corruption and extreme edge cases, the predictions are still not as perfect as humans.

References

[1] H.-S. Yoon, J.-H. Hwang, C. Kim, E. I. Son, S.-W. Yoo, and S.-W. Seo, "Adaptive robot traversability estimation based on self-supervised online continual learning in unstructured environments," *IEEE Robotics and Automation Letters*, vol. 9, no. 6, pp. 4902–4909, 2024.

[2] R. R. Murphy, S. Tadokoro, and A. Kleiner, *Disaster Robotics*, pp. 1577–1604. Cham: Springer International Publishing, 2016.

[3] L. F. P. Oliveira, A. P. Moreira, and M. F. Silva, "Advances in agriculture robotics: A state-of-the-art review and challenges ahead," *Robotics*, vol. 10, no. 2, 2021.

[4] S. Hao, Y. Zhou, and Y. Guo, "A brief survey on semantic segmentation with deep learning," *Neurocomputing*, vol. 406, pp. 302–321, 2020.

[5] T. Guan, D. Kothandaraman, R. Chandra, A. J. Sathyamoorthy, K. Weerakoon, and D. Manocha, "Ga-nav: Efficient terrain segmentation for robot navigation in unstructured outdoor environments," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 8138–8145, 2022.

[6] K. Viswanath, K. Singh, P. Jiang, P. Sujit, and S. Saripalli, "Offseg: A semantic segmentation framework for off-road driving," in *2021 IEEE 17th international conference on automation science and engineering (CASE)*, pp. 354–359, IEEE, 2021.

[7] N. Lin, W. Zhao, S. Liang, and M. Zhong, "Real-time segmentation of unstructured environments by combining domain generalization and attention mechanisms," *Sensors*, vol. 23, no. 13, p. 6008, 2023.

[8] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, p. 1180–1189, JMLR.org, 2015.

[9] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *Proceedings of the IEEE international conference on computer vision*, pp. 1501–1510, 2017.

[10] X. Yue, Y. Zhang, S. Zhao, A. Sangiovanni-Vincentelli, K. Keutzer, and B. Gong, "Domain randomization and pyramid consistency: Simulation-to-real generalization without accessing target domain data," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 2100–2110, 2019.

[11] D. Ulyanov, V. Lebedev, A. Vedaldi, and V. Lempitsky, "Texture networks: Feed-forward synthesis of textures and stylized images," *arXiv preprint arXiv:1603.03417*, 2016.

[12] M. D. Augustin, H. Leder, F. Hutzler, and C.-C. Carbon, "Style follows content: On the microgenesis of art perception," *Acta psychologica*, vol. 128, no. 1, pp. 127–138, 2008.

[13] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255, Ieee, 2009.

[14] R. Singh, N. S. Mangat, R. Singh, and N. S. Mangat, "Stratified sampling," *Elements of survey sampling*, pp. 102–144, 1996.

[15] J. Xue, H. Zhang, and K. Dana, "Deep texture manifold for ground terrain recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 558–567, 2018.

[16] M. Wigness, S. Eum, J. G. Rogers, D. Han, and H. Kwon, "A rugd dataset for autonomous navigation and visual perception in unstructured outdoor environments," in *International Conference on Intelligent Robots and Systems (IROS)*, 2019.

[17] P. Jiang, P. Osteen, M. Wigness, and S. Saripalli, "Rellis-3d dataset: Data, benchmarks and analysis," 2020.

[18] K. A. Metzger, P. Mortimer, and H.-J. Wuensche, "A Fine-Grained Dataset and its Efficient Semantic Segmentation for Unstructured Driving Scenarios," in *International Conference on Pattern Recognition (ICPR2020)*, (Milano, Italy (Virtual Conference)), Jan. 2021.

[19] A. Valada, G. Oliveira, T. Brox, and W. Burgard, "Deep multispectral semantic scene understanding of forested environments using multimodal fusion," in *International Symposium on Experimental Robotics (ISER)*, 2016.

[20] D. Maturana, P.-W. Chou, M. Uenoyama, and S. Scherer, "Real-time semantic mapping for autonomous off-road navigation," in *Field and Service Robotics*, pp. 335–350, Springer, 2018.

[21] P. Mortimer, R. Hagmanns, M. Granero, T. Luettel, J. Petereit, and H.-J. Wuensche, "The goose dataset for perception in unstructured environments," 2024.

[22] C. Yu, C. Gao, J. Wang, G. Yu, C. Shen, and N. Sang, "Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation," *International Journal of Computer Vision*, vol. 129, pp. 3051–3068, 2021.

[23] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 801–818, 2018.

[24] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, *et al.*, "Searching for mobilenetv3," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1314–1324, 2019.

[25] 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. 12077–12090, 2021.

[26] Z. Yang, Y. Tan, S. Sen, J. Reimann, J. Karigiannis, M. Yousefhussien, and N. Virani, "Uncertainty-aware perception models for off-road autonomous unmanned ground vehicles," *arXiv preprint arXiv:2209.11115*, 2022.

[27] Y. Li, N. Wang, J. Shi, J. Liu, and X. Hou, "Revisiting batch normalization for practical domain adaptation," 2017.

[28] L. Hoyer, D. Dai, Y. Chen, A. Köring, S. Saha, and L. V. Gool, "Three ways to improve semantic segmentation with self-supervised depth estimation," *CoRR*, vol. abs/2012.10782, 2020.

[29] C. Viroli and G. J. McLachlan, "Deep gaussian mixture models," *Statistics and Computing*, vol. 29, pp. 43–51, 2019.

[30] G. Casella and R. L. Berger, *Statistical Inference*. Pacific Grove, CA: Duxbury Press, 2nd ed., 2002.

[31] T. Bayes, "An essay towards solving a problem in the doctrine of chances," *Biometrika*, vol. 45, no. 3-4, pp. 296–315, 1958.

[32] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," in *NIPS Deep Learning and Representation Learning Workshop*, 2015.

[33] J. Lin, Y.-G. Wang, W. Tang, and A. Li, "Multi-feature co-learning for image inpainting," in *2022 26th International Conference on Pattern Recognition (ICPR)*, pp. 296–302, IEEE, 2022.

[34] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, p. 84–90, may 2017.

[35] D. Hendrycks and T. G. Dietterich, "Benchmarking neural network robustness to common corruptions and surface variations," *arXiv preprint arXiv:1807.01697*, 2018.

[36] FrodoBots Lab, "Frodobots-2k (revision 1abf1b8)," 2024.

[37] M. Contributors, "MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark." `https://github.com/open-mmlab/mmsegmentation`, 2020.