FIFO: Learning Fog-invariant Features for Foggy Scene Segmentation

Sohyun Lee
GSAI, POSTECH
lshig96@postech.ac.kr

Taeyoung Son*
NALBI
taeyoung@nalbi.ai

Suha Kwak
GSAI, POSTECH
suha.kwak@postech.ac.kr

http://cvlab.postech.ac.kr/research/FIFO/

Abstract

Robust visual recognition under adverse weather conditions is of great importance in real-world applications. In this context, we propose a new method for learning semantic segmentation models robust against fog. Its key idea is to consider the fog condition of an image as its style and close the gap between images with different fog conditions in neural style spaces of a segmentation model. In particular, since the neural style of an image is in general affected by other factors as well as fog, we introduce a fog-pass filter module that learns to extract a fog-relevant factor from the style. Optimizing the fog-pass filter and the segmentation model alternately gradually closes the style gap between different fog conditions and allows to learn fog-invariant features in consequence. Our method substantially outperforms previous work on three real foggy image datasets. Moreover, it improves performance on both foggy and clear weather images, while existing methods often degrade performance on clear scenes.

1. Introduction

We have witnessed great advances in semantic segmentation for the last decade. However, most of existing models and datasets focus merely on improving accuracy under controlled environments, without considering image degradation caused by adverse weather conditions (e.g., fog, rain, and snow), over- and under-exposure, motion blur, sensor noise, etc. The robustness of semantic segmentation models against these factors is of great importance in safety-critical applications and recently has gained increasing attention [3, 6, 8, 50–52, 56, 66].

Motivated by this, we study semantic segmentation of foggy scenes, whose goal and results are illustrated in Fig. 1. The task is challenging since fog often damages visibility of images seriously, leading to substantial performance degradation. Attaching a fog removal network to the front of an existing model is not always useful for mitigating this issue [47, 52] as well as being heavy in computation and memory. The other reason for the difficulty is the absence of fully annotated data for the task. Collecting a large set of foggy scenes is not straightforward since they can be captured under only a specific condition, and it is hard to label them due to their limited visibility.

Existing methods [8, 51, 52] tackle these issues through synthetic foggy image datasets, which are obtained by applying realistic fog effects to fully annotated clear weather images and are used for supervised learning of semantic segmentation. Furthermore, they introduce curriculum learning approaches [8, 51] that gradually adapt a model from light synthetic fog to dense real fog using unlabeled real foggy images additionally. Although these methods have achieved impressive robustness, there remains room for further improvement in that their training strategies are limited to ordinary supervised learning. In addition, the curriculum adaptation demands external modules to control the fog density of real foggy images in training, and tends to make the final model biased to foggy scenes; it thus requires extra computation and additional hyper-parameters in training, and often degrades performance on clear images.

To resolve the above issues, we propose a new method that learns Fog-Invariant features for FOggy scene segmentation, dubbed FIFO. Its overall pipeline is illustrated in Fig. 2. FIFO considers the fog condition of an image as its...
style, ideally independent of its content, and aims to learn a segmentation model insensitive to fog style variation of input image. To this end, we first define three different domains of training images, i.e., clear weather (CW), synthetic fog (SF), and real fog (RF), where images of the first two domains are labeled while those of the last one are not. FIFO then encourages the segmentation network to close the style discrepancy between different fog domains in feature spaces so that it learns fog-invariant features.

Then the success of FIFO depends heavily on the quality of the fog style representation. Unfortunately, existing style representation schemes [17, 60] are not desirable for our task since they are manually designed to capture the holistic style of an image that is affected also by factors other than fog (e.g., when and where the image was taken) and even the content of the image [6]; the direct use of these neural styles thus introduce side-effects like content alteration and result in suboptimal solutions consequently.

To address this issue, we present fog-pass filters, learnable modules that take an ordinary neural style—the Gram matrix of a feature map [17] as input and extract only a fog-relevant information from the style precisely in the form of embedding vectors, called fog factors. In particular, they learn to draw fog factors of the same domain together and hold those of different domains apart so that they discriminate fog conditions of input images through their fog factors. The segmentation model is in turn encouraged to reduce the gap between fog factors of images from different domains during training. The alternating optimization of the fog-pass filter and the segmentation network gradually closes the fog style gap between different domains and eventually leads to fog-invariant features of the segmentation network. Note that the fog-pass filters are auxiliary modules for training only, thus not required in testing.

FIFO has advantages over the previous work [8, 51, 52] in terms of both simplicity in training and efficacy in testing. Unlike the previous work, FIFO does not need to control fog density levels of synthetic and real foggy images in training, thus allowing end-to-end learning of a segmentation model with fewer hyper-parameters. More importantly, for the same reason, it demands no extra module for estimating and manipulating fog density of real foggy images in training. Regarding the effectiveness, FIFO clearly outperforms all existing records and improves performance on both foggy and clear weather domains, while existing methods often degrade performance on clear scenes.

2. Related Work

Semantic Foggy Scene Segmentation. Previous work [8, 51, 52] has developed fog simulators that are applied to clear images with full annotations to obtain labeled synthetic foggy images. Since supervised learning on the synthetic data limits performance due to the visual gap between synthetic and real foggy images, recent methods [8, 51] further employ curriculum learning to gradually adapt a model from light synthetic fog to dense real fog. However, the curriculum adaptation often degrades performance on clear weather images and demands extra modules to control density levels of real foggy images during training.

Image Dehazing. Fog damages visibility of image, and accordingly, degrades visual recognition performance substantially. Numerous dehazing algorithms have been proposed so far to restore latent clean image from foggy input [2, 5, 11, 12, 21, 36, 40, 67]. However, they are usually too heavy in computation to be attached to the front of recognition models. Further, recent studies [47, 52] suggest
that most dehazing models do not help improve recognition performance on foggy scenes. Hence, instead of having a separate dehazing module, our work learns a segmentation model whose features are invariant to fog condition.

**Robustness.** Robustness of recognition models against adverse conditions has been actively investigated due to its importance in real-world applications [13, 19, 23, 25, 27, 31, 49, 53, 66], and a variety of methods have been proposed to improve robustness [6, 35, 54–56, 64]. FIFO shares a similar idea with RobustNet [6], which regards adverse condition of input as its style and removes the effect of photometric transform of input from a neural style representation of a model so that the model becomes invariant to the transform. Compared to RobustNet, FIFO more explicitly quantifies the effect of adverse condition through a learnable module (i.e., fog-pass filter), so is able to more precisely manipulate the adverse effects during training.

**Style Transfer.** Neural style transfer has been studied to comprehend the style of an image apart from its content [1, 10, 18, 28, 37, 57, 59]. In particular, a seminal work [17] studied the Gram matrix of a feature map as a neural style representation and showed that the style of an image can be transferred to another by approximating its Gram matrix; the efficacy of Gram matrix has been proven further in later studies [32, 41]. Also, Li et al. [38] proved that matching Gram matrices is useful for domain adaptation since it is equivalent to minimizing maximum mean discrepancy between domains. However, we found that Gram matrix is not appropriate as-is for quantifying the effect of fog in our task since it is affected other style factors or even content [6] as well as fog. We thus introduce the fog-pass filter to precisely capture only a fog-relevant factor from the Gram matrix of a feature map.

**Unsupervised Domain Adaptation (UDA).** Our work is also relevant to UDA since both adapt models to an unlabeled target domain. UDA methods for semantic segmentation can be categorized by the level at which adaptation is performed: Input-level [26, 33, 45, 48], feature-level [42, 58, 63], and output-level [43, 68, 69]. FIFO is related in particular to the feature-level adaptation that learns domain-invariant features. Most of existing methods in this category [42, 58, 63] train a discriminator together with a segmentation model so that the discriminator maximizes a discrepancy between source and target domains while the segmentation model learns to minimize the discrepancy. FIFO shares a similar idea with these methods, but as will be demonstrated, closing the gap between fog factors in FIFO is more effective than fooling a fog domain classifier in fog-invariant feature learning.

### 4. Proposed Method

The training procedure for the fog-pass filters and the segmentation network is illustrated in Fig. 2 and their structures are presented in Fig. 3. The segmentation network is first pretrained on the Cityscapes dataset [7], and fog-pass filters initialized randomly are attached to different feature maps of the network. Thereafter, on the dataset of the three fog domains introduced in Sec. 3, the two parts of FIFO are trained alternately per mini-batch, except for the first 5K iterations where the fog-pass filters are solely trained to avoid cold-start. Note that the fog-pass filters are used only in training for fog-invariant feature learning of the segmentation network. In other words, FIFO imposes no additional inference-time complexity since it employs only the segmentation network for testing.

To construct a mini-batch, we randomly sample the same number of images from CW and RF domains, and choose SF counterparts of the sampled CW images. Given such a mini-batch, the fog-pass filters are learned to draw fog factors of the same fog domain together and hold those of different domains apart so that they discriminate input according to its fog domain. On the other hand, the segmentation network is optimized for closing the distance between fog factors of different domains as well as for minimizing the ordinary segmentation loss. This alternating optimization closes the fog style gap between different domains precisely, leading to fog-invariant features.

The remaining part of this section first gives details of training the fog-pass filtering modules and the segmentation network, then empirically verifies the key ideas of FIFO.
of the segmentation network. Then the fog factors of the two images are computed by $\phi_{i}^{a,l} = F^{l}(u_{i,a,l}^{a})$ and $\phi_{i}^{b,l} = F^{l}(u_{i,b,l}^{b})$, respectively, where $u_{i,a,l}^{a}$ and $u_{i,b,l}^{b}$ denote the vectorized upper triangular parts of the Gram matrices computed from their $l$th intermediate feature maps. The role of the fog-pass filter is to inform the segmentation network how $I^{a}$ and $I^{b}$ are different in terms of fog condition through $\phi_{i}^{a,l}$ and $\phi_{i}^{b,l}$. For this purpose, the fog-pass filter learns a space of fog factors where those of the same fog domain are grouped closely together and those of different domains are far from each other. Given the set of every image pair $\mathcal{P}$ in the mini-batch, the loss function for $F^{l}$ is designed as follows:

$$
\mathcal{L}_{F^{l}} = \sum_{(a,b) \in \mathcal{P}} \left\{ (1 - I(a,b)) \left[ m - d(\phi_{i}^{a,l}, \phi_{i}^{b,l}) \right]_{+} + I(a,b) \left[ d(\phi_{i}^{a,l}, \phi_{i}^{b,l}) - m \right]_{+} \right\},
$$

where $d(\cdot)$ is the cosine distance, $m$ is a margin, and $I(a,b)$ denotes the indicator function that returns $1$ if $I^{a}$ and $I^{b}$ are of the same fog domain and $0$ otherwise.

### 4.2. Segmentation Network

The segmentation network is trained using three different objectives, which are designed for semantic segmentation, fog-invariant feature learning, and consistent prediction regardless of fog condition of input, respectively. We elaborate on each of the loss functions below.

**Segmentation Loss.** For learning semantic segmentation, we apply the pixel-wise cross-entropy loss to individual images. To be specific, the loss is given by

$$
\mathcal{L}_{seg}(\mathbf{P}, \mathbf{Y}) = -\frac{1}{n} \sum_{i} \sum_{j} Y_{i,j} \log P_{i,j},
$$

where $P_{i,j} \in \mathbb{R}$ and $Y_{i,j} \in \{0, 1\}$ denote the predicted score and groundtruth label of class $j$ at pixel $i$, respectively, while $n$ is the number of pixels.

**Fog Style Matching Loss.** Given a pair of images from different fog domains, the segmentation network learns fog-invariant features that close the distance between their fog factors. To this end, the second loss matches the two fog factors given by the frozen fog-pass filters. Let $f_{i}^{a,l}$ and $f_{i}^{b,l}$ be the fog factors of the images computed by the fog-pass filter $F^{l}$. Then the loss is given by

$$
\mathcal{L}_{fsm}^{l}(\phi_{i}^{a,l}, \phi_{i}^{b,l}) = \frac{1}{4d_{l}n_{l}^{2}} \sum_{i=1}^{d_{l}} \left( \phi_{i}^{a,l} - \phi_{i}^{b,l} \right)^{2},
$$

where $d_{l}$ and $n_{l}$ denote the dimension of their fog factors and the spatial size of the $l$th feature map, respectively.

**Prediction Consistency Loss.** A CW image and its SF counterpart have exactly the same semantic layout. By forcing predictions for these images being identical, we can
align the CW and SF domains more aggressively in the learned representation. Hence, only for CW and SF images of the same origin, we encourage the model to predict the same segmentation map. Let $P^\text{CW}_i \in \mathbb{R}^c$ and $P^\text{SF}_i \in \mathbb{R}^c$ denote their class probability vectors predicted by the segmentation model for pixel $i$, where $c$ is the number of classes. The third loss is designed to force the consistency between $P^\text{CW}_i$ and $P^\text{SF}_i$ for all pixels, and is given by

$$L_{\text{con}}(P^\text{CW}_i, P^\text{SF}_i) = \sum_i \text{KLdiv}(P^\text{CW}_i, P^\text{SF}_i),$$  \hspace{1cm} (4)$$

where $\text{KLdiv}(\cdot, \cdot)$ is the Kullback–Leibler divergence. This loss shares the same goal with the fog style matching loss in Eq. (3), but more strongly forces fog-invariance in the prediction level via a small number of CW–SF pairs. Also, it is complementary to the segmentation loss in Eq. (2) since the probability distributions in Eq. (4) provide information beyond the class labels used by the segmentation loss.

**Training Strategy.** Given a mini-batch, the same number of image pairs are sampled from each of the three different domain pairs, i.e., CW–SF, CW–RF, and SF–RF. Note that images of each CW–SF pair are of the same semantic layout so that the prediction consistency loss is applied to them. For CW–SF pairs, the segmentation network is trained by minimizing

$$L^\text{CW-SF}_S = L_{\text{seg}}(P^\text{CW}, Y^\text{CW}) + L_{\text{seg}}(P^\text{SF}, Y^\text{SF})$$

$$+ \lambda_{\text{fsm}} \sum_l L_{\text{fsm}}(f^\text{CW,l}, f^\text{SF,l}) + \lambda_{\text{con}} L_{\text{con}}(P^\text{CW}, P^\text{SF}),$$  \hspace{1cm} (5)$$

where $\lambda_{\text{fsm}}$ and $\lambda_{\text{con}}$ are balancing hyper-parameters and $Y^\text{CW} = Y^\text{SF}$. On the other hand, for the other pairs of input domains including RF, the loss consists of the segmentation and fog style matching terms only, and is given by

$$L^\text{RF-SF}_S = L_{\text{seg}}(P^\text{D}, Y^\text{D}) + \lambda_{\text{fsm}} \sum_l L_{\text{fsm}}(f^\text{D,l}, f^\text{F,l}),$$  \hspace{1cm} (6)$$

where $D \in \{\text{CW, SF}\}$. Note that $L_{\text{seg}}$ is not applied to the prediction for real foggy image $P^\text{RF}$ due to the absence of its segmentation label.

### 4.3. Empirical Verification

**Impact of Fog-pass Filtering Modules.** To justify the use of the fog-pass filters, we compare Gram matrices and their fog factors computed by a fog-pass filter in how well they disentangle the fog condition and the other aspects of an image. To this end, we examine distributions of Gram matrices and fog factors of CW, SF, and RF. To be specific, training images of the Cityscapes dataset [7], their synthetic foggy counterparts given by the fog simulator [8] with the attenuation coefficient $\beta = 0.005$, and those of the Foggy Zurich dataset [51] are adopted as CW, SF, and RF images, respectively; their Gram matrices are computed from ResBlock1 outputs of RefineNet-lw [46] pre-trained on the Cityscapes, and the corresponding fog factors are computed from the Gram matrices through the fog-pass filtering module. Fig. 4(a) presents t-SNE visualization [61] of the distributions, in which Gram matrices of CW and SF largely overlap each other while fog factors are well separated according to their fog domains. This result suggests that Gram matrices are affected substantially by image content while fog factors represent only fog-relevant information as desired. The same trend is observed in Fig. 4(b) that quantitatively evaluates the quality of $k$-means clusters [20] of the Gram matrices and fog factors via adjusted Rand index [29].

**Fog-invariance Learned by FIFO.** To investigate the impact of FIFO on fog-invariance learning, we first demonstrate that FIFO effectively reduces the gap between fog domains in the space of fog factors. To this end, each domain is represented as the set of fog factors of its images, and the gap between a pair of domains is measured by the average Hausdorff distance [9] between such sets of the domains before and after training with FIFO. Fig. 4(c) shows that FIFO closes the fog-style gap in all three domain pairs. The impact is also verified qualitatively through images reconstructed from intermediate features of the segmentation.
network trained by FIFO. As a reconstruction model, we adopt RefineNet-lw with two additional upsampling layers; its decoder is first trained to reconstruct images while freezing its encoder pretrained on the Cityscapes dataset, then the encoder is replaced with that of the segmentation network trained by FIFO. Also, for comparisons, we reconstruct images using a baseline (i.e., training only on the Cityscapes) and a variant of FIFO with no fog-pass filter (i.e., training by directly reducing the gap between Gram matrices) in the same manner. Fig. 5 presents examples of the reconstructed images, which demonstrate that FIFO allows to sharpen images effectively, well emphasize object boundaries in particular, and make fewer artifacts.

5. Experiments

5.1. Implementation Details

Network Architecture. We adopt RefineNet-lw [46] with ResNet-101 [22] backbone as our segmentation network. Two fog-pass filtering modules are then respectively attached to the outputs of Conv1 and ResBlock1 layers of the segmentation network. As illustrated in Fig. 3(a), the two modules are implemented by multi-layer perceptrons with leaky ReLU activation functions [44].

Optimization and Hyper-parameters. The segmentation network is trained by SGD with a momentum of 0.9 and the initial learning rate of $6e^{-4}$ for the encoder and $6e^{-3}$ for the decoder; both learning rates are decreased by polynomial decay with a power of 0.5. The two fog-pass filtering modules are trained by Adamax [34] with initial learning rates of $5e^{-4}$ (Conv1) and $1e^{-3}$ (ResBlock1), respectively; the dimensionality of fog factors is set to 64. Each mini-batch is constructed by sampling 4 images from each fog domain, thus its size is 12. During training, images are resized, cropped to $600 \times 600$, and flipped horizontally at random. Finally, the hyper-parameters $\lambda_{\text{ fsm }}$, $\lambda_{\text{con }}$, and $m$ are set to $5e^{-8}$, $1e^{-4}$, and 0.1, respectively.

1We conduct this experiment only for qualitative analysis on the effect of FIFO. Note that FIFO directly learns fog-invariant features while bypassing explicit fog removal of input image, thus does not reconstruct images at all in both training and testing.

5.2. Datasets for Evaluation

FIFO is evaluated and compared with previous work on three real foggy datasets: Foggy Zurich (FZ) test v2 [51], Foggy Driving (FD) [52], and Foggy Driving Dense (FDD) [51]; FDD is a subset of FD. Images of these datasets depict various fog densities and are fully annotated. Also, they share the same class set with the Cityscapes dataset as described in [8]. We further apply FIFO and previous work to an unseen clear weather dataset, Cityscapes lindau 40 introduced in [8], to evaluate their performance on clear weather scenes as well.

5.3. Quantitative Analysis

Quantitative results of FIFO and previous arts are summarized in Table 1. As shown in the table, FIFO largely outperforms CMAda3+ [8], the current best performing model based on RefineNet backbone [39], on all the three foggy image datasets. These results indicate that our method of closing the fog style gap is superior to the curriculum adaptation on real images with various fog densities.

We also validate the performance of the models on the clear weather dataset. In this experiment, the accuracy of CMAda3+ drops substantially; we suspect this is a side effect of the curriculum adaptation, which may lead to overfitting to foggy scenes due to catastrophic forgetting. On the other hand, FIFO enhances performance on clear weather, probably because of the data augmentation effects of using the three domains altogether during training.

5.4. Qualitative Results

FIFO is qualitatively compared with two other methods. One is RefineNet-lw trained only on the Cityscapes dataset, which we call baseline. The other is a reduced version of FIFO using no fog-pass filtering module; this method learns the segmentation network while closing the gap between Gram matrices from different domains. Qualitative examples of their predictions are presented in Fig. 6. The baseline yields poor results in most cases. The reduced version of FIFO outperforms the baseline, especially for car, road, and vegetation classes, but FIFO using the fog-pass filtering modules clearly demonstrates the best segmentation results.
Table 1. Quantitative results in mean intersection over union (mIoU) on three real foggy datasets—Foggy Zurich (FZ) test v2, Foggy Driving Dense (FDD), Foggy Driving (FD), and a clear weather dataset—Cityscapes lindau 40.

<table>
<thead>
<tr>
<th>Method</th>
<th>Clear-weather Cityscapes</th>
<th>Synthetic &amp; Real fog SDBF</th>
<th>Real fog GoPro</th>
<th>FZ test v2 (%)</th>
<th>FDD (%)</th>
<th>FD (%)</th>
<th>Cityscapes lindau 40 (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RefineNet [39]</td>
<td>✓</td>
<td></td>
<td></td>
<td>34.6</td>
<td>35.8</td>
<td>44.3</td>
<td>67.2</td>
</tr>
<tr>
<td>RefineNet-lw [46]</td>
<td>✓</td>
<td></td>
<td></td>
<td>28.5</td>
<td>35.9</td>
<td>43.6</td>
<td>63.8</td>
</tr>
<tr>
<td>AdSegNet [58]</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>25.0</td>
<td>15.8</td>
<td>29.7</td>
<td>-</td>
</tr>
<tr>
<td>AdvEnt [62]</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>39.7</td>
<td>41.7</td>
<td>46.9</td>
<td>61.7</td>
</tr>
<tr>
<td>FDA [65]</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>22.2</td>
<td>29.8</td>
<td>21.8</td>
<td>39.3</td>
</tr>
<tr>
<td>DANN [16]</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>43.1</td>
<td>44.1</td>
<td>46.0</td>
<td>60.1</td>
</tr>
<tr>
<td>DANN-Gram</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>43.4</td>
<td>43.3</td>
<td>47.3</td>
<td>67.1</td>
</tr>
<tr>
<td>CMAda2+ [8]</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>43.4</td>
<td>40.1</td>
<td>49.9</td>
<td>-</td>
</tr>
<tr>
<td>CMAda3+ [8]</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>46.8</td>
<td>43.0</td>
<td>49.8</td>
<td>59.6</td>
</tr>
<tr>
<td>FIFO</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td><strong>48.4</strong></td>
<td><strong>48.9</strong></td>
<td><strong>50.7</strong></td>
<td>64.8</td>
</tr>
</tbody>
</table>

Table 5.5. Comparison with UDA

The task of FIFO is ostensibly identical to that of unsupervised domain adaptation (UDA) since both of them adapt models to an unlabeled target domain. Hence, one may wonder how well UDA models work for semantic foggy scene segmentation under the setting of FIFO. In this context, FIFO is compared with multiple UDA methods based on various levels of adaptation: FDA [65] for input-level adaptation, AdSegNet [58] and AdvEnt [62] for output-level adaptation, and DANN [16] for feature-level adaptation. We also evaluate a variant of DANN whose domain classifier takes as input a Gram matrix, instead of a raw feature map, like FIFO; this variant is denoted by DANN-Gram. The UDA models are trained using the same datasets, i.e., CW, SF, and RF. The input- and output-level adaptation models are pretrained on CW and further trained using SF while adapting to RF. DANN and DANN-Gram are trained using CW, SF, and RF at once like FIFO: Their discriminators are optimized to maximize the discrepancy of fog domains like our fog-pass filters while their segmentation networks are learned to minimize the discrepancy.

The performance of these UDA methods is reported in Table 1. As shown in the table, the UDA models are all inferior to FIFO and CMAda3+, which suggests that, even though it is apparently similar to UDA for semantic segmentation, semantic foggy scene segmentation is of a different nature and has its own challenges. In the typical UDA setting, each source and target domain has its own style, by which it can be defined. However, the style of a foggy scene is not determined only by its fog condition, rather is the result of the chemical combination of fog and other style factors of the scene. The UDA methods that consider SF and RF as domains with unique styles are thus not well suited to the task. Moreover, fog substantially damages visibility, and enlarges intra-domain variations since the effect of fog varies significantly according to the 3D configuration of the scene [8]. Due to these differences and challenges, foggy scene segmentation demands dedicated solutions like FIFO.

Figure 6. Qualitative results on the real foggy datasets. (a) Input images. (b) Baseline. (c) FIFO without the fog-pass filtering. (d) FIFO. (e) Groundtruth.
Frost denote clear weather, synthetic fog, and real fog, respectively. pares FIFO with its variants with and without \( L \) ing loss contribute to performance on real foggy images. boosts performance, which suggests that all the three pairs favors the use of the fog-pass filters.

Moreover, unlike the current best-performing method, it enables end-to-end learning and demands no extra module nor human intervention for training.

Acknowledgement: This work was supported by Samsung Research Funding & Incubation Center of Samsung Electronics under Project Number SRFC-IT1801-05.

Table 2. Analysis on the impact of domain pairs. CW, SF and RF denote clear weather, synthetic fog, and real fog, respectively.

<table>
<thead>
<tr>
<th>CW–SF</th>
<th>CW–RF</th>
<th>SF–RF</th>
<th>FZ</th>
<th>FDD</th>
<th>FD</th>
<th>CW</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓</td>
<td>✓</td>
<td></td>
<td>43.7</td>
<td>38.6</td>
<td>46.1</td>
<td>67.6</td>
</tr>
<tr>
<td>✓</td>
<td></td>
<td>✓</td>
<td>37.7</td>
<td>40.3</td>
<td>47.2</td>
<td>66.0</td>
</tr>
<tr>
<td></td>
<td>✓</td>
<td>✓</td>
<td>39.3</td>
<td>42.8</td>
<td>49.7</td>
<td>61.6</td>
</tr>
</tbody>
</table>

Table 3. Analysis on the impact of the fog style matching loss, the prediction consistency loss, and the fog-pass filtering modules.

<table>
<thead>
<tr>
<th>Method</th>
<th>FZ</th>
<th>FDD</th>
<th>FD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>28.5</td>
<td>35.9</td>
<td>43.6</td>
</tr>
<tr>
<td>FIFO w/o ( L_{\text{fsm}} )</td>
<td>31.7</td>
<td>38.5</td>
<td>45.1</td>
</tr>
<tr>
<td>FIFO w/o ( L_{\text{con}} )</td>
<td>41.6</td>
<td>45.4</td>
<td>48.9</td>
</tr>
<tr>
<td>FIFO w/ Gram</td>
<td>41.3</td>
<td>43.8</td>
<td>49.1</td>
</tr>
<tr>
<td>FIFO</td>
<td>48.4</td>
<td>48.9</td>
<td>50.7</td>
</tr>
</tbody>
</table>

Table 4. Analysis on the impact of synthetic fog densities. \( \beta = 0 \) denotes a model trained with CW and RF.

<table>
<thead>
<tr>
<th>( \beta )</th>
<th>FZ</th>
<th>FDD</th>
<th>FD</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>37.7</td>
<td>40.3</td>
<td>47.2</td>
</tr>
<tr>
<td>0.0025</td>
<td>45.5</td>
<td>40.4</td>
<td>45.0</td>
</tr>
<tr>
<td>0.01</td>
<td>42.4</td>
<td>45.4</td>
<td>50.0</td>
</tr>
<tr>
<td>0.02</td>
<td>42.9</td>
<td>42.5</td>
<td>48.6</td>
</tr>
<tr>
<td>0.005</td>
<td>48.4</td>
<td>48.9</td>
<td>50.7</td>
</tr>
</tbody>
</table>

The fog style matching loss \( L_{\text{fsm}} \) is applied. Specifically, \( L_{\text{fsm}} \) is applied to the output of the first convolutional layer (C1), the first residual block (R1), or both of them (C1+R1). As summarized in Table 5, the performance improves as more feature maps are affected by the loss. Overall, C1+R1, which is our final model, shows the best performance.

5.7. Generalization to Other Weather Conditions

We investigate the generalization ability of FIFO on other weather conditions, rain and frost. To this end, RefineNet-lw trained by FIFO in Sec. 4 is evaluated as-is on rainy [27] and frosty [23] versions of the Cityscapes dataset. On the rainy Cityscapes dataset, our model is compared with existing methods reported in [13] that aim at improving robustness using CW images only. As summarized in Table 6, our model largely outperforms the previous work based on a stronger segmentation network (i.e., DeepLab v3+ [4], indicated by \( \dagger \) in the table). Fig. 7 demonstrates that FIFO generalizes to the frosty images also. More results can be found in the supplementary material.

6. Conclusion

We have presented FIFO, a new approach to learning fog-invariant features for foggy scene segmentation. It precisely quantifies the fog style of an image through the fog-pass filtering modules and learns a segmentation network for closing the gap between images of different fog conditions in the fog style space. FIFO outperforms previous arts without sacrificing performance on clear weather images. Moreover, unlike the current best-performing method, it enables end-to-end learning and demands no extra module nor human intervention for training.

Table 5. Analysis on the layers to which the fog style matching loss \( L_{\text{fsm}} \) is applied. C1 and R1 indicate the output of the first convolution layer and that of the first residual block, respectively.

<table>
<thead>
<tr>
<th>Loss</th>
<th>FZ</th>
<th>FDD</th>
<th>FD</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>45.3</td>
<td>41.0</td>
<td>48.3</td>
</tr>
<tr>
<td>R1</td>
<td>45.1</td>
<td>39.1</td>
<td>47.0</td>
</tr>
<tr>
<td>C1+R1 (Ours)</td>
<td>48.4</td>
<td>48.9</td>
<td>50.7</td>
</tr>
</tbody>
</table>

Table 6. Quantitative results on Rainy Cityscapes (RC).

Figure 7. Qualitative results under rain and frost conditions.
References


[12] Raanan Fattal. Dehazing using color-lines. ACM transactions on graphics (TOG), 2014. 2


