Let’s Get Dirty: GAN Based Data Augmentation for Soiling and Adverse Weather Classification in Autonomous Driving

Michal Uřičár¹, Ganesh Sistu¹, Hazem Rashed¹, Antonín Vobecký²,¹, Pavel Křížek¹, Fabian Bürger¹ and Senthil Yogamani¹
¹Valeo ²CTU, Prague
firstname.lastname@valeo.com

Abstract

Cameras are getting more and more important in autonomous driving. Wide-angle fisheye cameras are relatively cheap sensors and very suitable for automated parking and low-speed navigation tasks. Four of such cameras form a surround-view system that provides a complete and detailed view around the vehicle. These cameras are usually directly exposed to harsh environmental settings and therefore can get soiled very easily by mud, dust, water, frost, etc. The soiling on the camera lens has a direct impact on the further processing of the images they provide. While adverse weather conditions, such as rain, are getting attention recently, there is limited work on lens soiling. We believe that one of the reasons is that it is difficult to build a diverse dataset for this task, which is moreover expensive to annotate. We propose a novel GAN based algorithm for generating artificial soiling data along with the corresponding annotation masks. The manually annotated soiling dataset and the generated augmentation dataset will be made public. We demonstrate the generalization of our fisheye trained soiling GAN model on the Cityscapes dataset. Additionally, we provide an empirical evaluation of the degradation of the semantic segmentation algorithm with the soiled data.

1. Introduction

Level 5 autonomous driving stands out as a natural goal of a huge part of the computer vision and machine learning community. While some people argue that one type of sensor can handle all challenging situations, others believe that a combination of multiple sensory types is a must. One example for all is the fish-bone parking [12]. Although this is a quite limited scenario, such situation cannot be resolved completely via cheap ultrasonic sensors as they are misleading for the correct maneuvering. And here camera can help. Without exaggeration, it can be stated that surround view cameras are becoming de facto standard in autonomous parking [21].

A huge progress can be noticed in typical image processing tasks, such as semantic segmentation or object detection [30, 31, 32]. This is mostly attributed to the prevailing success of deep learning. However, there are other less “popular” problems slowly getting into attention which have to be solved as well for the ultimate goal of the full Level 5 autonomy [14].

One of these problems is the reliability of the sensory signal, which in case of surround view cameras means, inter alia, the ability to detect soiling on the camera lens or recognition of severe weather conditions leading to deterioration of the image quality to such a level that any further image processing is unreliable. Figure 2 shows how the surround view camera can get soiled and the corresponding image output, as well as an example of images taken during a heavy rain. Figure 1 shows an example of the strong impact of large rain drops on the camera lens for object detection and semantic segmentation tasks.

In this work, we focus on soiling caused by a variety of unwanted particles reposing on the camera lens. The source of these particles is mostly mud, dirt, water or foam created.

Figure 1: The example of a semi-transparent soiling in form of a water drop on the camera lens. The detection of the bus behind the water drop works still well, while the road segmentation (green) is highly degraded in the soiled region.
Figure 2: Automotive surround view cameras are exposed to harsh environmental setup. Left: camera lens covered by mud. Middle: image produced by the soiled camera from the left picture. Right: camera lens soiled during heavy rain.

by a detergent. Based on the state of aggregation, such soil- ing can be either static (e.g. highly viscous mud tends to dry up very quickly, so it does not change its position on the output image over time) or dynamic (mostly water, and foam). Because of that, the acquisition of suitable data for training machine learning models or simply testing the effect on existing classification models is quite tedious. The human annotation is also very time demanding and not very reliable, since the precise labeling of soiling on a single image can be sometimes very challenging. Our contributions include:

- A proposition of a baseline pipeline for an opaque soil- ing generation, based on CycleGAN [42] and semantic segmentation learned from weak labels.
- Novel DirtyGAN network, which is an end-to-end generalization of the baseline pipeline.
- Public release of an artificial soiling dataset as a companion to the recently published WoodScape Dataset [40], coined DirtyWoodScape Dataset, to encourage further research in this area.
- An empirical evaluation of degradation of several clas- sification algorithms on soiled images.

The paper is organized as follows. Section 2 covers the related work. In Section 3, we give a detailed description of the proposed algorithms. Section 4 describes the empirical evaluation of the classification algorithms’ degradation in presence of soiling and also provides the evaluation of the quality of the generated images. Finally, Section 5 con- cludes the paper.

2. Related Work

In recent years, the task of artificial image generation was mostly overtaken by Generative Adversarial Networks (GANs) [9]. GANs show great ability in synthesizing realistic-looking images as shown in [24]. Despite this great success, we can observe that GANs have much more difficulty in synthesizing certain image classes in contrast to others when trained on multi-class datasets such as ImageNet [28]. To tackle this problem, the authors of [41] introduce a self-attention mechanism into convolutional GANs. The proposed mechanism helps to model long-range dependencies across image regions and thus improves the overall quality of the generated samples.

Another well-studied area of the computer vision is the semantic segmentation aiming at assigning a label to each image pixel. This task requires a large number of pixel-level annotations that are very difficult to obtain. This problem is addressed in [33] by a design of a semi-supervised framework based on GANs. It consists of a generator network that provides additional training samples to a multi-class classifier, acting as discriminator in the GAN framework. More recently, Deeplabv3 [3] proposed to use dilated convolutions to account for larger receptive fields without the need to downscale the image.

The task of soiling detection on camera lenses in au- tonomous driving is shortly described in [34], where the au- thors present a sort of a proof of concept idea how GANs could be applied for dealing with the insufficient data prob- lem in terms of an advanced data augmentation. In the same paper the authors also outline other potential usage of GANs in the autonomous driving area. A more formal introduc- tion to the soiling detection and categorization is provided in [35], where the problem is formalized as a multilabel classification task.

2.1. Image-to-Image Translation

Image-to-Image translation is a part of graphics and computer vision that aims to learn a mapping between a source domain $X$ and a target domain $Y$ with the use of paired data. In [15], the authors present a method using GANs to tackle the problem of image-to-image translation using paired data. However, obtaining such paired data can be difficult and sometimes even impossible. Therefore, an unsupervised version, without use of any examples of cor- responding pairs, is even more important and challenging.

This problem is tackled by CycleGAN [42] with the use
of two mappings $G: X \rightarrow Y$ and $F: Y \rightarrow X$. Since these mappings are highly under-constrained, they propose to use a cycle consistency loss to enforce $F(G(X)) \approx X$. Even though it is not emphasized that much in the CycleGAN paper, the authors in their implementation use also identity losses, $G(Y) \approx Y$ and $F(X) \approx X$, which improved the results significantly.

In [20], the authors assume that a pair of corresponding images in different domains can be mapped to the same latent representation in a shared-latent space. With this assumption in hand, they propose an unsupervised image-to-image translation framework based on Coupled GANs. This work is extended to a multimodal use in MUNIT [13], which assumes that the image representation can be decomposed into a content code that is domain-invariant, and a style code that captures domain-specific properties. The authors combine its content code with a random style code to transfer a source image to a different domain.

In the unsupervised case of the image-to-image translation, the network needs to learn which parts of the scene should be preserved and which should be changed. To tackle this, works such as [22, 17] propose the use of an attention mechanism. In [11], they manipulate only certain attributes of a face image while preserving other details. This is achieved by decoding the latent representation of the given face conditioned on the desired attributes and applying an attribute classification constraint to the generated image to enforce the correct change of the desired attributes.

StarGAN [5] is a model that takes in training data of multiple domains, and learns the mappings between all available domains using only a single generator that takes as inputs both image and domain information, and learns to flexibly translate the image into the corresponding domain. However, this approach can only generate a discrete number of expressions, determined by the content of the dataset. To address this limitation, the authors of [26] introduce a conditioning scheme based on Action Units annotations, which describes in a continuous manifold the anatomical facial movements defining a human expression.

When an image has multiple target instances, and a translation task involves significant changes in shape, the standard methods often fail. To tackle this issue, InstaGAN [23] incorporates the instance information such as segmentation masks and translates both image and corresponding instance attributes.

2.2. Water Soiling & Deraining

Soiling on the camera lens has a direct impact on further processing of the images, such as semantic segmentation or object detection. One of the ways to deal with this situation is to run an image restoration algorithm that improves the quality of the image. Most of the attention is currently paid to rain removal (deraining) and water soiling. The work of [19] provides a comprehensive analysis of this topic.

The authors of [39] address the problem of rain removal from videos by a two-stage recurrent network. The rain-free image is estimated from the single rain frame at the first stage. This initial estimate serves as guidance along with previously recovered clean frames to help to obtain a more accurate clean frame at the second stage.

In [27] the authors propose a progressive recurrent deraining network by repeatedly unfolding a shallow ResNet with a recurrent layer.

In [38] a dataset of $\approx 29.5k$ rain/rain-free image pairs are constructed and a SPatial Attentive Network (SPANet) is proposed to remove rain streaks in a local-to-global manner.

Authors of [25] presented a method that improves the segmentation tasks on images affected by rain. They also introduced a dataset of clear-soiled image pairs which is used to train a denoising generator that removes the effect of real water drops.

2.3. Other Types of Soiling/Adversarial Weather

Another type of the image quality degradation is caused by the presence of aerosols (e.g., mist, fog, fumes, dust, ...) in the environment surrounding the car. Due to the light scattering caused by these aerosol particles, the resulting image tends to have faint colors and looks hazy, which can inherently also impact the further image processing.

Fattal presents in [8] a method for single image dehazing, based on a refined image synthesis model and a depth estimation. Berman et al. [2], on the other hand, propose a solution, which is not based on local priors and builds on an assumption that a dehazed image can be approximated by a few hundred distinct colors which form tight clusters in the RGB color space. Ki et al. [16] propose fully end-to-end learning based boundary equilibrium GANs to perform an ultra high resolution single image dehazing. Uricar et al. [36] provided a desoiling dataset benchmark.

3. Artificial Soiling Generation

The task of single image soiling annotation on the fish-eye cameras is quite tedious. We make use of polygonal annotation, which is a compromise of annotation speed and quality. However, even this kind of polygonal annotation is sometimes very hard to interpret even by human annotators. This is particularly true for the soiling boundary which is usually very fuzzy.

However, an even bigger problem is how to obtain this kind of data. In our setup, we apply a random pattern of soiling on the camera lens using a certain soiling source. Then drivers ride the car for a while and repeat the process several times. This has a lot of limitations: Firstly, it is very inconvenient to record data for all admissible scenarios (e.g., driving through the city, rural area, highway, etc.). Secondly, it is not possible to measure the real impact
of soiling on the images, because we would need a clean version of the exact same images for a comparison.

All these limitations motivate the use of synthetic soiling data. In the following sections the proposed soiling generation algorithms are described.

### 3.1. Soiling Generation Baseline Pipeline

The core of our baseline pipeline is formed by a CycleGAN [42] network, which we train to perform the image-to-image translation from clean images to their soiled counterparts. The main problem of the CycleGAN method is that it modifies the whole image. For our desired application this can lead to undesired artifacts in the generated images. Besides that, the generated synthetic soiling patterns are often relatively realistic. Note, that due to GPU memory requirements and time constraints, the CycleGAN training uses rescaled images (1/4 of both width and height).

Next, we train a soiling semantic segmentation network, $\mathcal{M}$, using the weak polygonal annotation of soiling (see Figure 3 for several examples). Even though the annotation is quite coarse, the segmentation network $\mathcal{M}$ is able to fit the soiling patterns more precisely. See Figure 4 for a few examples of $\mathcal{M}$ outputs in comparison to the original annotations. We use WoodScape Dataset [40] for training the soiling segmentation network.

Last but not least, we train a super-resolution network $\mathcal{U}$, which we use to transform the GAN generated image to the original image resolution (i.e. up-scaling of 4× factor).

The idea of a baseline data generation algorithm is described in Figure 5. We take the generator transforming a clean image to the soiled image ($G_{C2S}$) and apply it on the clean image $I$. This gives us an image with a random soiling pattern $I_s$. Next, we obtain the soiling mask $m$. This is achieved by applying the semantic segmentation network on the generated soiled image followed by a Gaussian smoothing filter $\gamma$: $m = \gamma (\mathcal{M}(I_s))$. The resulting soiling mask
Figure 5: The soiling generation baseline pipeline. From left to right: an image into which we would like to paint a random soiling pattern; CycleGAN generated “soiled” version; blurred mask of the segmented soiling from the generated image; the resulting artificially “soiled” image obtained by convex combination of the original image and the generated soiling via the segmented mask.

\( m \) is an image with values in range \([0, 1]\), where 0 means background, and 1 means soiling. The intermediate values can be understood as a semi-transparent soiling. We apply the Gaussian smoothing filter because it mimics the physical nature of the soiling phenomenon where the edges of the soiling patterns are typically semi-transparent, due to photon scattering. Finally, the artificially soiled version of the original image \( I \) and is a composition of the original image \( I \) and the soiling pattern \( I_s \) via the estimated mask \( m \)

\[
\hat{I} = (1 - U(m)) \cdot I + U(m) \cdot U(I_s) .
\]  

Note, it is possible to use arbitrary images for the final composition, once we obtain \( I_s \) and \( m \). The mask \( m \) obtained by the semantic segmentation network \( \mathcal{M} \) serves as an automatic annotation of the soiling in the generated image.

This simple pipeline has certain limitations. The biggest one is that it cannot be expected to work smoothly for soiling types caused by water (e.g., raindrops, snow, etc.) in this specific formulation. One option how to deal with this issue is the following approach of [1], where the authors model the reflection patterns of the water drops using the whole image and apply filters and transformations consequently. The other option is to formulate a CycleGAN-like approach, which is able to cope with the problem of changing only those parts of the image that correspond to the soiling pattern and keep the rest unchanged. We formulate this approach in the following section.

### 3.2. DirtyGAN

The problem of applying CycleGAN in artificial soiling synthesis is twofold. Firstly, CycleGAN does not constrain the image generation to any specific regions and rather re-generates the whole image, affecting all pixels. In case of the artificial soiling generation, this is highly undesired, as for the investigation of the soiling impact on the further image processing, it is crucial that the background (i.e., regions of the image not affected by the soiling) remain untouched. Secondly, the generation branch “clean” \( \rightarrow \) “soiled” is ill posed, as there is no visual clue for where the soiling should be produced. In fact, there are infinitely many patterns that could be created. Furthermore, there is also no control over the soiling pattern production process.

The first problem can be addressed, e.g., by InstaGAN [23]. However, in such case, the second problem becomes even a bigger issue. We decided to guide the pattern generation process via a Variational AutoEncoder (VAE) [7] and modify the CycleGAN algorithm so that it applies only on the masked regions of the source and target domain images. We coin the proposed network DirtyGAN.

We use the weak polygonal soiling annotations from the WoodScape Dataset for training the VAE. The main idea of using VAE for the soiling patterns generation is as follows. By using the encoder of the trained VAE, we can obtain the projection of an existing sample from the dataset to a lower dimensional representation. If we select two samples \( z_1 \) and \( z_2 \) that are close on the soiling pattern manifold, we can obtain a novel sample \( z \) by taking their convex combination

\[
z = \alpha z_1 + (1 - \alpha) z_2 ,
\]

where \( \alpha \in [0, 1] \). Then, we simply take this intermediate representation \( z \) and apply the trained decoder from VAE to reconstruct the corresponding soiling pattern. In Figure 6, we depict several examples of this intermediate soiling pattern reconstruction.

The benefit of using this sampling from the learned VAE is that we could even use it to create animated masks, e.g., to mimic dynamic soiling effects, such as water drops in heavy rain or to be able to investigate the impact of dynamic soiling in general.

After training the VAE, we simply limit CycleGAN to be applied only on the masked regions corresponding to the generated mask for the “clean” \( \rightarrow \) “soiled” translation or the mask obtained by the soiling semantic segmentation mask \( \mathcal{M} \). We use the similar composition as in the baseline presented in Section 3.1. In Figure 7, we depict the whole DirtyGAN scheme.
3.3. Dirty Datasets

We have used the baseline pipeline to generate artificial soiling on our recently published WoodScape Dataset [40] with 10k images, which comes with semantic segmentation annotation. This makes it a suitable candidate for soiling generation, since we can merge the provided annotation with the soiling mask and measure the direct impact on classification models. Our generated data with updated annotation will be released as a WoodScape Dataset companion under the name of DirtyWoodScape Dataset. In Figure 8, we show several generated examples together with their automatically obtained annotations.

Since our method of soiling generation is not limited to fisheye images only and since we would like to support standard benchmarkings as well, we also release a DirtyCityscapes dataset. It is, as the name suggests, based on the Cityscapes [6] dataset.

4. Experimental Evaluation

4.1. Degradation Effect

To demonstrate the impact of the soiling on the camera lens, we provide an empirical evaluation on semantic segmentation task. Pixel level classification is commonly used in autonomous driving applications. Our dataset consists of 10k pairs of images (clean and synthetically soiled using the proposed baseline framework). We split our data into training and testing by 80 : 20 ratio according to sampling strategy described in [37]. We trained two DeepLabV3+ [4] models on the clean and soiled images, respectively. We evaluate the performance separately on clean and soiled test data. Table 3 summarizes the obtained results.

A segmentation model trained on clean images records 56.6% mIoU on clean test data and 34.8% on soiled data, a performance drop of 21.8% compared to clean images. This significant drop shows that presence of a soiling can cause serious degradation effect to a standard visual perception task in autonomous driving.

A model trained on the synthetic soiling data shows a limited degradation to 16.1%. Training on the soiled images, however, shows 4% accuracy degradation on clean test data compared to the baseline when evaluated on clean images. 8% degradation is observed when evaluated on soiled images compared to the baseline model. This result motivates the need for studying the effect of soiling on various perception tasks.

Figure 9 depicts a qualitative evaluation of the soiling impact on the segmentation task. The baseline model trained on clean images 9a is evaluated on soiled images in 9c showing a high level of degradation due to the soiling.
Figure 8: Examples of the generated images from the DirtyWoodScape Dataset, together with the generated annotations. Top: WoodScape Dataset RGB images with generated opaque soiling. Bottom: corresponding automatically generated annotations. White pixels represent the soiling, while black pixels represent the background.

Figure 9: Qualitative evaluation using clean vs soiled images on the semantic segmentation network DeepLabV3+.

Figure 9d shows the ground truth annotations, while 9f and 9g illustrate results of model trained on the soiled images while testing on the clean images in 9a and soiled images in 9e. In a realistic scenario, annotations are not available for the occluded region of soiled images. Using our GAN generated dataset, we make use of the annotations in the soiled area to enable the model to interpolate segmentation classes in occluded soiled parts. Figure 9f shows the capability of segmentation networks to perform segmentation even behind the soiled area. However, it is less reliable compared to the clean baseline and sensitive to overfitting.

A similar trend is observed in case of the Cityscapes dataset [6]. Results are presented in Table 2.

4.2. Artificial Soiling Quality

To start with, we performed a subjective visual study of the quality of the artificial soiling similarly as it is done in other GAN based image generation algorithms. We selected representative human participants with a variable level of the problem understanding: ranging from absolutely no knowledge about the computer vision and surround view camera imaging systems for autonomous driving to people working with the soiling data on a daily basis. The participants were asked to classify the presented images either as real ones or fakes. Then we randomly showed images with the real soiling from the WoodScape Dataset [40] and with the artificially generated soiling. To make the task even more difficult, we used soiled images showing similar scenes as they occur in the WoodScape Dataset. Otherwise, the participants might eventually spot the differences in the data distribution as the real soiling data comes from a limited scenarios only.

The non-expert participants were not able to recognize real images from fakes. The expert participants were some-
Figure 10: Examples of the generated images from the DirtyCityscapes Dataset. Top: Cityscapes Dataset RGB images. Bottom: Generated soiled images corresponding to clean images.

Table 1: Comparison of Soiling Segmentation model trained on generated and real soiled images. Accuracy is computed on a real test dataset with 2000 images.

| Soiling Segmentation Model | Accuracy [%] on real test 2K dataset (mIoU) |
|---------------------------|---------------------------------------------|
| Trained solely on generated images (8000 samples) | 47.41 |
| Trained solely on real images (8000 samples) | 73.95 |
| Trained on real & generated images (16000 samples) | 91.71 |

Table 2: Quantitative evaluation on Cityscapes using ResNet50+FCN8. The accuracy measure is mIoU [%].

| Train | Test | Clean | Soiled |
|-------|------|-------|--------|
| Clean | 38.1 | 26.6  |
| Soiled | 35.5 | 38.0  |

Table 3: Quantitative evaluation on our dataset using DeepLabV3+. The accuracy measure is mIoU [%].

| Train | Test | Clean | Soiled |
|-------|------|-------|--------|
| Clean | 56.6 | 34.8  |
| Soiled | 52.1 | 48.2  |

5. Conclusions

In this paper, we proposed two algorithms for the generation of soiling on images from surround view fisheye cameras. The first algorithm is a pipeline built from several well known blocks, such as CycleGAN [42], semantic segmentation of the generated soiling and image composition. The second algorithm is a novel DirtyGAN network, which is able to generate similar results as the baseline pipeline in an end-to-end fashion. The possibility to generate random, but realistic soiling pattern on camera images is an integral component in examining the degradation of other image processing methods. We provided empirical evaluation of the performance degradation on several typical classification tasks common for autonomous driving scenarios. We demonstrate that our soiling model trained on fisheye images generalizes well on Cityscapes dataset enabling to create dirty versions of public datasets. Last but not least, we release a public dataset as a companion to the recently published WoodScape Dataset [40], coined DirtyWoodScape Dataset, which can serve as a benchmark for measuring the degradation of off-the-shelf classification algorithms.
References

[1] Stefano Alletto, Casey Carlin, Luca Rigazio, Yasunori Ishii, and Sotaro Tsukizawa. Adherent raindrop removal with self-supervised attention maps and spatio-temporal generative adversarial networks. In *The IEEE International Conference on Computer Vision (ICCV)* Workshops, Oct 2019.

[2] Dana Berman, Tali Treibitz, and Shai Avidan. Non-local image dehazing. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, Las Vegas, NV, USA, June 27-30, 2016, pages 1674–1682. IEEE Computer Society, 2016.

[3] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(4):834–848, 2018.

[4] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018.

[5] Yunjey Choi, Min-Je Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018*, Salt Lake City, UT, USA, June 18-22, 2018, pages 8789–8797. IEEE Computer Society, 2018.

[6] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3213–3223, 2016.

[7] Carl Doersch. Tutorial on variational autoencoders. *CoRR*, abs/1606.05908, 2016.

[8] Raanan Fattal. Single image dehazing. *ACM Trans. Graph.*, 27(3):72:1–72:9, 2008.

[9] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[11] Zhenliang He, Wangmeng Zuo, Meina Kan, Shiguang Shan, and Xilin Chen. AttGAN: Facial attribute editing by only changing what you want. *IEEE Trans. Image Processing*, 28(11):5464–5478, 2019.

[12] Markus Heimberger, Jonathan Horgan, Ciarán Hughes, John McDonald, and Senthil Yogamani. Computer vision in automated parking systems: Design, implementation and challenges. *Image and Vision Computing*, 68:88–101, 2017.

[13] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 172–189, 2018.

[14] S. A. E. International. Taxonomy and definitions for terms related to on-road motor vehicle automated driving systems J3016, 2018.

[15] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, Honolulu, HI, USA, July 21-26, 2017, pages 5967–5976. IEEE Computer Society, 2017.

[16] Sehwan Ki, Minjae Kim, Hyeonwoo Kang, and Kwanghee Lee. U-GAT-IT: unsupervised generative attentional networks with adaptive layer-instance normalization for image-to-image translation. *CoRR*, abs/1907.10830, 2019.

[17] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014.

[18] Siyuan Li, Iago Breno Araujo, Wenqi Ren, Zhangyang Wang, Eric K. Tokuda, Roberto Hirata Junior, Roberto Cesar-Junior, Jiawan Zhang, Xiaojie Guo, and Xiaochun Cao. Single image deraining: A comprehensive benchmark analysis. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[19] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Andrew Ng, Lawrence Saul, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, 4-9 December 2017, Long Beach, CA, USA, pages 700–708, 2017.

[20] Pullaro Maddu, Wayne Doherty, Ganesh Sistu, Isabelle Leang, Michel Uricar, Sumanth Chennupati, Hazem Rashed, Jonathan Horgan, Ciaran Hughes, and Senthil Yogamani. Fisheyemultinet: Real-time multi-task learning architecture for surround-view automated parking system. In *Proceedings of the Irish Machine Vision and Image Processing Conference*, 2019.

[21] Youssef Alami Mejjati, Christian Richardt, James Tompkin, Darren Cosker, and Kwang In Kim. Unsupervised attention-guided image-to-image translation. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018*, NeurIPS 2018, 3-8 December 2018, Montréal, Canada., pages 3697–3707, 2018.
[23] Sangwoo Mo, Minsu Cho, and Jinwoo Shin. Instagan: Instance-aware image-to-image translation. In 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019. OpenReview.net, 2019. 3, 5
[24] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019, pages 2337–2346. Computer Vision Foundation / IEEE, 2019. 2
[25] Horia Porav, Tom Bruls, and Paul Newman. I can see clearly now: Image restoration via de-raining. CoRR, abs/1901.00893, 2019. 3
[26] Albert Pumarola, Antonio Agudo, Aleix M. Martínez, Alberto Sanfeliu, and Francesc Moreno-Noguer. Ganimation: Anatomically-aware facial animation from a single image. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part X, volume 11214 of Lecture Notes in Computer Science, pages 835–851. Springer, 2018. 3
[27] Dongwei Ren, Wangmeng Zuo, Qinghua Hu, Pengfei Zhu, and Deyu Meng. Progressive image deraining networks: A better and simpler baseline. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2019. 3
[28] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV), 115(3):211–252, 2015. 2
[29] Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional networks for semantic segmentation. IEEE Trans. Pattern Anal. Mach. Intell., 39(4):640–651, 2017. 8
[30] Mennatullah Siam, Sara Elkordawy, Martin Jagersand, and Senthil Yogamani. Deep semantic segmentation for automated driving: Taxonomy, roadmap and challenges. In 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), pages 1–8. IEEE, 2017. 1
[31] Mennatullah Siam, Mostafa Gamal, Moemen Abdel-Razek, Senthil Yogamani, and Martin Jagersand. Rtsge: Real-time semantic segmentation comparative study. In 2018 25th IEEE International Conference on Image Processing (ICIP), pages 1603–1607. IEEE, 2018. 1
[32] Mennatullah Siam, Heba Mahgoub, Mohamed Zahran, Senthil Yogamani, Martin Jagersand, and Ahmad El-Sallab. MODNet: Motion and appearance based moving object detection network for autonomous driving. In Proceedings of the 21st International Conference on Intelligent Transportation Systems (ITSC), pages 2859–2864, 2018. 1
[33] Nasim Souly, Concetto Spampinato, and Mubarak Shah. Semi supervised semantic segmentation using generative adversarial network. In IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017, pages 5689–5697. IEEE Computer Society, 2017. 2
[34] Michal Uřičář, Pavel Křížek, David Hurych, Ibrahim Sobh, Senthil Yogamani, and Patrick Denny. Yes, we GAN: applying adversarial techniques for autonomous driving. Electronic Imaging, 2019(15):48–1–48–17, 2019. 2
[35] Michal Uřičář, Pavel Křížek, Ganesh Sistu, and Senthil Yogamani. Soilingnet: Soiling detection on automotive surround-view cameras. In 2019 22nd International Conference on Intelligent Transportation Systems (ITSC). IEEE, 2019. 2
[36] Michal Uřičář, Ján Uličný, Ganesh Sistu, Hazem Rashed, Pavel Křížek, David Hurych, Antonín Vobecký, and Senthil Yogamani. Desoiling dataset: Restoring soiled areas on automotive fisheye cameras. In The IEEE International Conference on Computer Vision (ICCV) Workshops, Oct 2019. 3
[37] Michal Uí, David Hurych, Pavel Kek, and Senthil Yogamani. Challenges in designing datasets and validation for autonomous driving. In Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP, pages 653–659. INSTICC, SciTePress, 2019. 6
[38] Tianyu Wang, Xin Yang, Ke Xu, Shaozhe Chen, Qiang Zhang, and Rynson W.H. Lau. Spatial attentive single-image deraining with a high quality real rain dataset. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2019. 3
[39] Wenhan Yang, Jiaying Liu, and Jiashi Feng. Frame-consistent recurrent video deraining with dual-level flow. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2019. 3
[40] Senthil Yogamani, Ciaran Hughes, Jonathan Horgan, Ganesh Sistu, Padraig Varley, Derek O’Dea, Michal Uricar, Stefan Milz, Martin Simon, Karl Amende, Christian Witt, Hazem Rashed, Sumanth Chennupati, Sanjaya Nayak, Saquib Man-soor, Xavier Perrotton, and Patrick Perez. Woodscape: A multi-task, multi-camera fisheye dataset for autonomous driving. In The IEEE International Conference on Computer Vision (ICCV), October 2019. 2, 4, 6, 7, 8
[41] Han Zhang, Ian J. Goodfellow, Dimitris N. Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA, volume 97 of Proceedings of Machine Learning Research, pages 7354–7363. PMLR, 2019. 2
[42] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. In IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017, pages 2242–2251, 2017. 2, 4, 8
A. Generated Data Influence on Soiling Semantic Segmentation

In Figure 11, we depict the problems we are facing when using the WoodScape Dataset. Because only coarse polygonal annotation is available (which is moreover prone to errors), the classical fully supervised training of the semantic segmentation for the soiling is affected by the annotation quality. If we, however, add the generated images along with the precisely generated annotations, the same fully supervised training exploits these annotations, with a desired effect on the semantic segmentation output quality.

B. Network Architectures

In this section, we provide details about the network architectures we used in our experiments.

B.1. Baseline Networks Architecture

Generator and Discriminator of CycleGAN

In the CycleGAN generator network (see Figure 12), we use the residual block depicted in Figure 14. Note, that we do not use any kind of normalization, like BatchNorm or InstanceNorm. In Discriminator (see Figure 13, we use Leaky ReLU and a fully convolutional classification layer at the end.

Soiling Mask Segmentation

In Figure 15, we show the semantic segmentation network for the soiling mask detection. For the soiling mask segmentation network, we use Instance Normalization in convolutional blocks and BatchNorm in the residual block.

Variational AutoEncoder

We depict the encoder part of the VAE in Figure 16. The $\mu$ and $\sigma$ are combined into $z$ via reparameterization trick. The VAE’s decoder is shown in Figure 17.

B.2. DirtyGAN Architecture

DirtyGAN is composed of the building blocks used in the baseline algorithm, therefore the networks architecture is mostly the same, only with some minor tweaks.

C. Generated Data

In Figure 18 and 19, we depict random frame from generated videos, which we enclose to the supplementary. To fit the size limit, we resized the images by $1/4$ for both width and height. The first image in the row is the original frame from the video sequence as it was recorded. The second image in the row is the artificially generated soiled version of the first image and the third image in the row is the soiling mask, which can serve as the automatic annotation for the soiled image. Although it is possible to use our algorithm to generate dynamic soiling effects, we present here only the static soiling. Please, see the video files for the best experience.
Figure 11: The problem of weak annotation labels and how generated data can help, presented on a testing image, which was not used during the training. The images come from the WoodScape Dataset. Left: original weak (polygonal) annotation; Note, that not only it is very coarse, but also prone to errors. Middle: semantic segmentation output, when trained on the weak annotation labels in a fully supervised manner. Right: the same semantic segmentation network output, however, in this case also the generated data were used. Note, how having precise annotation labels for the generated data help in refining the segmentation output.

Figure 12: The Generator’s architecture, used in the CycleGAN. “c7s1-32-R” reads as 2D convolutional layer with kernel size $7 \times 7$ pixels, stride 1, 32 output channels, followed by a ReLU activation layer. T stands for the tanh activation; “tc” is a shortcut for the transposed convolution; “rp” means reflection padding; “r-128” is a shorthand for residual block with 128 channels.

Figure 13: The Discriminator’s architecture, used in the CycleGAN. “c4s2-64-LR” reads as 2D convolutional layer with kernel size $4 \times 4$ pixels, stride 2, 64 output channels, followed by a Leaky ReLU activation layer. S stands for the sigmoid activation layer.
Figure 14: The residual block, which is used in the Generator’s architecture. Note, that convolutional blocks “c3s1” might be followed by a BatchNorm layer (used only in the soiling mask segmentation network).

Figure 15: The soiling mask segmentation network. The same type of shorthands as in previous figures are used with an addition of “IN” for Instance Normalization and “BN” for BatchNorm layers. “up2” is a simple nearest neighbor upsampling with scale factor 2.

Figure 16: The architecture of the VAE encoder used in our experiments. The notation “c3s2-64-R” corresponds to a 2D convolutional layer, with kernel size $3 \times 3$ pixels, stride of 2, 64 output channels, followed by a ReLU activation layer.

Figure 17: The architecture of the VAE decoder used in our experiments. The notation “c3s2-64-R” corresponds to a 2D convolutional layer, with kernel size $3 \times 3$ pixels, stride of 2, 64 output channels, followed by a ReLU activation layer. “tc3s2-256-R” represents a transposed convolution, with kernel $3 \times 3$ pixels, stride 2, 256 output channels, followed by ReLU activation. “rp-1” means a reflection padding of size 1.
Figure 18: Several frames extracted from the supplementary video files. Please, see the original video files for the best experience.
Figure 19: Several frames extracted from the supplementary video files. Please, see the original video files for the best experience.