

# CNN-based Lidar Point Cloud De-Noising in Adverse Weather

Robin Heinzler<sup>1,2</sup>, Florian Piewak<sup>1</sup>, Philipp Schindler<sup>1</sup> and Wilhelm Stork<sup>2</sup>

**Abstract**—Lidar sensors are frequently used in environment perception for autonomous vehicles and mobile robotics to complement camera, radar, and ultrasonic sensors. Adverse weather conditions are significantly impacting the performance of lidar-based scene understanding by causing undesired measurement points that in turn effect missing detections and false positives. In heavy rain or dense fog, water drops could be misinterpreted as objects in front of the vehicle which brings a mobile robot to a full stop.

In this paper, we present the first CNN-based approach to understand and filter out such adverse weather effects in point cloud data. Using a large data set obtained in controlled weather environments, we demonstrate a significant performance improvement of our method over state-of-the-art involving geometric filtering. Data is available at <https://github.com/rheinzler/PointCloudDeNoising>.

## I. INTRODUCTION

Given that lidar sensors are key for autonomous driving and robotics applications, they are currently being developed by numerous companies in a wide variety of designs. Nevertheless, lidar technology is heavily challenged in adverse weather as the range measurements are highly impaired by fog, dust, snow, rain, pollution, and smog [1]–[6]. Such conditions cause erroneous point measurements in the point cloud data which arise from the reception of back-scattered light from water drops (e.g. rain or fog) or arbitrary particles in the air (e.g. smog or dust).

For environment perception algorithms, these points are undesirable noise which needs to be specifically addressed in order to not restrain the scene understanding performance. This is particularly relevant for algorithms that make direct use of the low-level geometry of a measured point cloud, e.g. the Stixel algorithm [7], where noisy input data inevitably results in noisy Stixel output data. CNN-based lidar perception algorithms might be better able to cope with such issues given their learning capacity thereby reducing the need for an explicit handling of noisy measurements. Still, most lidar perception algorithms involve more classical bottom-up approaches for tasks such as object detection since they usually are implemented on the lidar sensors themselves with very limited computational resources. This has sparked a large body of research on algorithms to detect and handle noisy point cloud measurements in a pre-processing step before applying perception algorithms.

<sup>1</sup>Daimler AG, Benz-Str., 71063 Sindelfingen, Germany  
[firstname.lastname]@daimler.com

<sup>2</sup>Institute for Information Processing Technology (ITIV), Karlsruhe Institute of Technology (KIT), Germany  
[firstname.lastname]@kit.edu

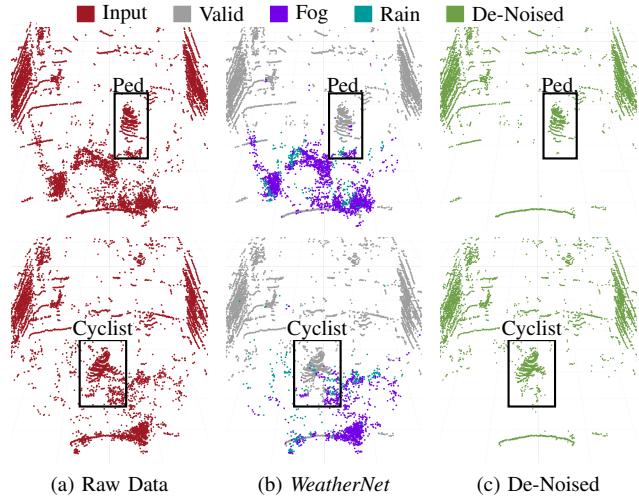


Fig. 1. De-noising results shown on snapshots of two dynamic scenes in dense fog at 20 and 30m visibility. Fig. (a) shows the raw point-cloud data (red), (b) the point-wise weather segmentation by *WeatherNet* (fog, rain, valid) and (c) the de-noised result (green) with the remaining valid points, where reflections from fog and rain are removed. The pedestrian and cyclist, which are barely recognizable in the scene, are highlighted in a black box.

To that extent, a large quantity of 2D image anti-aliasing algorithms have been developed that focus on smoothing noisy surface points resulting from marginal sensor errors [8]–[14]. De-noising algorithms in 3D space are often based on spatial features to discard noise points caused by rain or snow [4], [15]. As these techniques are discarding points based on the absence of points in their vicinity, smaller objects at medium to large distances might be falsely suppressed and marked as noise. In addition, the recordings in Fig. 1 indicate that modern lidar sensors, e.g. the Velodyne VLP32C, do not necessarily perceive drops of water from fog or rain as a single point, but often as multi-point clutter in the near to mid range which significantly reduces the applicability of filtering based on spatial vicinity only.

Experimentally validating such filtering algorithms in real-world scenarios under adverse weather conditions is very challenging due to the lack of proper ground truth. We address this by proposing an evaluation based on data recorded in controlled environments where we could obtain a large data set under various environmental conditions with point-wise ground truth annotations for different classes of weather, e.g. *clear*, *rain*, or *fog*. In addition, we employ a data augmentation approach to emulate adverse weather effects on real-world data that has been previously obtained in good weather conditions [16].

Our main contributions are as follows:

- The first CNN-based approach to lidar point cloud

de-noising with a significant performance boost over previous state-of-the-art while being very efficient at the same time.

- A data augmentation approach for adding realistic weather effects to lidar point cloud data.
- A quantitative and qualitative point-level evaluation of de-noising algorithms in controlled environments under different weather conditions.

## II. RELATED WORK

Adverse weather conditions such as fog, rain, dust or snow have a huge impact on the perception of lidar sensors, as shown in [1]–[3], [5], [17]–[23]. Consequently, point cloud processing algorithms either have to deal with these influences, or require preprocessing by filter algorithms. Nevertheless, only a few de-noising algorithms for sparse point cloud data have been developed or are publicly available yet ([4], [24]). Most state-of-the-art data sets are recorded mostly under favorable weather conditions only (e.g. [25]–[27]).

### A. Dense Point Cloud De-Noising

Previous work on 2D depth image de-noising is mainly based on dense depth information obtained by stereo vision and depth cameras (e.g. Intel RealSense, Microsoft Kinect, etc.) [13], [28]. Hence, traditional algorithms developed over years for camera image de-noising can be applied in a straightforward fashion. These approaches can be split in three different categories: (1) spatial, (2) statistical and (3) segmentation-based methods.

Spatial smoothing filters (1), e.g. the Gaussian low pass filter, calculate a weighted average of pixel values in the vicinity, where the weight decreases with the distance to the observed pixel. Points are smoothed by increasing distance from the derived weight [8]. For de-noising 2D point cloud data corrupted by snow, these filter types are providing successful results, as shown by [9] with a median filter. However by only assuming only small variations in the neighborhood, this approach generally fails at preserving edges. The bilateral filter, introduced by [8] for gray and color images, is replacing traditional low-pass filtering by providing an edge preserving smoothing filter for dense depth images [10].

Statistical filter methods (2) for dense point cloud denoising are often based on maximum likelihood estimation [29] or Bayesian statistics [11]. By optimizing the decision whether a points lies on a surface or not these approaches are smoothing surfaces and remove minor sensor errors.

By applying a segmentation step before filtering, segment-based filters (3) are smoothing only local segments of point clouds with identical labels. Therefore corners and finer structures are better preserved. Region growing [12], a maximum a-posteriori estimator [13] or edge detection [14] is used for segmentation, while bilateral filters are used for smoothing local segments.

Lidar point clouds are significantly less dense compared to camera images, particularly at larger distances. As such,

the direct application of camera algorithms does typically not achieve the desired result, as exemplified in [4] for a median filter applied to point cloud data. Since conventional lidars have a resolution of a tenths of a degree and a range of two to three hundred meters, the density of the point cloud decreases significantly in the middle and far range.

A first machine learning approach for de-noising dense point clouds corrupted by fog with a visibility of 2m and 6m is introduced in [24]. By manually extracting features, a  $k$  nearest neighbor (kNN) and a support vector machine (SVM) are trained. The feature vector is in particular based on the eigenvalues of the covariance matrix of the Cartesian coordinates, therefore it is only derived if there are more than ten points in a  $50\text{mm}^3$  cubic voxel. For a sparse lidar point cloud this assumption is rarely satisfied.

### B. Sparse Point Cloud De-Noising

In the 3D domain many approaches are based on the spatial vicinity or statistical distributions of the point cloud [15], such as the statistical outlier removal (SOR) and radius outlier removal (ROR) filter. The SOR defines the vicinity of a point based on its mean distance to all  $k$  neighbors compared with a threshold derived by the global mean distance and standard deviation of all points. The ROR filter directly counts the number of neighbors within the radius  $r$  in order to decide whether a point is filtered or not. Recently, Charron et al. [4] have shown that these filter types are not suited for the de-noising task of sparse point clouds corrupted by snow. Thus the enhanced dynamic radius outlier removal (DROR) filter was introduced by [4] which increases the search radius  $r$  for neighboring points with increasing distance of the measured point. Since this approach takes the raw data structure of lidar sensors into account, which is less dense at far distances, a better performance could be achieved.

Nevertheless, these approaches are based on spatial vicinity and consequently discard single reflections without points in the neighborhood. As a result, points at greater distances are increasingly filtered, as shown in [4] for the SOR, ROR and even DROR. Hence, valuable information for an autonomous vehicle, especially at higher speeds, is discarded and the sensor's range is additionally limited by the filter.

In addition, sparsity is not a valid feature to filter scatter caused by fog or drizzle, as soon as the density of the distribution of water drops increases (Fig. 1). In conclusion, we argue that these filter approaches are prone to failure in the near and far range, as only spatial neighborhood is used.

### C. Semantic Segmentation for Sparse Point Clouds

We propose a filter approach based on a convolutional neural network, which understands the underlying data structure and can generalize its characteristics for various distances and clutter distributions. Furthermore, this approach is able to also incorporate the intensity information of the point cloud. The semantic segmentation task is being further developed by a large scientific community and is already applied to the lidar point cloud domain, showing very promising results

[30]–[32]. A major advantage is that the algorithms can generalize very well and thus recognize objects at different distances and orientations.

There are various established approaches for the input data layer and the network structure itself, which we utilize and adapt to the task of semantic weather segmentation [31]–[36]. Since preprocessing algorithms have strong requirements on computation speed, we focus on 2D input layer approaches, which commonly use a birds eye view (BEV) [33]–[35] or an image projection view [31], [32], [36].

The recently introduced *PointPillars* by [35] is based on a feature extraction network, generating a pseudo image out of the point cloud which is used as input for a backbone *CNN*. The approach excels on the KITTI's object detection challenge [25] in terms of detection performance and inference time. However, the approach has not yet been applied to point-wise semantic segmentation but is in wide-spread use for object detection only. Thus, we propose a 2D approach inspired by the *CNN* architecture of *LiLaNet* [31].

### III. METHOD

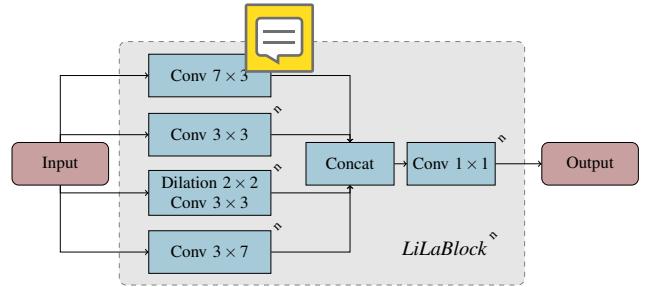
#### A. Lidar 2D Images

State-of-the-art lidar sensors commonly provide raw data in spherical coordinates with the radius  $r$ , azimuth angle  $\phi$  and elevation angle  $\theta$ , often combined with an estimated intensity or echo pulse width of the backscattered light. The used rotating lidar sensors (Velodyne 'VLP32c') contain 32 vertically stacked send/receive modules, which are rotating to obtain the  $360^\circ$  scan. Similarly to [31] we merge one scan to a cylindrical depth image as a 2D matrix  $M = (m_{i,j}) \in \mathbb{R}^{(n \times m)}$  where each row  $i$  represents one of the 32 vertically stacked send/receive modules and each column  $j$  one of the 1800 segments over the full  $360^\circ$  scan with the corresponding azimuth angle  $\phi$  and timestamp  $t$ . As a consequence we obtain the distance matrix  $D \in \mathbb{R}^{(n \times m)}$  and intensity image  $I \in \mathbb{R}^{(n \times m)}$ .

### B. Autolabeling for Noise Caused by Rain or Fog

In order to evaluate the quality of a trained classification approach, ground truth annotations are essential. For sparse lidar point clouds, the manual annotation task is very challenging and even more difficult for semantic weather segmentation, where the decision is whether a point is caused by a water droplet or not. Human comprehension of camera images is much more powerful than of lidar point clouds, therefore a time-synchronized camera image as additional information is helpful for labeling lidar point clouds in order to significantly improve the label quality. However, since water droplets cannot be captured directly by passive camera sensors, especially at large distances, this label aid is not available for semantic labeling of weather information. Thus, we utilize the recorded static scenes in controlled environments to develop an automated labeling procedure, which does not involve human perception.

We stack all  $f$  lidar images for each frame  $k$  from one sensor in reference conditions to obtain one single point



**Fig. 2.** The modified *LiLaBlock* is based on [31] and enlarged by a dilated convolution [29].

cloud  $D^{GT} = (d_{i,j,k}^{GT}) \in \mathbb{R}^{(n \times m \times f)}$ . Subsequently, we compare each distance image  $D$  captured during rain or fog with all ground truth images  $D^{GT}$  to decide whether a point is labeled as clutter or not. Since the reference measurements are accumulated over several frames, minor measurement inaccuracies of the sensor are already taken into account when comparing the distance images. In addition, a threshold  $\Delta R$  is added to the search region of valid distances. The threshold value  $\Delta R = \pm 35\text{ cm}$  was chosen rather high, compared to the specified distance precision of the sensor, in order to minimize the number of false negatives. Hence, the labels are derived mathematically for each distance  $d_{i,j}$  with the corresponding ground truth vector for this element  $d_{i,j,k}^{GT}$ :

$$p = \begin{cases} \text{clutter,} & \text{if } \Delta R \geq \min_{1 \leq k \leq f} |d_{i,j,k}^{GT} - d_{i,j}| \\ \text{no clutter,} & \text{else} \end{cases} \quad (2)$$

(no clutter,      else      (2)

By directly comparing the distances of the same transmitter and receiver pairs, this method is very fast, directly based on the sensor raw data and does not require 3D information. Alternatively, a 3D point cloud comparison was implemented by a kd-tree approach without showing significantly different results. In order to be able to distinguish between different weather conditions, fog and rain sequences are provided with different labels.

To quantify the error of our ground truth labeling, we applied the label procedure on the reference recording itself. We split the recording during reference for one setup into equally sized parts. The evaluation is done by taking the accumulation of the first split as valid points for labeling the second split and vice versa. As there are no changes in weather conditions, we would expect that all points will be labeled as valid. The evaluation demonstrates a mean per pixel false rate of  $0.367 \pm 0.053\%$  for both tests.

### C. Data Augmentation

State-of-the-art sparse point cloud data sets which are publicly accessible tend to be recorded under favorable weather conditions. To be able to utilize these data sets for semantic weather segmentation, we developed an augmentation approach for rain based on the fog model of [16]. Hence, we obtained a large training data set without requiring manual annotation while providing error-free ground truth. The augmentation algorithm is applied to lidar images to enable manipulations for each individual distance measurement, whereby occlusion is conceptually impossible. The

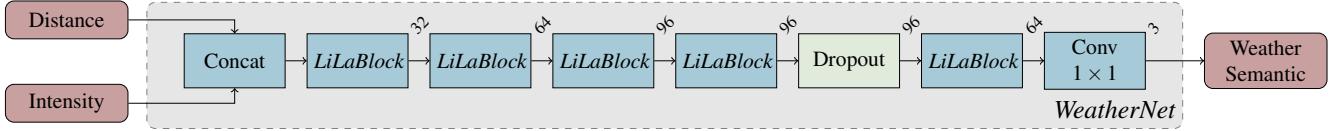


Fig. 3. The proposed *WeatherNet* architecture is based on *LiLaNet* introduced by [31] and optimized for the de-noising purpose. Therefore, the depth is reduced, a dropout layer is inserted and a dilated convolution is added to base block of the network. The modified *LiLaBlock* is given in detail in Fig. 2.

proposed augmentation based on the model of [16] does not only add individual points but alters additional attributes of the point cloud: Adverse weather affects viewing range and lowers the contrast of intensity and echo pulse widths, respectively.

**1) Fog Model:** First of all the maximum sensing range is derived by the runway visual range  $V = -\ln(C_T)/\beta$  based on the atmospheric extinction coefficient  $\beta$  and the observers contrast threshold  $C_T$ . For lidar sensors  $C_T$  can be interpreted as a detector threshold, where the sensor is able to perceive an object above the noise floor [16]. As lidar is an active sensor system, the maximum sensor range is the half of the maximum viewing distance and results in:

$$d_{max} = \frac{-\ln(\frac{n}{L_{fog}+g})}{2 \cdot \beta} \quad (3)$$

The sensor threshold is a function of the received laser intensity  $L_{fog}$ , the adaptive laser gain  $g$  and the detectable noise floor  $n$ . Scattering points due to waterdrops are added according to the model of [16]. In contrast to [16] the intensities of augmented points are derived from a logarithmic normal distribution  $LN(\mu, \sigma^2)$ , which is assumed as the underlying probability distribution function. The parameters  $\mu$  and  $\sigma$  are derived from the intensities of all clutter points based on the autolabeled climate chamber data from the previous section. Hence we are able to model the intensity distribution of fog between 10 – 100 m meteorological visibility and for rainfall rates at 15, 33 and 55 mm/h. We preferred this method because in the model of [16] the original scene is mirrored in the intensity distribution, as the augmented intensities  $\tilde{I}$  are a function of the perceived intensities  $I$  of the sensor ( $\tilde{I} = I \cdot e^{-\beta \cdot d}$ ). The augmented fog corresponds to a visibility  $V$  of 30 – 3000 m, hence we use a atmospheric extinction coefficient  $\beta$  between 0.001 and 0.1.

**2) Rain Model:** Besides our modifications of the fog augmentation based on [16], we further developed a rain augmentation. Thereby the parameters from [16] are adapted to make the augmented scatter points equivalent to natural rainfall. The atmospheric extinction coefficient  $\beta$  is set to 0.01 for rain augmentation. The point scatter rate  $p$  defines the per point probability of random scatter points. The obtained point-wise ground truth data enables the calculation of  $p$  for raindrops, which is 10.61%, 0.73% and 4.70% for 15, 33 and 55 mm/h in the climate chamber. For the augmentation we finally fixed  $p$  at 7.5%, which stabilizes the CNN training, matches the quantity of scatter points in natural rainfall and is in the range of the derived probabilities from the climate chamber. The rain augmentation is described in

**Algorithm 1:** Point cloud rain augmentation model  $rain(D, I, \beta, R)$  with distance matrix  $D$ , intensity matrix  $I$ , atmospheric extinction coefficient  $\beta$  and point scatter rate  $p$ .

```

Function  $rain(D, I, \beta, p)$ 
   $B = \text{betafunction}(\beta)$ 
   $D_{max} = -\ln(\frac{n}{I \cdot g}) / (2B)$ 
   $D_{rand} = \text{random.uniform}(D_{max})$ 
   $P_{lost} = 1 - \exp(-\beta \cdot D_{max})$ 
  foreach  $d \in D, d_m \in D_{max}, d_s \in D_{scatter}, d_r \in D_{rand}, p_l \in P_{lost}$  do
    if  $d_m < d$  then
      if  $p_l$  then
        pass;  $\triangleright$  point is lost, do nothing
      else if  $rand < p$  then
         $d = d_r;$   $\triangleright$  random scatter point
         $i = LN(\mu, \sigma^2);$   $\triangleright$  rain int.
      else
        pass
    else
       $i = i \cdot \exp(-\beta \cdot d);$   $\triangleright$  attenuate int.
  return  $D, I$ 
```

detail in Algorithm 1.

#### D. Network Architecture

For the de-noising of lidar images we adopt state-of-the-art CNN architectures for semantic segmentation of sparse point clouds. The proposed *WeatherNet* is an efficient variant of the *LiLaNet* introduced by [31]. In order to optimize the network for the de-noising task, we reduced the depth of the network given that the complexity of our task (3 classes) is reduced in comparison to full multi-class semantic segmentation (13 classes) [31], [37].

Additionally, we adapted the inception layer (Fig. 2) to include a dilated convolution to provide more information about the spatial vicinity by increasing the receptive field. Further, a dropout layer is inserted to increase the capability of generalization. The resulting network architecture is illustrated in Fig. 3. After optimization on the validation data set, the batch size is set to  $b = 20$ , the learning rate to  $\alpha = 4 \cdot 10^{-8}$  with a learning rate decay of 0.90 after every epoch. Adam solver is used to perform the training, with the suggested default values  $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$  and  $\epsilon = 10^{-8}$  [38].

## IV. DATASET

### A. Road Data Set

Creating a large-scale data set for training, validation and testing for the purpose of weather segmentation is very challenging, due to the fact that weather conditions are very unique and manual annotations are very difficult and complex. In order to re-use data sets, which were recorded under favorable weather conditions, like the data set from [31], we apply the developed data augmentation. Hence, we are able to utilize data sets recorded under favorable weather conditions with various traffic scenarios and roads types for the training of semantic weather segmentation.

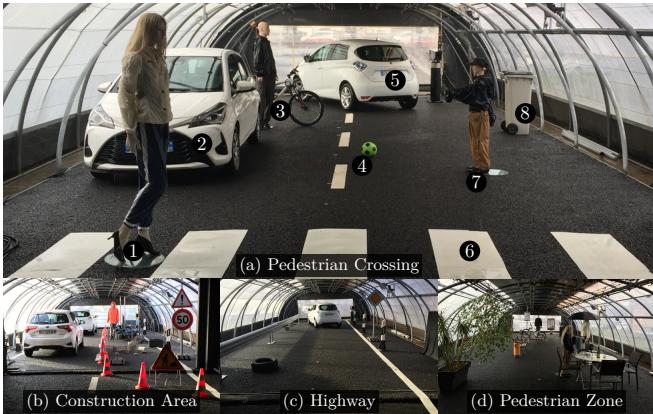


Fig. 4. Static setups in the chamber representing four complex real traffic situations. The upper picture shows a scene with a pedestrian (1) on a zebra crossing (6), a child (7) imitating chasing a ball (4) on the street, a parking car (2), a cyclist who pushes his bike across the street and a car (5) that turns left. In addition there is a garbage can (8) rightmost. The bottom pictures show several traffic scenarios with various different objects like a black tire as lost cargo, guardrails, cars, lane markings, reflector posts, traffic signs, a plant and pedestrian mannequins with and without umbrella.

### B. Climate Chamber Scenarios and Ground Truth Labels

Furthermore, a large data set with four very realistic road scenarios was recorded in CEREMA's climatic chamber to obtain constant and reproducible fog, rain and reference conditions [39], [40]. The data set will be published and will be available under the link in the abstract. Fig. 4 illustrates these static scenes with various objects, which are intended to provide a remarkably realistic representation. The climate chamber data set contains fog visibility ranges of 10 – 100 m and rain intensities of 15, 33 and 55 mm/h. The rainfall rate is close-loop controlled at a constant level. The fog recording was started at a visibility of 10m and recorded until 100m during a continuous dissipation and measurement of the actual visibility provided by a reference system of the climate chamber [40]. This procedure enables very accurate determination of the visibility and was repeated three times to generate more samples for each meteorological visibility.

As described in Section III, the reference data recorded without any rainfall or fog, enables the proposed *autolabeling* procedure. Therefore we obtained a large point-wise annotated data set without the error susceptibility of manually labeled weather data. The label set contains the classes 'clear' representing any point in the point cloud which is not caused by adverse weather, 'rain' for rain clutter and 'fog' for fog clutter respectively.

Based on the data set, we are able to analyze the weather influence on a point level, as illustrated in Fig. 5. The analysis shows that the lidar point cloud reflects the weather conditions in a very detailed level, since the number of points scattered by fog or rain is correlated with the visibility or rainfall rate. In addition, the results indicate that no points are being lost and therefore the sum of fog or rain and valid points is equivalent to the number of points in reference conditions. Thus the point cloud contains the information to estimate the meteorological visibility or rainfall rate by determining the number of weather induced scattering points.

As an increase in the rainfall rate does not necessarily results in an increase of scatter points, the rainfall rate cannot be estimated directly, but the extent of the degradation of the lidar sensor can be estimated. This information is incredibly valuable for an autonomous vehicle to adapt behavior to environmental conditions and sensor performance.

### C. Data Split

In total, the data set contains about 175,941 samples for training, validation and testing containing chamber (72,800) and road (103,141) scenes, which can be used thanks to augmentation. Details about the number of samples and class distributions are stated in Table I. In order to reduce time correlations between samples which were recorded in the climate chamber, each setup is only used in the training (Fig. 4d, 4c), validation (Fig. 4b) or test data split (Fig. 4a). In total we obtain a data split of about (60% – 15% – 25%) for training, validation and test.

To reduce an over-fitting to local dependencies and the scenes in the climate chamber, we cropped the image for training to a forward facing view of about 60° in the horizontal field of view. In addition, a subset of the data set, already used in [31], is added as samples in favorable weather conditions to increase the diversity and add road recordings while maintaining a balanced class distribution. Thus, the '*chamber & road*' data set contains 103,878 and 31,078 road samples without augmentation or adverse weather. Further, the amount and diversity of the training data can be increased by a manifold, because the augmentation enables the utilization of large-scale road data set, which results in 103,141 road samples and in 175,941 in total.

## V. EXPERIMENTS

As described in section III and IV, we obtained a large-scale data set recorded on public roads and in a dedicated climate chamber with different types of point-wise annotations. In this Section we describe several approaches to train the proposed *WeatherNet* in order to maximize the performance and analyze the benefit of weather augmentation, especially for the generalization to natural rainfall recorded on roads. We apply the **Intersection-over-Union (IoU)** metric for performance evaluation, according to the Cityscapes Benchmark Suite [31], [37]. An overview of all experiments and their results is given in Table I. In order to evaluate the influence of the weather augmentation in detail, we trained the network on three different data subsets with and without augmentation, defined as experiment 1, 2 and 3:

- 1) **Chamber:** only chamber data as baseline experiment.
- 2) **Chamber & Road:** Climate chamber data set and a subset of road data without any augmentation or adverse weather on roads.
- 3) **Chamber & Road with Augmentation:** Climate chamber data set and class balanced road data set without adverse weather, but with augmentation.

Note, all evaluations are based on the test data set from experiment 2, which contains autolabeled annotations and road data without fog, rain or augmentation. Table I shows

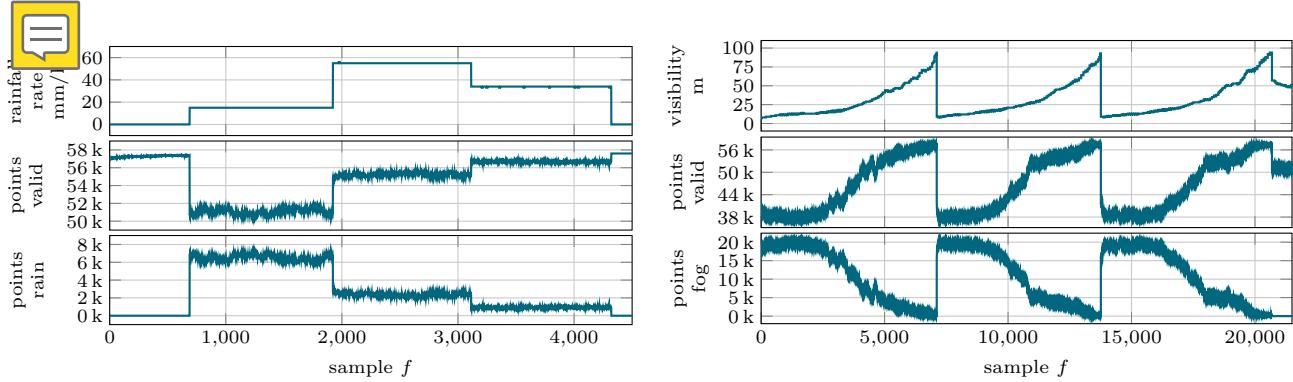


Fig. 5. Illustration of the meteorological visibility in  $m$  and rainfall rate in  $mm/h$  provided by the climate chamber, the number of valid points and the number of scatter points during rainfall or fog.

TABLE I

RESULTS ON THE TEST DATA SET. THE BEST PERFORMANCE IN TERMS OF IOU PER COLUMN IS PRINTED IN BOLD, THE OVERALL GREATEST IN BLUE.

Approach	IoU in % / Number of Samples												Runtime <sup>3)</sup> in ms	Parameter <sup>4)</sup> in Mio
	Experiment 1			Experiment 2			Experiment 3							
	Clear	Fog	Rain	Mean	Clear	Fog	Rain	Mean	Clear	Fog	Rain	Mean		
Samples Chamber <sup>1)</sup>	14,386	29,777	28,637	72,800	14,386	29,777	28,637	72,800	14,386	29,777	28,637	72,800	—	—
Samples Road <sup>1),2)</sup>	—	—	—	—	31,078	—	—	—	34,381	34,381	34,381	103,143	—	—
<i>DROR</i> [4]	<b>88.13</b>	6.94	7.37	34.15	88.13	6.94	7.37	34.15	88.13	6.94	7.37	34.15	100.00	4e <sup>-6</sup>
<i>RangeNet53</i> [32]	74.73	77.32	<b>91.22</b>	81.09	87.75	<b>86.46</b>	<b>94.23</b>	<b>89.48</b>	86.50	87.19	<b>94.36</b>	89.35	51.90	66.17
<i>RangeNet21</i> [32]	71.53	71.40	86.13	76.35	86.71	80.90	87.01	84.87	85.10	79.94	85.35	83.46	33.83	38.50
<i>LiLaNet</i> [31]	82.72	<b>79.57</b>	88.16	<b>83.48</b>	<b>91.60</b>	84.96	88.62	<b>88.39</b>	<b>93.85</b>	<b>88.74</b>	90.82	<b>91.14</b>	91.93	7.84
<i>WeatherNet</i> [Ours]	<b>91.65</b>	<b>86.40</b>	<b>89.29</b>	<b>89.11</b>	<b>90.89</b>	<b>85.15</b>	<b>88.84</b>	88.29	<b>93.35</b>	<b>88.81</b>	<b>90.92</b>	<b>91.03</b>	34.45	1.53

<sup>1)</sup> the column Mean states the total number the samples <sup>2)</sup> fog and rain are augmented samples <sup>3)</sup> on GeForce GTX 1080 Ti <sup>4)</sup> number of trainable parameters

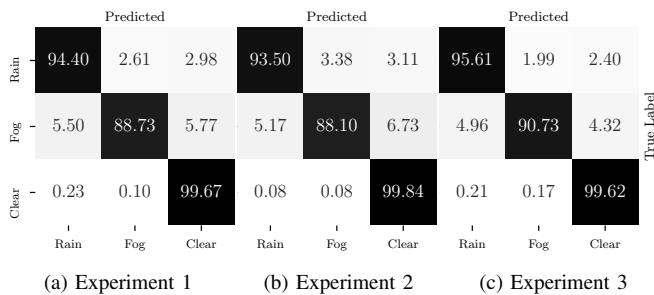


Fig. 6. Confusion matrix for *WeatherNet* segmentation result.

that the performance is significantly increased by using road data and the proposed weather augmentation. Besides validating the classes fog and rain only on chamber data, the usage of road data and the augmentation leads to an increase in the overall performance and per class IoU. This indicates that the network is able to identify weather influences in both domains and gains a general understanding of the scene.

The results of the baseline *DROR* filter indicates, that the local vicinity alone is not a proper feature to filter scatter points caused by dense water drops. The proposed CNN approach is outperforming *DROR* by an order of magnitude. The parameters for the *DROR* are taken from [4], except for the horizontal sensor resolution which is adapted to the utilized 'VLP32C'.

Furthermore, we compare our approach to the state-of-the-art semantic segmentation models *RangeNet21*, *RangeNet53* [32] and *LiLaNet* [7], which provide comparable results. Consequently, we prove that the basic idea of CNN-based weather segmentation and de-noising is valuable and sur-

passes geometrically based approaches. In addition, the proposed optimized *WeatherNet* is mostly outperforming the other CNNs, especially on the final experiment 3, and has a significantly lower number of trainable parameters and inference time. Thus, the network can be applied as pre-processing step.

The confusion matrices Fig. V show that mostly classes rain and fog are mixed up. Since fog and rain ultimately consist of water droplets and differ only in distribution, density and size of the water droplets, this is plausible. Moreover, lidar sensors are not designed to perceive this difference. For point cloud filtering these mix-ups are not important; a confusion is disadvantageous only with regard to classifying distinct weather conditions. Furthermore the augmentation leads to a significant decrease in confusion between rain and fog.

#### A. Qualitative Results on Dynamic Chamber Data

Qualitative results on challenging dynamic scenes are presented in this section. Whereas there is no ground truth data available due to the fact that the very same dynamic scenes cannot be recorded under two different weather conditions. Hence, our proposed auto-labeling procedure cannot be applied. Nevertheless, Fig. 1, 7 and 8 show that our approach is able to handle dynamic scenes and gives remarkable filter results. The de-noised point cloud reveals a pedestrian and a cyclist (highlighted by black boxes) directly in front of the ego-vehicle, who almost disappear in the scatter points of the haze (Fig. 1c, 7g). Although the evaluated performance of *RangeNet53* and *WeatherNet* are comparable (Table I), the qualitative results show that *RangeNet53* does preserve fine

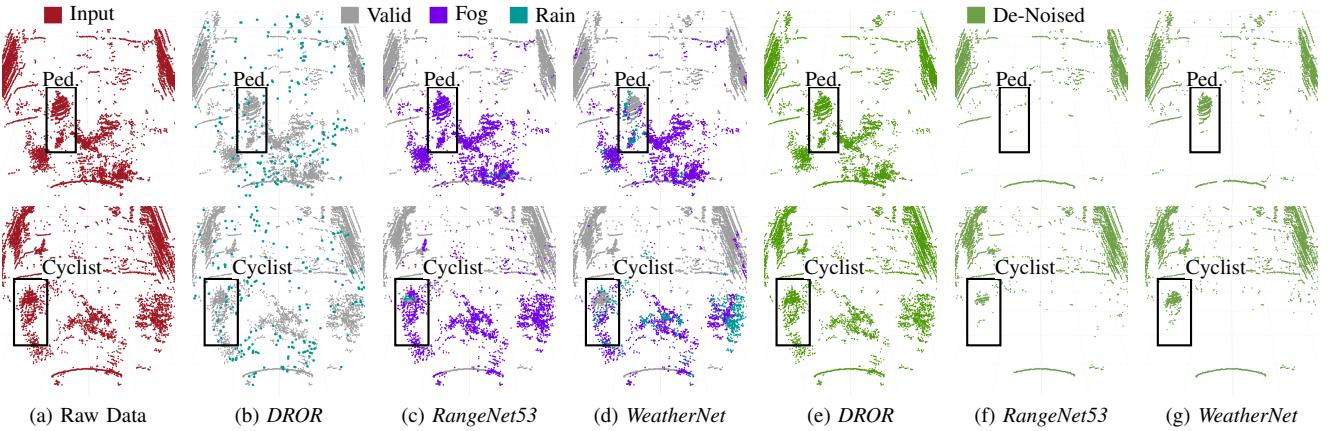


Fig. 7. De-noising results shown on a snapshot with two dynamic objects in dense fog at 30m visibility. The color coding is similar to Fig. 1. For the baseline *DROR* filter all points are colored as rain points, since no distinction between rain and fog is possible. In addition the segmentation and de-noising results from *RangeNet53* and *WeatherNet* are given. The cyclist and pedestrian, which are barely recognizable in the scene, are highlighted in a black box. Note that the pedestrian and cyclist remains after filtering by *WeatherNet* while discarding the fog clutter.

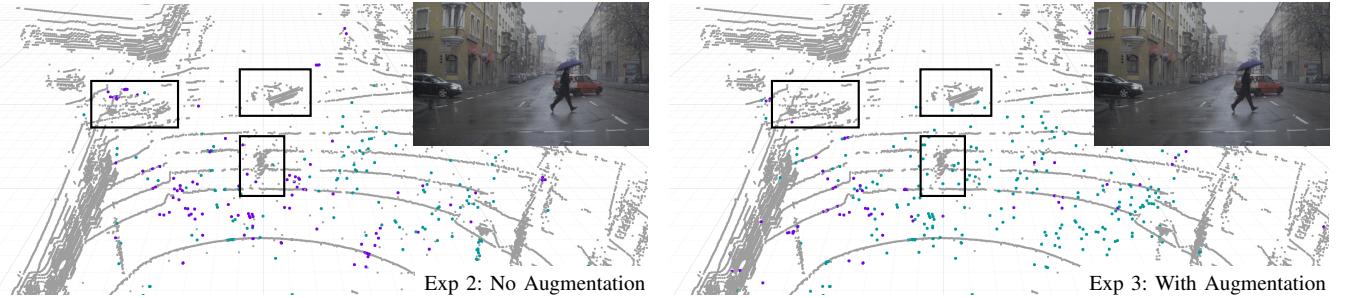


Fig. 8. *WeatherNet* segmentation results for road data recorded under light rainfall. The result shows, that the approach is able to generalize the noise pattern and predicts well results for recordings during natural rainfall. Note that the training data set neither contains natural rain nor fog scenes on roads. The color coding is similar to Fig. 1 and the objects shown in the camera image are highlighted with a black box. The training with augmentation (right) leads to a better segmentation result in terms of number of detected raindrops and less false negatives for object detection (e.g. left car).

structures and edges of small objects (Fig. 7f), as most parts of the cyclist and pedestrian are filtered. Whereas, *WeatherNet* is able to distinguish between the pedestrian/cyclist and scatter points (Fig. 7g).

A filter algorithm based on the spatial distribution of the point cloud, as shown in Fig. 7e is not able to filter the noise in this scenario, since the fog points are similarly densely distributed as those of real objects. Nevertheless, the cyclist can be recognized slightly better, due to the ability to filter single scattering points. Note that the DROR also filters various single points at greater distances, which are not caused by the weather.

Another benefit of our *CNN*-approach is the capability of detecting the weather condition by means of lidar point clouds. As shown in Fig. 5, the number of scatter points caused by fog is correlated with the meteorological visibility, hence by utilizing the result of our weather segmentation, the visibility could be estimated. Moreover, the level of degradation of the lidar sensor could be estimated by taking the ratio of scatter to valid points into account.

### B. Qualitative Results on Dynamic Road Data

Additionally, the proposed approach is able to work on a point cloud corrupted by natural rainfall recorded on roads.

In Fig. 8 shows a key frame where a pedestrian is crossing the street and multiple cars are passing by. Despite our algorithm is only trained on climate chamber data, complemented with augmented real-world data, it shows well performance in a real world scenario with light rain and proofs the generalization to a complete different scenario, see Fig. 8.

## VI. CONCLUSION

We presented a *CNN*-based approach for point cloud weather segmentation as an essential pre-processing step for lidar-based environment perception to distinguish between scatter points from adverse weather and valid points from solid objects. As opposed to previous approaches that analyze the statistics of the local spatial vicinity of individual points, we opted for a learning-based approach that involves a global understanding of a traffic scene as a whole to estimate the validity of point-level measurements. The issue of requiring annotated ground truth data for our approach is significantly alleviated by our proposed data augmentation strategy. Our quantitative and qualitative results demonstrate the superior performance of our *CNN*-based approach over state-of-the-art, while being very efficient at the same time. For more qualitative results on dynamic scenes we refer to our data set page (link in the abstract) and the supplementary video.

This work was done in cooperation with the Dense Project, contract no. 692449, of the European Union under the H2020 ECSEL programme. We thank Mario Bijelic and Tobias Gruber for enabling this work by recording the data set.

## REFERENCES

- [1] M. Bijelic, T. Gruber, and W. Ritter, "A Benchmark for Lidar Sensors in Fog: Is Detection Breaking Down?" in *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2018, pp. 760–767.
- [2] R. Heinzler, P. Schindler, J. Seekircher, W. Ritter, and W. Stork, "Weather Influence and Classification with Automotive Lidar Sensors," in *2019 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2019, pp. 1527–1534.
- [3] T. G. Phillips, N. Guenther, and P. R. McAree, "When the dust settles: The four behaviors of LIDAR in the presence of fine airborne particulates," *Journal of field robotics*, vol. 34, no. 5, pp. 985–1009, 2017.
- [4] N. Charron, S. Phillips, and S. L. Waslander, "De-Noising of Lidar Point Clouds Corrupted by Snowfall." *2018 15th Conference on Computer and Robot Vision (CRV)*, pp. 254–261, may 2018.
- [5] M. Kutila, P. Pykkönen, H. Holzhüter, M. Colomb, and P. Duthon, "Automotive LiDAR performance verification in fog and rain," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, nov 2018, pp. 1695–1701.
- [6] A. Djuricic and B. Jutzi, "Supporting UAVs in low visibility conditions by multiple-pulse laser scanning devices," *The international archives of the photogrammetry, remote sensing and spatial information sciences*, vol. 1, p. W1, 2013.
- [7] F. Piewak, P. Pinggera, M. Enzweiler, D. Pfeiffer, and M. Zöllner, "Improved Semantic Stixels via Multimodal Sensor Fusion," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2019.
- [8] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proceedings of the IEEE International Conference on Computer Vision*. IEEE, 1998, pp. 839–846.
- [9] S. Rönbäck and Å. Wernersson, "On filtering of laser range data in snowfall," *2008 4th International IEEE Conference Intelligent Systems, IS 2008*, vol. 3, pp. 1733–1739, 2008.
- [10] Y. Li, J. Li, L. Wang, J. Zhang, D. Li, and M. Zhang, "A weighted least squares algorithm for time-of-flight depth image denoising," *Optik*, 2014.
- [11] P. Jenke, M. Wand, M. Bokeloh, A. Schilling, and W. Straßer, "Bayesian point cloud reconstruction," *Computer Graphics Forum*, 2006.
- [12] L. Chen, H. Lin, and S. Li, "Depth image enhancement for Kinect using region growing and bilateral filter," *Proceedings - International Conference on Pattern Recognition*, no. November, pp. 3070–3073, 2012.
- [13] J. Shen and S. C. S. Cheung, "Layer depth denoising and completion for structured-light RGB-D cameras," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2013.
- [14] A. V. Le, S. W. Jung, and C. S. Won, "Directional joint bilateral filter for depth images," *Sensors (Switzerland)*, vol. 14, no. 7, pp. 11 362–11 378, 2014.
- [15] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *Proceedings - IEEE International Conference on Robotics and Automation*, 2011.
- [16] M. Bijelic, F. Mannan, T. Gruber, W. Ritter, K. Dietmayer, and F. Heide, "Seeing Through Fog Without Seeing Fog: Deep Sensor Fusion in the Absence of Labeled Training Data," 2019. [Online]. Available: <http://arxiv.org/abs/1902.08913>
- [17] A. Filgueira, H. González-Jorge, S. Lagüela, L. Díaz-Vilariño, and P. Arias, "Quantifying the influence of rain in LiDAR performance," *Measurement: Journal of the International Measurement Confederation*, vol. 95, pp. 143–148, 2017.
- [18] S. Hasirlioglu, A. Kamann, I. Doric, and T. Brandmeier, "Test methodology for rain influence on automotive surround sensors," in *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*. IEEE, 2016, pp. 2242–2247.
- [19] S. Hasirlioglu, A. Riener, W. Ruber, and P. Wintersberger, "Effects of exhaust gases on laser scanner data quality at low ambient temperatures," in *28th IEEE Intelligent Vehicles Symposium*. IEEE, 2017, pp. 1708–1713.
- [20] S. Hasirlioglu, I. Doric, A. Kamann, and A. Riener, "Reproducible Fog Simulation for Testing Automotive Surround Sensors," in *2017 IEEE 85th Vehicular Technology Conference (VTC Spring)*, 2017, pp. 1–7.
- [21] M. Kutila, P. Pyykonen, W. Ritter, O. Sawade, and B. Schaufele, "Automotive LIDAR sensor development scenarios for harsh weather conditions," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, 2016, pp. 265–270.
- [22] J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, and T. Aila, "Noise2Noise: Learning Image Restoration without Clean Data," no. 3, 2018. [Online]. Available: <http://arxiv.org/abs/1803.04189>
- [23] R. H. Rasshofer, M. Spies, and H. Spies, "Influences of weather phenomena on automotive laser radar systems," *Advances in Radio Science*, vol. 9, pp. 49–60, 2011.
- [24] A. U. Shamsudin, K. Ohno, T. Westfechtel, S. Takahiro, Y. Okada, and S. Tadokoro, "Fog removal using laser beam penetration, laser intensity, and geometrical features for 3D measurements in fog-filled room," *Advanced Robotics*, vol. 30, no. 11–12, pp. 729–743, 2016.
- [25] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.
- [26] Gaurav Pandey, James R. McBride, and Ryan M. Eustice, "Ford campus vision and lidar data set," *International Journal of Robotics Research*, vol. 30, no. 13, pp. 1543–1552, 2011.
- [27] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, and Q. Xu, "nuScenes : A multimodal dataset for autonomous driving," 2019.
- [28] M. Carfagni, R. Furferi, L. Governi, M. Servi, F. Uccheddu, and Y. Volpe, "On the Performance of the Intel SR300 Depth Camera: Metrological and Critical Characterization," *IEEE Sensors Journal*, 2017.
- [29] O. Schall, A. Belyaev, and H. P. Seidel, "Robust filtering of noisy scattered point data," in *Point-Based Graphics, 2005 - Eurographics/IEEE VGTC Symposium Proceedings*, 2005.
- [30] B. Wu, A. Wan, X. Yue, and K. Keutzer, "SqueezeSeg: Convolutional Neural Nets with Recurrent CRF for Real-Time Road-Object Segmentation from 3D LiDAR Point Cloud," oct 2017. [Online]. Available: <http://arxiv.org/abs/1710.07368>
- [31] F. Piewak, P. Pinggera, M. Schäfer, D. Peter, B. Schwarz, N. Schneider, M. Enzweiler, D. Pfeiffer, and M. Zöllner, "Boosting LiDAR-based semantic labeling by cross-modal training data generation," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11134 LNCS, 2019, pp. 497–513.
- [32] A. Milioto, I. Vizzo, J. Behley, and C. Stachniss, "RangeNet++: Fast and Accurate LiDAR Semantic Segmentation," 2019. [Online]. Available: <https://github.com/PRBonn/lidar-bonnetal>.
- [33] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3D object detection network for autonomous driving," in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017.
- [34] B. Yang, W. Luo, and R. Urtasun, "PIXOR: Real-time 3D Object Detection from Point Clouds," 2018.
- [35] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "PointPillars: Fast Encoders for Object Detection from Point Clouds," dec 2018. [Online]. Available: <http://arxiv.org/abs/1812.05784>
- [36] B. Li, "3D fully convolutional network for vehicle detection in point cloud," in *IEEE International Conference on Intelligent Robots and Systems*, 2017.
- [37] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The Cityscapes Dataset for Semantic Urban Scene Understanding," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016.
- [38] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," pp. 1–15, 2014. [Online]. Available: <http://arxiv.org/abs/1412.6980>
- [39] T. Gruber, M. Bijelic, F. Heide, W. Ritter, and K. Dietmayer, "Pixel-Accurate Depth Evaluation in Realistic Driving Scenarios," jun 2019. [Online]. Available: <https://arxiv.org/abs/1906.08953>
- [40] M. Colomb, K. Hirech, P. André, J. J. Boreux, P. Lacôte, and J. Dufour, "An innovative artificial fog production device improved in the European project "FOG"," *Atmospheric Research*, vol. 87, no. 3–4, pp. 242–251, 2008.