Gated3D: Monocular 3D Object Detection From Temporal Illumination Cues (Supplemental Document)

Frank Julca-Aguilar[1*] Jason Taylor[1*] Mario Bijelic[2,3] Fahim Mannan[1] Ethan Tseng[3] Felix Heide[1,3]

[1]Algolux [2]Mercedes-Benz AG [3]Princeton University

In this Supplemental Document we present additional information and results in support of the main manuscript. Section 1 describes results using the KITTI evaluation metrics and shows additional qualitative examples of our method as well as several baseline methods. Section 2 presents a detailed ablation study of our proposed architecture. Section 3 provides additional insight and experiments on the generalization to different objects, orientations and positions. Section 4 provides implementation details. Section 5 describes our gated imaging sensor setup. Section 6 provides further details on our Gated3D dataset annotation and capture procedures. Please see the *additional Supplemental Video* for qualitative results of the proposed approach in challenging conditions.

1. KITTI Evaluation Metrics and Qualitative Comparisons

Table 1 gives additional results of our method and the baselines described in our main manuscript, using the KITTI evaluation metrics. Overall the proposed approach mostly outperforms the baselines in the Easy, Moderate and Hard categories, and narrowly trails the leading result in the few cases where another method has the best result. In particular, Gated3D consistently outperforms the baselines in the Hard category. This validates that our model is able to robustly detect small and occluded objects, and is consistent with the results in our main manuscript, which show larger improvements for pedestrians at all distance ranges as well as for cars in the furthest range considered of 50-80m.

Figure 1 shows additional qualitative results of our method compared to the baseline approaches. We can see that our proposed method detects objects with more accurate localization in both the image and birds eye view spaces. Our method especially outperforms other methods on pedestrians and objects that are far away from the camera.

2. Ablation Study

Table 2 shows the impact of some of the main components of our proposed *Gated3D* model. The results of our complete architecture are compared against variants of the model without the attention layers, with a smaller backbone (ResNet-50 FPN), and without the frustum-based depth prediction (regression of absolute depth). Since the attention layers serve as a learned pooling between the convolutional and fully-connected layers, we test replacing attention with max-pooling, mean-pooling and flattening. Note that for the version with flattening in place of the attention layer, the RoIAlign crop size is reduced from 28 to 7 due to memory constraints.

Due to the relatively small size of the test set and label noise caused by sensor synchronization issues, it is best to consider the overall performance across a full row of Table 2 or both daytime and nighttime results for each class rather than attempt a fine-grained analysis.

While attention layers improve the results overall for both object classes, the impact is larger for pedestrian detection. This matches the intuition that attention helps separate object features from background or occlusion features in the ROI crops. Pedestrian crops have proportionally more background or occlusion on average, both because they are smaller and less rectangular than cars, and because the 2D detection performance is lower, leading to more poorly-fit region proposals to the 3D detection network. Additionally, as expected, we see that attention does not consistently impact the 2D metrics as it is only used in the 3D detection network.

[*]indicates equal contribution.

Figure 1: Qualitative comparison against baseline methods on the captured dataset. Bounding boxes from the proposed method are tighter and more accurate than the state-of-the-art methods. The BEV lidar overlays show that our method offers more accurate depth and orientation than the baselines. The advantages of our method are most noticeable for pedestrians, as cars are easier to detect for other methods due to being large and specular (please zoom in on the electronic version for details).

Table 1: Object detection performance over Gated3D dataset (test split). Similar to the results using distance range-based metrics, our method outperforms monocular and stereo methods, as well as pseudo-lidar based methods, over the three KITTI categories (Easy, Moderate and Hard).

(a) Average Precision on *Car* class.

Method	Modality	Daytime Images									Nighttime Images								
		2D object detection			3D object detection			BEV detection			2D object detection			3D object detection			BEV detection		
		Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard
PointPillars [5]	Lidar	85.74	77.91	76.74	85.45	76.20	72.73	86.19	76.68	73.77	86.41	79.42	80.30	86.43	77.87	76.84	86.66	78.01	77.05
M3D-RPN [2]	RGB	89.39	80.29	71.33	32.13	31.21	26.99	33.28	32.46	27.59	89.58	80.42	71.45	30.70	**30.65**	26.40	30.99	31.02	26.76
Stereo-RCNN [6]	Stereo	90.13	88.67	79.88	31.54	26.68	23.10	32.50	28.21	24.15	90.24	88.79	80.13	28.05	27.20	23.04	32.04	28.52	24.05
Pseudo-Lidar	Gated	90.56	90.43	89.98	18.59	17.94	15.59	19.30	18.73	16.41	90.74	90.63	90.23	27.09	26.92	26.34	33.18	28.63	28.36
Pseudo-Lidar++ [9]	Gated	90.55	90.43	89.97	18.59	18.22	15.97	19.76	19.88	20.07	90.74	90.60	90.19	26.31	26.89	26.82	29.96	27.90	28.34
PatchNet [8]	Gated	90.87	90.79	90.39	13.41	13.77	13.97	16.18	14.30	14.55	90.87	**90.87**	90.47	17.91	15.29	15.51	19.18	18.74	16.58
Gated3D	Gated	**90.91**	**90.90**	**90.64**	**36.15**	**32.04**	**28.93**	**38.03**	**35.33**	**30.52**	**90.88**	90.75	**90.53**	**35.58**	29.15	**28.58**	**36.34**	**29.87**	**29.17**
Gated3D w/ dense depth	Gated	**90.91**	**90.88**	81.79	34.97	28.8	27.69	35.4	29.12	28.1	**90.86**	90.75	81.67	**35.24**	28.72	27.57	35.98	**29.40**	28.41

(b) Average Precision on *Pedestrian* class.

Method	Modality	Daytime Images									Nighttime Images								
		2D object detection			3D object detection			BEV detection			2D object detection			3D object detection			BEV detection		
		Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard
PointPillars [5]	Lidar	48.70	47.48	48.01	45.55	43.12	42.75	47.16	44.85	44.60	44.29	42.19	41.74	41.99	38.29	37.49	42.89	39.77	38.77
M3D-RPN [2]	RGB	75.14	67.83	67.01	17.16	16.70	14.70	18.78	18.19	18.29	73.04	66.22	65.57	12.02	11.43	11.21	15.17	13.05	12.59
Stereo-RCNN [6]	Stereo	85.90	84.17	77.75	30.92	28.37	25.43	31.88	30.78	28.95	85.06	77.65	76.90	28.54	25.00	24.55	29.76	28.72	25.52
Pseudo-Lidar	Gated	88.62	88.31	87.87	4.91	3.90	3.91	9.45	9.68	9.85	89.19	87.91	87.53	6.68	6.62	6.71	12.50	12.60	12.66
Pseudo-Lidar++ [9]	Gated	88.49	88.18	87.93	4.44	4.64	4.60	9.35	9.55	9.61	88.90	87.98	87.63	5.91	4.66	4.76	11.48	12.23	12.30
PatchNet [8]	Gated	89.78	89.30	88.58	21.43	20.94	20.62	26.74	25.81	22.75	89.86	88.80	**87.91**	13.26	12.54	12.66	17.94	15.22	15.36
Gated3D	Gated	**90.20**	**89.79**	**89.51**	28.60	27.85	27.34	34.00	29.61	29.39	89.60	**89.35**	81.04	**29.95**	**28.45**	**28.33**	**31.12**	**29.86**	**29.32**
Gated3D w/ dense depth	Gated	**90.17**	81.31	81.17	**31.12**	**30.6**	**30.6**	**36.52**	**31.74**	**31.50**	**90.09**	81.34	81.25	27.53	26.85	22.74	28.91	28.41	28.46

Replacing the backbone of the 2D detection network with a ResNet-50 FPN leads to a relatively larger performance drop than replacing the attention layers. This may be due to a lack of hyperparameter tuning on this variation of the architecture, as this ablation represents a more substantial change to the network than the attention replacements. It may be possible to achieve results closer to the full architecture by adjusting the loss weights and learning rate schedule.

The last row of Table 2 shows results of the proposed architecture with RGB images as input. Results show that our model obtains competitive results with RGB images only but perform better with gated images, especially at night and in the detection of pedestrians.

Overall, the proposed frustum segment-based depth prediction has the largest impact on 3D and BEV detection metrics of all the ablations considered. This is expected, as directly regressing the absolute depth of the object implies a larger range of possible values and greater variance on the loss. This may be why the 2D detection metrics are also lower, as larger depth loss values may interfere with training the other losses. Additionally, regressing the depth directly means that any dataset imbalances have a larger impact on training and the network may learn to hedge its prediction towards the mean depth. This is less of an issue with the frustum segment-based depth prediction since the predicted offset is in a smaller range and is more likely to be symmetric. We did attempt to mitigate these issues by hyperparameter tuning with different loss weights and more aggressive gradient clipping, but were unable to improve the results for this variant of the model. As such, we conclude that the frustum segment-based depth prediction is a critical component of the proposed model.

3. Generalization Across Object Types, Dimensions and Orientations

Our approach to robust 3D object detection consists of introducing general priors based on the object statistics and camera calibration, and letting the network learn additional information from data. This allows our method to generalize to different object types, dimensions and orientations. In this subsection, we show how our model can be robustly extended to recognize other objects than the ones covered in the main manuscript, as well as its generalization to different orientations and poses.

In order to extend the method to the detection of large vehicles such as trucks or vans, our approach only requires calculating the corresponding object dimension statistics and adding classification heads for the target objects, as is typically done in 2D object detection. This approach does not require any additional annotation process, as the statistics can be computed from 3D labels. To illustrate this approach we trained our Gated3D model on the KITTI dataset, using the Car, Pedestrian, Truck, Van and Cyclist labels. In this experiment, we use the training-validation split as proposed by Chen et al. [3]. Figure 2 shows some output examples of this extension. *Although our model has not been optimized to have RGB as input*, results in

Table 2: Ablations of our proposed model.

(a) Average Precision on *Car* class.

Gated3D variant	Daytime Images									Nighttime Images								
	2D object detection			3D object detection			BEV detection			2D object detection			3D object detection			BEV detection		
	0-30 m	30-50 m	50-80 m	0-30 m	30-50 m	50-80 m	0-30 m	30-50 m	50-80 m	0-30 m	30-50 m	50-80 m	0-30 m	30-50 m	50-80 m	0-30 m	30-50 m	50-80 m
COMPLETE	90.78	90.55	**90.91**	**52.15**	**28.31**	14.85	52.31	**29.26**	15.02	90.84	81.82	**90.33**	51.42	**25.73**	12.97	**53.37**	**29.13**	13.12
WITHOUT ATTENTION: MAX POOL	90.75	81.77	90.85	39.77	19.95	9.22	47.47	21.75	12.17	90.87	81.82	81.74	44.70	21.62	**13.35**	47.39	22.59	**15.86**
WITHOUT ATTENTION: MEAN POOL	**90.91**	90.58	90.68	49.81	21.47	13.03	50.77	22.01	14.09	**90.91**	81.82	81.82	40.83	21.62	9.19	47.34	23.15	9.67
WITHOUT ATTENTION: FLATTEN	90.71	**90.61**	90.62	48.16	20.65	8.61	50.17	21.92	8.92	90.80	81.82	81.67	48.10	25.32	13.10	49.59	28.30	14.50
WITH RESNET-50 FPN	89.04	81.68	89.44	43.65	15.23	7.28	44.87	18.10	7.45	88.30	81.62	81.30	27.33	13.78	8.25	28.42	14.42	8.58
WITHOUT FRUSTUM SEGMENT	90.65	79.74	80.19	18.43	11.45	10.74	22.58	13.08	12.16	90.60	72.36	71.64	16.73	8.74	10.85	17.97	9.99	11.19
WITH RGB INPUT	90.45	90.61	90.55	51.80	26.51	**18.57**	**53.44**	26.51	**18.91**	90.81	**90.83**	81.39	**51.50**	20.06	9.08	52.25	22.61	10.82

(b) Average Precision on *Pedestrian* class.

Gated3D variant	Daytime Images									Nighttime Images								
	2D object detection			3D object detection			BEV detection			2D object detection			3D object detection			BEV detection		
	0-30 m	30-50 m	50-80 m	0-30 m	30-50 m	50-80 m	0-30 m	30-50 m	50-80 m	0-30 m	30-50 m	50-80 m	0-30 m	30-50 m	50-80 m	0-30 m	30-50 m	50-80 m
COMPLETE	89.72	81.47	**86.73**	**50.94**	**20.59**	**14.14**	53.26	22.15	**16.51**	81.52	81.23	80.18	**48.53**	**23.99**	**14.98**	**49.82**	25.57	15.46
WITHOUT ATTENTION: MAX POOL	90.31	81.25	79.26	34.77	15.10	9.85	44.44	21.51	10.77	81.73	81.35	79.64	33.42	14.00	7.26	41.98	23.62	13.95
WITHOUT ATTENTION: MEAN POOL	89.96	**89.40**	79.11	37.67	15.34	11.65	**53.75**	22.67	15.82	81.56	81.32	80.55	28.80	16.97	9.75	47.64	25.54	12.20
WITHOUT ATTENTION: FLATTEN	90.44	80.98	78.92	36.69	18.09	6.38	45.50	**23.34**	11.65	81.69	81.34	**80.77**	34.34	20.38	13.61	38.45	**26.16**	**17.28**
WITH RESNET-50 FPN	81.20	71.01	60.10	37.16	13.48	10.24	39.64	14.45	10.57	80.11	78.38	72.99	23.78	12.26	4.93	27.26	15.59	7.22
WITHOUT FRUSTUM SEGMENT	89.57	81.08	75.14	21.55	15.33	10.77	48.29	16.14	13.07	80.89	81.05	69.01	23.32	17.80	6.19	38.04	18.62	8.01
WITH RGB INPUT	**90.47**	88.42	80.03	41.97	10.34	5.68	43.21	12.78	5.79	**89.64**	**81.41**	75.54	29.70	13.88	3.29	31.47	15.33	4.10

Figure 2 show that our method can be effectively extended to recognize other kinds of objects and furthermore, can potentially be used for monocular 3D object detection from RGB images. Although we note that the number of samples in the Van and Truck KITTI categories is not large enough to compute definite metrics, this preliminary extension shows encouraging results regarding the generalization capability of our proposed model.

Our frustum-based depth estimation can be considered as a generalized anchor-based approach for object depth estimation. The method uses object dimension statistics and the camera calibration to estimate an approximated depth, and trains a network to compute the offset between this estimation and the actual depth. In order to effectively train the network, this offset is scaled by a frustum region (as defined in Equation (7) in the main manuscript) that encodes the uncertainty at near and long distances. We note that this approach does not require the objects to be inside the frustum segment in order to be recognized. The model also does not require the projected 2D height to correspond exactly to the object height for the anchors. Similar to anchors in 2D object detection networks, where no anchor matches exactly the target 2D objects dimensions, our network learns the offset adjustments between the depth anchor and the target object depth from data. To illustrate the robustness of our model to different object orientations and poses, please see the supplemental video.

4. Additional Network Details

As described in our main manuscript, the proposed *Gated3D* network is composed of a 2D object detection network that generates 2D detection candidates, and a 3D object detection network that predicts 3D object parameters for each candidate generated in the first step. The 2D object detection network uses a ResNet-101 FPN backbone and is initialized from a Faster RCNN checkpoint pretrained on COCO [7]. RoIAlign is used to extract features from both the 2D backbone features and gated input images, with a pooling resolution of 28×28.

Table 3 shows the architecture of our 3D object detection network.

We train our network for up to $50,000$ steps with a batch size of 2 with 64 candidate detections per image and use the 3D average precision on the validation set to control early stopping. The entire network is trained on full-resolution images using stochastic gradient descent with a learning rate of 0.0016 and momentum of 0.9. Gradient clipping is used to limit the norm of the gradients to 1.0. We define the loss weights for the depth, orientation, and dimensions as 2.0, 0.1 and 0.02, respectively.

During training we randomly sample data augmentation operations from the following set, each with an independent 75% chance of being applied: saturation ± 0.15, brightness ± 0.15, contrast ± 0.15, and additive Gaussian noise of mean 0 and standard deviation 0.001. The additive Gaussian noise is randomly sized between $1/4$ and $1/40$ the image size and then resampled with bilinear interpolation to match the input image size. This is done to perturb the inputs over a range of spatial frequencies, since the saturation, brightness and contrast augmentations are all low-frequency perturbations.

Additionally, we augment the RoIAlign crops with a probability of 50% independently for each of the backbone feature and gated image crops. The augmentation randomly downsamples the crops to a size between 22×22 and 27×27 (inclusive) and then upsamples back to 28×28. This simulates the blurring inherent in the features and inputs of far-away objects and

Figure 2: Generalization Experiment: Although our model has not been optimized for RGB data, it generalizes to KITTI RGB data, see text. We present qualitative results of our Gated3D model applied to RGB from KITTI validation (top four rows) and test (bottom four rows) sets. 3D boxes are color-coded as green=Car, yellow=Van, purple=Truck, red=Pedestrian, white=Cyclist. BEV shows an area in the [0m, 100m] and [-40, +40m] ranges in the Z and X camera coordinates, respectively. Our model effectively recognizes objects of varying dimensions and types, i.e. Cars, Vans, and Trucks.

Table 3: 3D object detection network architecture. In the table, "conv-k(a)-s(b)-ReLU" represents a convolution layer with an $a \times a$ kernel window, using stride b, followed by a ReLU activation function. Similarly, "res-d(d)-k(a)-s(b)-BN-ReLU" indicates repeating residual layers of depth d with batch normalization prior to the ReLU activation. "RoIAlign-size(a)-scale(b)-r(c)" represents an RoIAlign layer with $a \times a$ pooling resolution, using a spatial scale of 2^{-b}, and a sampling ratio of c. The RoIAlign layers use the output of the 2D object detector to crop regions.

3D Object Detection Network		
Layer Name	Type	Channels
slice0_input	0th Gated Slice	1
slice0_RoI	RoIAlign-size28-scale0-r2	1
slice0_conv0	conv-k3-s1-ReLU	16
slice0_conv1	conv-k3-s1-ReLU	32
slice0_conv2	conv-k3-s1-ReLU	32
slice1_input	1st Gated Slice	1
slice1_RoI	RoIAlign-size28-scale0-r2	1
slice1_conv0	conv-k3-s1-ReLU	16
slice1_conv1	conv-k3-s1-ReLU	32
slice1_conv2	conv-k3-s1-ReLU	32
slice2_input	2nd Gated Slice	1
slice2_RoI	RoIAlign-size28-scale0-r2	1
slice2_conv0	conv-k3-s1-ReLU	16
slice2_conv1	conv-k3-s1-ReLU	32
slice2_conv2	conv-k3-s1-ReLU	32
P_2	P_2 layer of ResNet-101 FPN backbone	256
P_2_RoI	RoIAlign-size28-scale2-r2	256
P_3	P_3 layer of ResNet-101 FPN backbone	256
P_3_RoI	RoIAlign-size28-scale8-r2	256
P_4	P_4 layer of ResNet-101 FPN backbone	256
P_4_RoI	RoIAlign-size28-scale16-r2	256
P_5	P_5 layer of ResNet-101 FPN backbone	256
P_5_RoI	RoIAlign-size28-scale32-r2	256
concat	Concat(slice*_conv2, P_*_RoI)	352
fusion	res-d5-k3-s1-BN-ReLU	352
attention_conv0	conv-k3-s1-ReLU	352
attention_conv1	conv-k3-s1-ReLU	352
attention_conv2	conv-k3-s1-ReLU	352
attention_conv3	conv-k3-s1-ReLU	352
attention_softmax	Softmax	352
z_fc1	Linear	1024
z_fc2	Linear	1024
box3d_pred	Linear	8

we found it to improve the 3D detection results for objects in the 50-80m range.

During inference we set the IoU threshold for non-maximum suppression to 0.7 and use the top 300 proposals for detection. From these, the detections with object confidence greater than or equal to 0.5 are selected. The frustum extent factor k is set to 1.

Laser		
Laser power	P_{laser}	500 W
Wavelength	λ	808 nm
Horizontal field of illumination	θ_H	24°
Vertical field of illumination	θ_V	8°

Camera		
Pixel pitch	ρ	10 µm
Aperture	F_{num}	1.2
Optical transmission	τ_{optics}	0.64
Focal length	f	23 mm
Horizontal field of view	θ_H	31.1°
Vertical field of view	θ_V	17.8°
Resolution		1280 px × 720 px

Table 4: Laser and camera specifications of the BrightwayVision BrightEye.

5. Gated Image Formation

In this section, we explain in more detail the gated image formation model that is based on the range-intensity-profile $C(r)$ given by

$$C\left(r\right) = \int_{-\infty}^{\infty} g\left(t-\xi\right) p\left(t-\frac{2r}{c}\right) \beta\left(r\right) \mathrm{d}t. \tag{1}$$

Both the temporally-modulated camera gate g and the laser pulse profile p are assumed to be rectangle-shaped with a duration given in Table 5. Eq. (1), ξ describes the delay between the start of illumination and the start of exposure, and $\beta\left(r\right)$ models atmospheric effects occurring not at object surfaces given by

$$\beta(r) = \frac{P_{\text{laser}}\tau_{\text{optics}}}{4\pi r^2 \tan\left(\frac{\theta_H}{2}\right)\tan\left(\frac{\theta_V}{2}\right)} \frac{\rho^2}{F_{\text{num}}^2} \frac{\lambda}{hc} e^{-2\gamma r} \tag{2}$$

where P_{laser} is the laser power, τ_{optics} is the optical transmission, θ_H and θ_V are the horizontal/vertical field of illumination, ρ is the pixel pitch, F_{num} is the aperture, λ is the wavelength, h is the Planck constant and γ is the atmospheric attenuation coefficient. Assuming now a scene with a dominating lambertian reflector with albedo α at distance $\tilde{r}$, the measurement for each pixel location is obtained by

$$z = \alpha C(\tilde{r}) + \eta_p\left(\alpha C(\tilde{r})\right) + \eta_g, \tag{3}$$

where η_p describes the Poissonian photon shot noise and η_g the Gaussian read-out noise [4]. In order to improve the SNR, multiple pulses are required before read-out as described in Table 5. We use a larger number of pulses to improve SNR at larger distances. Details on the laser and camera specifications can be found in Table 4.

For this work, we use the range-intensity-profiles defined by the gating parameters in Table 5 and as illustrated in Figure 3. The range-intensity-profiles have been manually designed with the objective of three overlapping slices covering a distance range of approximately 150 m. While the first slice covers the close range up to 60-70 m, the second slice starts at 20 m and reaches 120 m. The third slice covers the far range from 60-180 m. Note that slices at larger distances require more illumination pulses in order to compensate irradiance and atmospheric attenuation. The gated image provides a fully-illuminated scene and is obtained by integrating all three slices.

In the future, we believe that there is substantial potential in optimizing the exposure profiles for specific tasks. However, this work focuses on algorithms based on gated images and we show that the intrinsic depth encoding of gated imaging can be exploited for 3D object detection.

6. Gated3D Dataset

6.1. Capture

Within this work we utilize the dataset from Bijelic et al. [1] and refine the labels in the gated domain. The dataset includes gated images which cover a range of scenes, weather conditions, and ambient illuminations. In total, a test vehicle

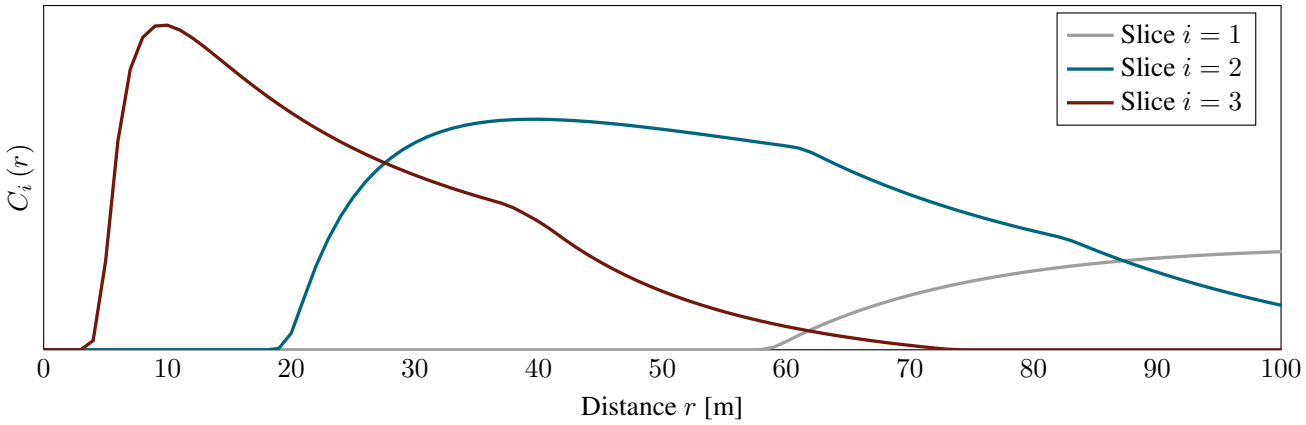

Figure 3: Range-intensity profiles $C_i(r)$ defined by the gating parameters given in Table 5.

	Laser duration	Gate duration	Delay ξ	Pulses
Slice 1	240 ns	220 ns	260 ns	202
Slice 2	280 ns	420 ns	400 ns	591
Slice 3	370 ns	420 ns	750 ns	770

Table 5: Gating parameters that we use in this work.

was driven for more than 10,000 km in Northern Europe, covering diverse scenes from Germany, Sweden, Finland and Denmark. In Germany, different sized cities, i.e. Biberach a. d. Riß, Blaubeuren, Ehingen, Hamburg, Immenstadt, Kempten (Allgäu), Kiel, Lindau, Memmingen, Münsingen, Oberstaufen, Oberstdorf, and Ulm are in the dataset. In northern Sweden, the captured cities include Gävle, Gothenburg, Karlstad, Linköping, Luleå, Örebro, Stockholm, Sundsvall, Umeå, Uppsala, Vårgårda, and Västerås. In Denmark, Copenhagen has been captured. In Finland, the dataset covers data both in the very north in Muonio, Oulo, and Rovaniemi, as well as Helsinki in the south. The main objective was to capture data under challenging nighttime and adverse weather conditions. Scandinavians seemed much more used to bad weather because even under extreme snowfall and rain, the streets were still busy with many pedestrians and cyclists.

In addition to the gated imaging system from Brightway Vision, the vehicle system was equipped with state-of-the-art sensors for environment perception, i.e. a 64-lines lidar scanner and a 2MP stereo camera. In total, 1.4 million frames at 10 Hz were recorded during all test drives. Frame were discarded if at least one sensor failed due to technical problems or being covered with snow or dirt. Moreover, before expensive and cumbersome annotation, only the most interesting frames were selected where the time shift between the gated camera and lidar is small enough that the projected point cloud matches the semantic image content.

6.2. Annotation

The dataset provides annotations on a subset of 13k images which we extend by an additional 2.5k samples. 3D bounding boxes are annotated on the lidar point clouds are visualized as a 2.5D bounding box projected on to the RGB and gated camera frame. Objects up to a distance of 80 m with a minimum number of 5 lidar points are annotated. As the first control measure, the projected 2.5D bounding boxes illustrate the object dimension and position, which enabled to correct those measures. Missing annotations are identified and added, along with tight 2D bounding boxes within the RGB camera frame, while the 2D boxen in the gated frame are only based on the projected 3D box hull. As a second automatic control measure, the projected 3D bounding boxes are refined based on the intersection over union (IoU) of the projected 3D box and tight 2D box inside the RGB camera. Note, 3D bounding boxes overestimate the object size at rounded object edges. Therefore, we require a minimum bounding box overlap of 0.7 IoU. Objects below that threshold are re-annotated by human annotators to correct either the 3D or 2D box. Finally, all frames are reviewed by another human annotator checking for label consistency across all sensor modalities. Additionally, the label *Don't Care* with a rough bounding box is used when we recognize objects that cannot intervene in the current traffic situation (e.g. cars in a parking lot or separated highway lane) or when objects

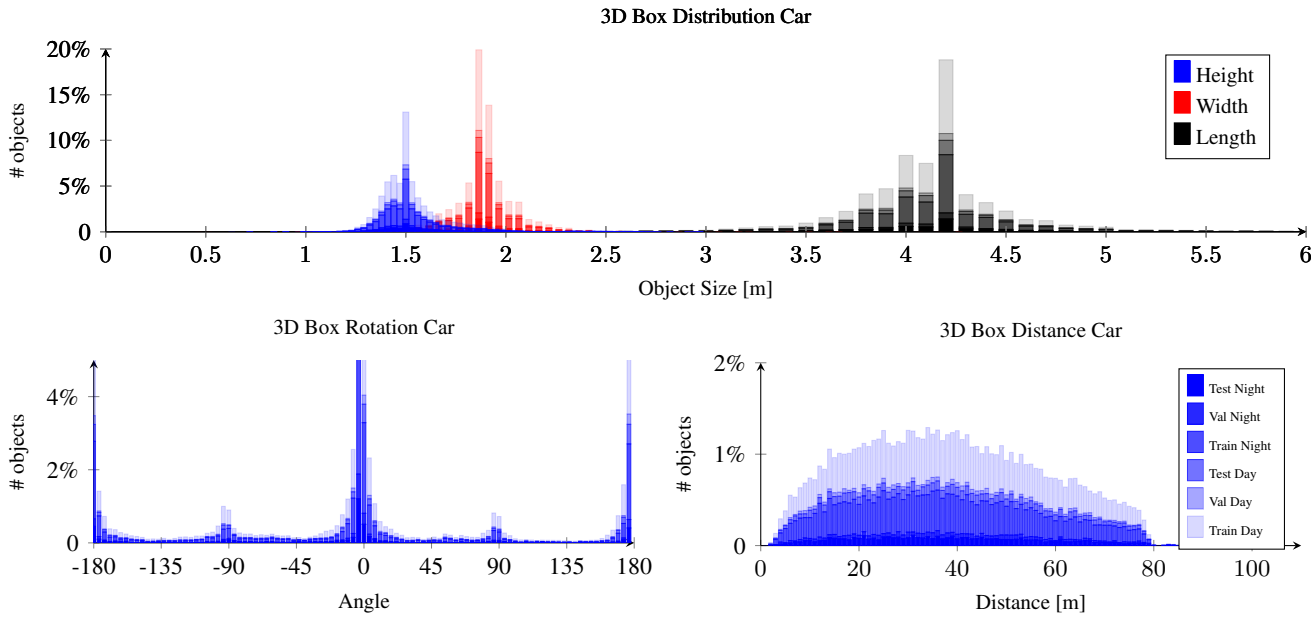

Figure 4: 3D bounding box distributions for the class *Car*. The box sizes (height, width, length), yaw angle and distance from ego vehicle are illustrated.

such as pedestrians at large distances cannot be distinguished. Groups where individual objects can not be differentiated are labelled with the property *is group*.

Note that the annotations from previous work in Bijelic et al. [1] were mainly done on the RGB and lidar modalities. In most cases the labels for the gated camera were transferred semi-automatically through constant calibrations. Hence, the 2D boxes are only based on the projected 3D box hull and are not tight around the objects within the gated frame. Only objects that were exclusively visible in the gated images were annotated independently. To remove this limitation and increase the annotation quality we re-annotate precise 2D bounding boxes for all 100k objects in the 13k gated images and filter non-matching 3D boxes due to time misalignment issues. To increase performance further we provide 2.5k labels from a similar gated setup with lidar and RGB image ground truth data to enrich the dataset further.

Bounding box and object distributions for the refined dataset can be found below.

6.3. Dataset Distribution

Figures 4 and 5 visualize the specific object distributions for the classes *Car* and *Pedestrian*, respectively. The dimensions of a car are approximately normally distributed around a mean of 1.8 m × 4.1 m × 1.5 m (W × L × H) while the mean dimensions for pedestrians is 0.6 m × 0.5 m × 1.7 m (W × L × H). Most objects are either perpendicular or in line with the ego-car. The number of objects decreases slightly with distance up to the maximum annotation distance of 80 m. Figure 6 shows the total number of 52,580 cars and 38,567 pedestrians.

References

[1] Mario Bijelic, Tobias Gruber, Fahim Mannan, Florian Kraus, Werner Ritter, Klaus Dietmayer, and Felix Heide. Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather. *CVPR*, 2020. 7, 9

[2] Garrick Brazil and Xiaoming Liu. M3d-rpn: Monocular 3d region proposal network for object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9287–9296, 2019. 3

[3] Xiaozhi Chen, Kaustav Kundu, Yukun Zhu, Andrew G Berneshawi, Huimin Ma, Sanja Fidler, and Raquel Urtasun. 3d object proposals for accurate object class detection. In *Advances in Neural Information Processing Systems*, pages 424–432, 2015. 3

[4] Alessandro Foi, Mejdi Trimeche, Vladimir Katkovnik, and Karen Egiazarian. Practical poissonian-gaussian noise modeling and fitting for single-image raw-data. *IEEE Transactions on Image Processing*, 17(10):1737–1754, 2008. 7

[5] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12697–12705, 2019. 3

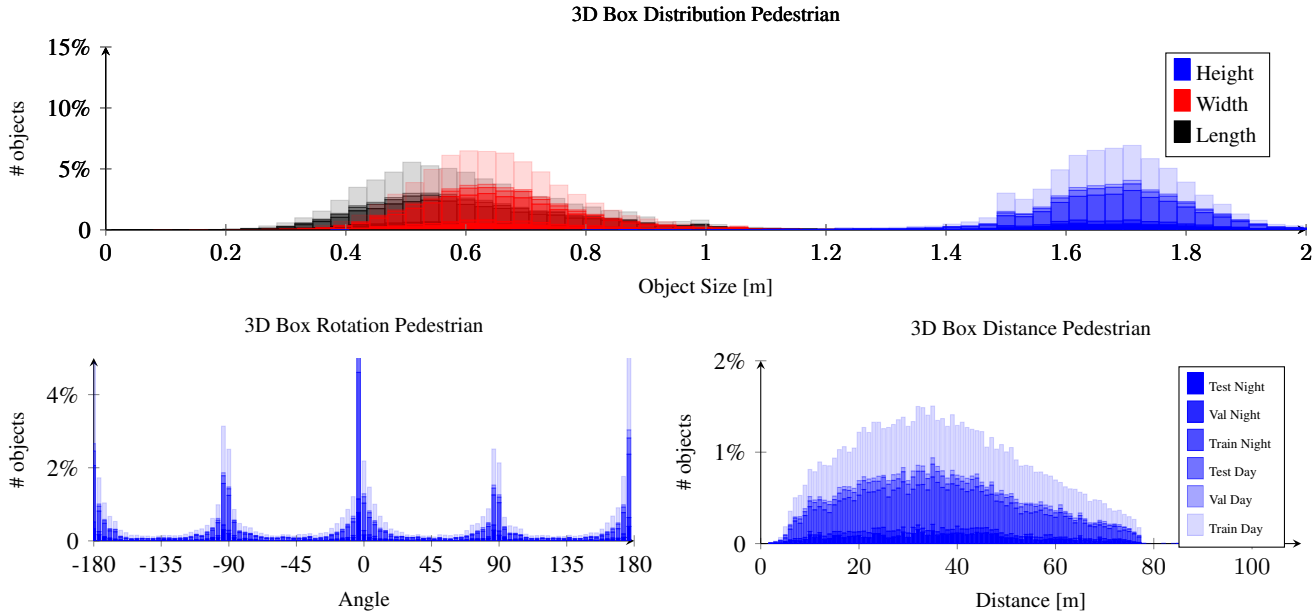

Figure 5: 3D bounding box distributions for the *Pedestrian* class. The box sizes (height, width, length), yaw angle and distance from ego vehicle are illustrated.

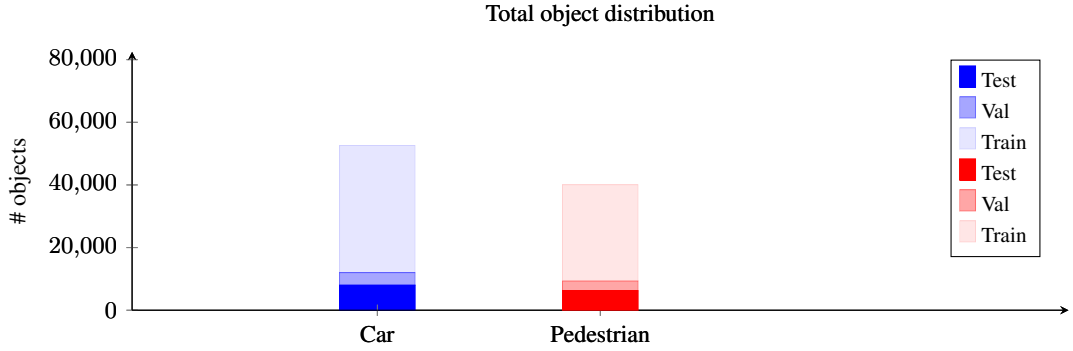

Figure 6: Object distribution for all training, validation and test splits.

[6] Peiliang Li, Xiaozhi Chen, and Shaojie Shen. Stereo r-cnn based 3d object detection for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 3

[7] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 740–755, Cham, 2014. Springer International Publishing. 4

[8] Xinzhu Ma, Shinan Liu, Zhiyi Xia, Hongwen Zhang, Xingyu Zeng, and Wanli Ouyang. Rethinking pseudo-lidar representation. *arXiv preprint arXiv:2008.04582*, 2020. 3

[9] Yurong You, Yan Wang, Wei-Lun Chao, Divyansh Garg, Geoff Pleiss, Bharath Hariharan, Mark Campbell, and Kilian Q Weinberger. Pseudo-lidar++: Accurate depth for 3d object detection in autonomous driving. *arXiv preprint arXiv:1906.06310*, 2019. 3