Gated3D: Monocular 3D Object Detection From Temporal Illumination Cues

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

Abstract

Today's state-of-the-art methods for 3D object detection are based on lidar, stereo, or monocular cameras. Lidar-based methods achieve the best accuracy, but have a large footprint, high cost, and mechanically-limited angular sampling rates, resulting in low spatial resolution at long ranges. Recent approaches using low-cost monocular or stereo cameras promise to overcome these limitations but struggle in low-light or low-contrast regions as they rely on passive CMOS sensors. We propose a novel 3D object detection modality that exploits temporal illumination cues from a low-cost monocular gated imager. We introduce a novel deep detection architecture, Gated3D, that is tailored to temporal illumination cues in gated images. This modality allows us to exploit mature 2D object feature extractors that guide the 3D predictions through a frustum segment estimation. We assess the proposed method experimentally on a 3D detection dataset that includes gated images captured over 10,000 km of driving data. We validate that our method outperforms state-of-the-art monocular and stereo methods, opening up a new sensor modality as an avenue to replace lidar in autonomous driving. https://light.princeton.edu/gated3d

1. Introduction

3D object detection is a fundamental vision task in robotics and autonomous driving. Accurate 3D detections are critical for safe trajectory planning, with applications emerging across disciplines such as autonomous drones, assistive and health robotics, as well as warehouse and delivery robots. RGB-D cameras using correlation time-of-flight [22, 29, 34], such as Microsoft's Kinect One, enable robust 3D detection indoors [55, 56] for small ranges. In the past, autonomous driving, which requires long ranges and high depth accuracy, has relied on scanning lidar for 3D detection [50, 60, 15, 64, 35, 11, 68, 30, 33]. However, while lidar provides accurate depth, existing systems are fundamentally limited by point-by-point acquisition, resulting in spatial resolution that falls off quadratically with distance and linearly with frame rate. In contrast to conventional cameras, lidar systems are three orders of magnitude more expensive, suffer from low resolution at long distances, and fail in the presence of strong back-scatter, e.g. in snow or fog [3].

Promising to overcome these challenges, a recent line of work proposed *pseudo-lidar sensing* [61], which relies on low-cost sensors, such as stereo [10, 7, 27] or monocular [9, 20, 14] to recover dense depth maps from conventional intensity imagers. Point-clouds are sampled from the depth maps and ingested by 3D detection methods that operate on point-cloud representations [33, 68]. More recent methods predict 3D boxes directly from the passive input images [36, 4, 54]. Although all of these methods promise low-cost 3D detection with the potential to replace lidar, they rely on *passive* camera-only sensing. Passive stereo approaches degrade at long ranges, where disparities are small, and in low-light scenarios, e.g. at night, when stereo or monocular depth cues are less visible.

In this work, we introduce the first 3D object detection method using gated imaging and evaluate this as a low-cost detection method, outperforming recent monocular and stereo detection methods. Similar to passive approaches, we use CMOS sensors but add active temporal illumination. The proposed gated imager captures illumination distributed in three wide gates ($>$ 30 m) for all sensor pixels. Gated imaging [25, 5, 2, 63, 49, 1, 21] allows us to capture several dense high-resolution images distributed continuously across the distances in their respective temporal bin. Additionally, back-scatter can be removed by the distribution of early gates. Whereas scanning lidar trades off temporal resolution with spatial resolution and signal-to-noise ratio (SNR), the sequential acquisition of gated cameras trades off dense spatial resolution and SNR (i.e. wide gates) with coarse temporal resolution. We demonstrate that the temporal illumination variations in gated images are a depth cue naturally suited for 3D object detection. Operating on 2D gated slices allows us to leverage existing 2D object detection architectures to guide the 3D object detection task with a novel frustum segmentation. The proposed architecture further exploits gated images by disentangling the

[*]indicates equal contribution.

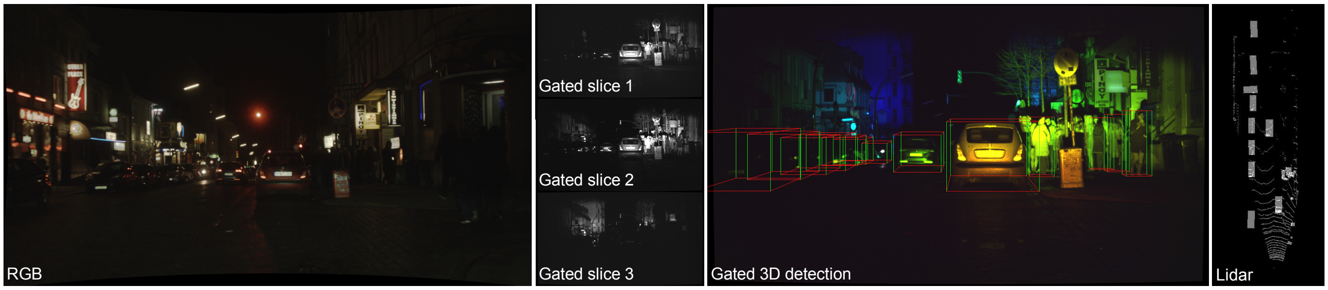


Figure 1: We propose a novel 3D object detection method, "Gated3D", which uses a flood-illuminated gated camera. The high resolution of gated images enables semantic understanding at long ranges. In the figure, our gated slices are color-coded with red for slice 1, green for slice 2 and blue for slice 3. We evaluate Gated3D on real data collected with a Velodyne HDL64-S3D scanning lidar as reference, as seen in the overlay on the right.

semantic contextual features from depth cues in the gates through a two stream feature extraction. Relying on the resulting high-resolution 2D feature stacks, the method outperforms existing methods especially at long ranges. Although the proposed architecture is trained using only gated images as input, it naturally supports fusion with other existing depth modalities, e.g. from RGB stereo or lidar depth maps. The method runs at real-time frame rates and outperforms existing passive imaging methods, independent of the ambient illumination – promising low-cost CMOS sensors for 3D object detection in diverse automotive scenarios.

Specifically, we make the following contributions:

- We formulate the 3D object detection problem as a regression from a frustum segment, computed using 2D detection priors and the object dimension statistics.
- We propose a novel end-to-end deep neural network architecture that solves this regression with depth cues and semantic features from gated images.
- We validate the proposed method on real-world driving data acquired in challenging automotive scenarios. The proposed approach detects objects with high accuracy up to 80 m, outperforming existing monocular, stereo and pseudo-lidar low-cost methods.
- We provide 3D annotations for gated data captured in northern Europe, along with code and models.

As an example, Figure 1 shows experimental results of the proposed method. The gated image contains dense information on objects further away in the scene. The advantage of gated sensors for nighttime scenes is also demonstrated in this example, where the pedestrians are not clearly visible in the RGB image.

2. Related Work

Depth Sensing and Estimation. Passive acquisition methods for recovering depth from conventional intensity images operate on single monocular images [8, 20, 32, 14, 48, 4], temporal sequences of monocular images [28, 58, 59, 67], or on multi-view stereo images [23, 51, 7, 44, 36]. These methods all suffer in low-light and low-contrast scenes. Active depth sensing overcomes these limitations by actively illuminating the scene, and scanning lidar [50] has emerged as an essential depth sensor for autonomous driving, independent of ambient lighting. However, the spatial resolution of lidar is fundamentally limited by the sequential point-by-point scanning frame rate and the sensor cost is significantly higher. Recently, gated cameras were proposed as an alternative for dense depth estimation [21]. Although promising depth estimates have been demonstrated with gated cameras, local artefacts and low-confidence regions in outputs from Gruber et al. [21] call into question if its performance for high-quality scene understanding tasks could surpass that of recent monocular and stereo-based methods – a gap addressed in this work in an end-to-end fashion by directly processing the gated input slices.

CNN 2D Object Detection. Convolutional neural networks (CNNs) for efficient 2D object detection have outperformed classical methods that rely on hand-crafted features by a large margin [47]. The key concept behind such learned object detectors is the classification of image patches at varying positions and scales [52]. Discretized grid cells and predefined object templates (anchor boxes) are regressed and classified by fully-convolutional network architectures [40]. To this end, two popular directions of research have been explored: single-stage [39, 46, 26, 38] and proposal-based two-stage detectors [19, 18, 47]. Two-stage approaches such as R-CNN [19] and Faster R-CNN [47] generate region proposals for objects in the first stage followed by object classification and bounding box refinement in the second stage [19]. Single-stage detectors such as SSD [39] and YOLO [46] directly predict the final detections and are usually faster than two-stage detectors but with lower accuracy. Recently, RetinaNet [38] proposed a focal loss that effectively down-weights easily-classified background examples and showed that single-stage detectors trained with this loss can match two-stage detectors in terms of accuracy.

3D Object Detection. A large body of work on 3D object detection has explored different scene and measurement representations. For lidar point cloud data, one direction is to rely on voxel-based representations [60, 15, 68, 12, 53]. Unfortunately, the computational cost of the 3D convolutions required for voxel-based approaches is prohibitive for real-time processing [60, 15]. Alternatively, the height dimension of the voxel grid can be collapsed into feature channels with 2D convolutions performed in the BEV plane [64, 33, 41], trading off height information for computational efficiency.

Although existing state-of-the-art methods rely on lidar, recent work aims to close the performance gap with low-cost passive sensors due to the limitations of scanning lidar, such as cost, size, low angular resolution and failure in back-scatter.

Earlier work on monocular [9, 54, 4, 6, 31] and stereo [36] methods leveraged convolutional architectures from 2D object detection, extracting depth information from stereo disparity, geometric constraints, or object dimensions [6, 31] in an end-to-end fashion. We integrate these concepts into a frustum segment-based approach that improves depth prediction.

More recently, pseudo-lidar [61] showed that point cloud input representations can be used with passive imaging approaches by first estimating depth maps. Several methods have since followed this approach with monocular [62, 43] and stereo [65] depth estimation. PatchNet [42] proposed that the advantage of pseudo-lidar is its explicit depth information in its input rather than the point cloud representation. Instead, PatchNet uses a 2D convolutional architecture with the estimated (x,y,z) coordinates of each pixel as its input. Estimating the depth prior to the detection network effectively disentangles depth information from object appearance, improving the detection accuracy.

In this work, we propose a method for 3D detection using 2D gated images, offering a low-cost solution comparable to passive sensors with improved detection accuracy. This input representation allows us to leverage the rich body of efficient 2D convolutional architectures for the task of 3D object detection, while the gated slices represent depth more effectively than RGB images.

3. Gated Imaging

Gated imaging is an emerging sensor technology for self-driving cars which relies on active flash illumination to allow for low-light imaging (e.g. night driving) while reducing back-scatter in adverse weather situations such as snow or fog [21].

As shown in Figure 2, a gated imaging system consists of a flood-illuminator and synchronized gated image sensor that integrates photons falling in a window of round-trip path-length ξc, where ξ is a delay in the gated sensor and c is the speed of light. Following [21], the range-intensity profile (RIP) $C(r)$ describes the distance-dependent integration, which is independent of the scene and given by

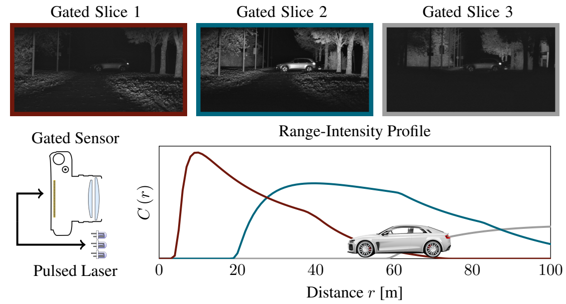


Figure 2: A gated system consists of a pulsed laser source and a gated imager that are time-synchronized. The range-intensity profile (RIP) $C_i(r)$ describes the distance-dependent illumination for a slice i. A car at a certain distance appears with a different intensity in each slice according to the RIP.

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

where g is the temporally modulated camera gate, p the laser pulse profile and β models atmospheric interactions. Assuming now a scene with dominating lambertian reflector with albedo α at distance $\tilde{r}$, the measurement for each pixel location is obtained by

$$z = \alpha C(\tilde{r}) + \eta_{\mathrm{p}}\left(\alpha C(\tilde{r})\right) + \eta_{\mathrm{g}}, \tag{2}$$

where η_{p} describes the Poissonian photon shot noise and η_{g} the Gaussian read-out noise [16]. In this work, we capture three images $\mathbf{Z}_i \in \mathbb{N}^{\text{height}\times\text{width}}$ for $i \in \{1,2,3\}$ with different profiles $C_i(r)$ that encode depth into these three slices.

4. 3D Object Detection from Gated Images

Next, we introduce *Gated3D*, a novel model for detecting 3D objects from temporal illumination cues in gated images. Given three gated images, Gated3D determines the 3D object location, dimensions, orientation and class.

Architecture Overview The proposed architecture is illustrated in Figure 3. Our model is composed of a 2D detection network, based on Mask R-CNN [24], and a 3D detection network designed to effectively integrate semantic, contextual, and depth information from gated images. Our model is trained end-to-end using only 3D bounding box annotations with no additional depth supervision. However, we also investigate the use of depth maps as a training signal. Although we focus on depth maps from gated-based modality, depth maps can also be generated from RGB or stereo images. Through this experimentation, we then show how our model can potentially be integrated with modalities that can add features orthogonal to gated cues.

The 2D detector predicts bounding boxes that guide the feature extraction with a FPN [37] backbone. These 2D boxes are used to estimate frustum segments that constrain the 3D location. In addition to these geometric estimates, the 3D detection network receives the cropped and resized regions of interest extracted from both the input gated slices and the backbone features. To extract contextual, semantic and depth information from the temporal intensity variations of the gated images, our 3D detection network applies two separate convolution streams: one for the backbone features and another for the gated input slices. The resulting features are fed into a sequence of fully-connected layers that predict 3D object location, dimensions, and orientation.

The remainder of this section details our proposed 2D object detection network 4.1, 3D prediction network architecture 4.2 and the loss functions for training 4.3.

4.1. 2D Object Detection Network

The proposed 2D detection network uses a FPN [37] as a backbone and RoIAlign for extracting crops of both the features and input gated slices. We extract features maps P_2, P_3, P_4 and P_5 of the backbone, as defined in [37].

Our 2D object detection network follows a two-stage architecture, where the final 2D box detections are refined from proposals output by a region proposal network (RPN). In contrast to Mask RCNN [24], we use these 2D detections instead of the RPN proposals for 3D detection. Using the refined 2D detections allows the 3D box prediction network to obtain more precise region features, especially from the input gated slices, and a more precise frustum segment, which is essential for depth estimation.

4.2. 3D Object Detection Network

Our 3D prediction network fuses the extracted features from both the input gated slices and the backbone features. The gated stream extracts depth cues from the cropped gated input slices with a sequence of convolutions per slice, without parameter sharing. These convolutions consist of three layers with $3 \times 3 \times 16$, $3 \times 3 \times 32$ and $3 \times 3 \times 32$ kernels. The network fuses the three gated features and the backbone features by concatenating along the channel dimension and processing with 5 residual layers. Instead of pooling or flattening the resulting features, an attention subnetwork produces softmax attention maps for each feature channel which are used for a weighted sum over the height and width of the features. The resulting feature vectors are fed into two fully connected layers, followed by a final layer that generates eight 3D bounding box coefficients.

We denote an object's predicted 2D bounding box as $P = (c, u, v, w_u, h_v)$, where c is object's class, (u, v) is the bounding box center, and (w_u, h_v) define its height and width, respectively. The 3D detection network takes P and estimates a set of parameters Q, that define a 3D bounding box whose projection is given by P. The problem of estimating Q is ill-posed as given a specific 2D bounding box P, there are an infinite number of 3D boxes that can be projected to P. However, we can restrict the range of locations of Q to a segment of the 3D viewing frustum extracted from P, using the object's approximate dimensions and P. See Figure 4 for an illustration.

Estimating the 3D location is aided by a frustum region similar to [45]. For lidar data, a frustum suffices to define an object in 3D space as lidar provides depth values. In our case, we only have data in the image space, without absolute depth value. Instead of considering the whole frustum as in [45], we leverage the camera calibration and object dimensions in the training set to guide depth estimation. This idea is illustrated in Figure 4, where a person is located at different distances relative to the camera. Using the object height and 2D bounding box projection, we can estimate the distance to the camera through triangulation. Assuming a bounded height, we can accurately estimate the segment of the frustum where the object is located. In the example in Figure 4 we define the minimum and maximum height values to be 1.5m and 2m.

For each 2D bounding box $P = (c, u, v, w_u, h_v)$ generated by the 2D detection network, our 3D bounding box network is trained to estimate the parameters $Q' = (\delta u', \delta v', \delta z', \delta h', \delta w', \delta l', \theta')$, which encode the location (x, y, z), dimensions (h, w, l), and orientation (θ') of a 3D bounding box as follows

3D Location. We estimate the objects location (x, y, z) using its projection over the image space, as well as a frustum segment. Specifically, we define the target $\delta u', \delta v'$ values as

$$\delta u' = (Proj2d_u(x, y, z) - u)/w_u \tag{3}$$

$$\delta v' = (Proj2d_v(x, y, z) - v)/h_v, \tag{4}$$

where $Proj2d_u(x, y, z), Proj2d_v(x, y, z)$ represent the u, v coordinates of the 2D projection of (x, y, z) over the image space.

To define the target z, we first define a frustum segment used as a reference for depth estimation. Given an object with height h, we can estimate the object distance to the camera with focal length f_v as

$$f(h_v, h) = \frac{h}{h_v} f_v. \tag{5}$$

If we assume that h follows a Gaussian Distribution with mean μ_h and standard deviation σ_h, given $P = (c, u, v, w_u, h_v)$ and f_v, we can constrain the distance from the object to the camera to a range of $[f(h_v, \mu_h - \sigma_h), f(h_v, \mu_h + \sigma_h)]$, or, more generally, we deduct that the frustum segment has a length d

$$d = f(h_v, \mu_h + k * \sigma_h) - f(h_v, \mu_h - k * \sigma_h), \tag{6}$$

where k is a scalar that adjusts the segment extent and is inversely proportional to our prediction confidence.

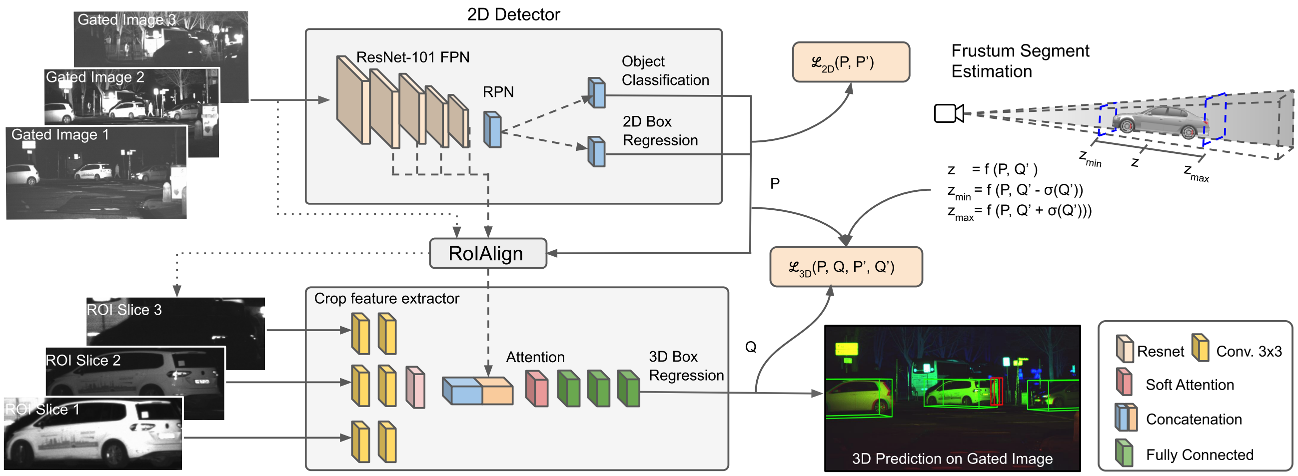


Figure 3: From three gated slices, the proposed *Gated3D* architecture detects objects and predicts their 3D location, dimension and orientation. Our network employs a 2D detection network to detect ROIs. The resulting 2D boxes are used to crop regions from both the backbone network and input gated slices. Our 3D network estimates the 3D object parameters using a frustum segment computed from the 2D boxes and 3D statistics of the training data. The network processes the gated slices separately, then fuses the resulting features with the backbone features and estimates the 3D bounding box parameters. P, Q denote ground-truth boxes, and P', Q' denote predicted boxes.

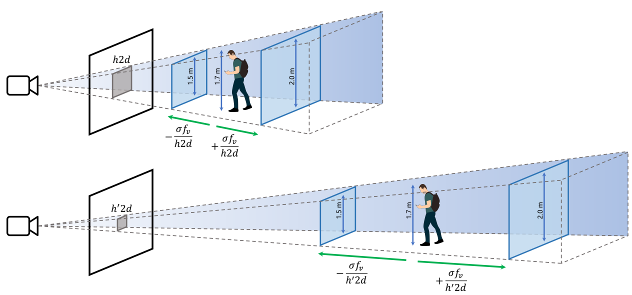


Figure 4: There is an infinite number of 3D cuboids that can project to a given bounding box P. However, the object location can be reasonably estimated using the object height, its projected height, and the vertical focal length.

Following these observations, the z coordinate of the 3D bounding box, $\delta z'$, is given as

$$\delta z' = \frac{z - f(h_v, h)}{d}. \tag{7}$$

Thus, the model is trained to predict an offset $\delta z'$ between the actual depth z and the approximate depth $f(h_v, h)$. Normalization with depth d is key to estimate the absolute depth of the objects. Intuitively, for higher distances z there is greater localization uncertainty in the labels and as such, the training loss needs to account for this proportionally. Note, that this does not require the object to be inside the frustum segment to be detected, but rather uses the frustum segment length to scale the offset. There are no additional constraints for different orientations or positions because the model can learn these offset adjustments from data.

Analogous to 2D detectors, the frustum segment can be considered as an anchor, except its position and dimensions are not fixed, instead using the camera model and object statistics to adjust accordingly. Note that other vehicle types, such as buses, can be separate classes, as is conventionally done in 2D object detection. We illustrate this point and show generalization to different orientations and positions in the Supplemental Material.

During training, we use h from ground-truth; during inference, we use the network prediction.

3D Box Dimensions and Orientation. The target 3D box dimensions are estimated using $\delta h', \delta w', \delta l'$, which are defined as the offset between the mean of the objects dimensions, per class, and the *true* dimensions.

$$\delta p' = \frac{p - \mu_p}{\mu_p}, \forall p \in \{h, w, l\}. \tag{8}$$

To learn the target orientation (observation angle) θ', the orientation is encoded as $(\sin_{\theta'}, \cos_{\theta'})$, and the network is trained to estimate each parameter separately.

Additional Depth Map Inputs. We also investigate the use of dense depth estimation as an additional training signal. Depth maps are estimated using the network proposed in [21] and are integrated into the Gated3D architecture in the second stage after RoIAlign cropping. Following the same architecture as the gated crop feature extractor, the depth map crop features are then concatenated with the gated and backbone features.

4.3. Loss Functions

Given a 3D box parameters ground-truth box $Q = (\delta u, \delta v, \delta z,\ \delta h, \delta w, \delta l, \sin_\theta, \cos_\theta)$, and its corresponding prediction $Q' = (\delta u', \delta v', \delta z', \delta h', \delta w', \delta l', \theta')$, we define our overall loss $\mathcal{L}_{3\mathrm{D}}(Q, Q')$ as

$$\mathcal{L}_{3\text{D}}(Q, Q') = \alpha \cdot \sum_{l \in \{u,v,z\}} L_{loc}(\delta l - \delta l') + \sum_{d \in \{h,w,l\}} L_{dim}(\delta d - \delta d') + \beta \cdot L_{ori}(\sin_\theta, \cos_\theta, \theta'), \tag{9}$$

where L_{loc} is the location loss, L_{dim} is the dimensions loss, and $L_{ori}(\theta, \theta')$ is the orientation loss. We use α and β to weight the location and orientation loss, and define these values during training. We define L_{loc} and L_{dim} as $SmoothL_1$, and $L_{ori}(\sin_\theta, \cos_\theta, \theta')$ as

$$L_{ori}(\sin_\theta, \cos_\theta, \theta') = (\sin_\theta - \sin(\theta'))^2 + (\cos_\theta - \cos(\theta'))^2. \tag{10}$$

The method runs at approximately 10 FPS on an Nvidia RTX 2080 GPU in TensorFlow without implementation optimization such as TensorRT. We refer to the Supplemental Material for additional method and implementation details.

5. Datasets

In this section, we describe the 3D object detection dataset we use to train and evaluate *Gated3D*.

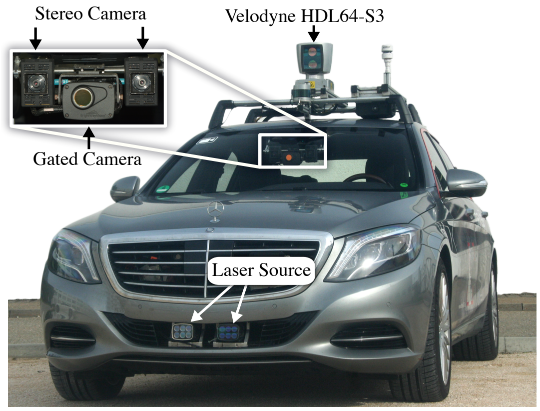
Sensor Setup. Since existing automotive datasets [57, 13, 17, 66] do not include measurements from gated cameras, we use the dataset from Bijelic et al. [3] who collected gated images, along with RGB, lidar and FIR data during a large-scale data acquisition in Northern Europe. We combine this dataset with additional validation and test data acquired with a test vehicle with the same gated system *BrightEye* from BrightwayVision:

- A gated CMOS pixel array of resolution 1280 px × 720 px with a pixel pitch of 10 µm. Using a focal length of 23 mm provides a horizontal and vertical field of view of 31.1° H × 17.8° V.
- Two repetitive pulsed vertical-cavity surface-emitting laser (VCSEL) which act as a pulsed illumination source at a wavelength of 808nm, not visible to humans. The peak power is within eye safety regulations. The source is mounted below the bumper of the vehicle, see Figure 5.

The gated images consist of three exposure profiles as shown in Figure 2, see gate settings (delay, laser duration, gate duration) in the Supplemental Document. For each single capture, multiple laser flashes are integrated before readout in order to increase the measurement signal-to-noise.

For comparison with state-of-the-art 3D detection approaches, following Bijelic et al.[3], we equip the test vehicle with a Velodyne HDL64 lidar scanner and a stereo camera. The stereo system consists of two cameras with OnSemi AR0230 sensors mounted at 20.3 cm baseline. All sensor specifications are listed in Figure 5.

3D Annotation and Dataset Split In addition to the data from Bijelic et al. [3], which contains 13k gated images, we capture an additional 2.5k gated images. We use the



	Gated Camera	Stereo Camera	Lidar
Sensor	BrightwayVision BrightEye	2x OnSemi AR0230	Velodyne HDL64-S3D
Resolution	1280px×720px	1920px×1080px	1440”×612”
Wavelength	808 nm	Color	905 nm
Frame Rate	120 Hz	30 Hz	10 Hz
Bit Depth	10 bit uint	12 bit uint	32 bit float

Figure 5: Sensor setup for recording the dataset used for training and evaluating the proposed method. We also capture corresponding lidar point clouds and stereo image pairs. The stereo camera is located at approximately the same position of the gated camera in order to ensure a similar viewpoint.

collected raw data and additional vehicle data with a similar system described above, and annotate 3D boxes using the time-synced lidar measurements. The annotation and capture procedures for the dataset are detailed in the Supplement Document. The gated images have been manually labeled with human annotators matching lidar, gated and RGB frames simultaneously. In total, more than 100,000 objects are labeled, which comprise 4 classes. The annotations were done over 15k image examples in total. To minimize annotation issues with temporal shift between the gated images and RGB images, we refine the RGB boxes projected into the gated frames for frames that are temporally offset.

The dataset is randomly split into a training set of 10k frames, a validation set of 1,000 frames and a test set of 4,441 frames. In addition to the gated images, our proposed dataset contains corresponding RGB stereo images captured by the stereo camera system described in the previous paragraph. We note that, in contrast to popular automotive datasets, including Waymo [57], KITTI [17] and Cityscapes [13], our dataset is significantly more challenging as it also includes many nighttime images and captures under adverse weather conditions such as snow and fog.

6. Assessment

Evaluation Setting. The BEV and 2D/3D detection metrics as defined in the KITTI evaluation framework are used for evaluation, as well as the ones described by [64], which

Table 1: Object detection performance over the experimental dataset (test split). Our method outperforms monocular and stereo methods (bottom part of the table) over most of the short (0-30m), middle (30-50m) and long (50-80m) distance ranges, as well as Pseudo-Lidar based methods trained over gated images. Interestingly, our model even outperforms PointPillars lidar reference for Pedestrian detection at long distance ranges.

(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		
		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
POINTPILLARS [33]	Lidar	90.12	82.83	56.63	91.51	84.63	54.28	91.59	86.54	54.71	90.73	84.88	54.22	90.29	87.40	52.32	90.29	87.51	52.60
M3D-RPN [4]	RGB	90.44	89.29	62.76	53.21	13.26	10.52	**60.80**	16.16	10.52	90.85	80.64	59.76	51.18	20.76	2.73	52.53	21.39	2.74
STEREO-RCNN [36]	Stereo	81.56	81.07	78.08	54.17	17.16	6.17	57.92	17.69	6.26	81.73	81.03	70.85	47.36	17.21	13.02	53.81	18.34	13.08
PSEUDO-LIDAR	Gated	81.74	81.33	80.88	26.17	16.06	10.27	26.94	17.26	10.87	89.35	89.02	88.31	36.58	23.05	19.88	39.50	28.68	22.82
PSEUDO-LIDAR++ [65]	Gated	81.74	80.29	81.59	30.44	15.47	11.76	32.49	16.97	12.83	90.21	81.75	81.78	36.36	21.93	**22.39**	37.46	23.12	**23.63**
PATCHNET [42]	Gated	90.46	81.74	89.78	23.91	10.86	7.34	24.87	11.33	7.84	90.87	**89.86**	88.89	23.74	16.79	7.16	25.15	17.76	8.29
Gated3D	Gated	**90.91**	**90.88**	**90.85**	**58.55**	**27.50**	**17.59**	59.05	**32.37**	**18.74**	**90.91**	81.82	**90.85**	**57.18**	**29.97**	17.93	**57.99**	**30.36**	18.49
Gated3D w/ dense depth	Gated	**90.91**	81.82	**90.88**	56.69	24.77	15.66	57.79	24.86	15.74	90.63	81.82	**90.65**	54.74	26.43	14.1	56.31	**30.35**	15.44

(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		
		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
POINTPILLARS [33]	Lidar	70.08	49.03	0.00	69.71	45.24	0.00	70.53	48.07	0.00	69.97	43.32	0.00	71.25	41.21	0.00	70.99	43.61	0.00
M3D-RPN [4]	RGB	79.08	66.41	36.98	26.20	14.50	9.84	30.68	17.47	10.07	78.36	62.99	36.76	25.09	6.43	2.07	26.42	7.69	2.74
STEREO-RCNN [36]	Stereo	88.57	75.63	59.82	48.58	23.26	7.77	50.11	25.10	8.38	80.38	69.13	60.94	46.09	21.63	11.57	47.58	25.47	11.84
PSEUDO-LIDAR	Gated	77.87	78.38	69.11	6.19	4.59	2.15	10.28	9.14	4.13	80.34	78.61	67.78	7.53	9.58	1.62	14.27	15.72	5.55
PSEUDO-LIDAR++ [65]	Gated	77.89	77.95	60.88	9.19	2.36	3.30	14.32	5.66	4.10	79.84	79.57	54.42	7.37	7.21	2.06	12.92	11.99	5.64
PATCHNET [42]	Gated	**90.48**	80.75	69.56	32.88	18.05	5.62	39.45	20.27	9.77	81.50	**88.62**	65.43	15.37	13.37	6.75	21.60	18.15	8.46
Gated3D	Gated	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**
Gated3D w/ dense depth	Gated	90.32	81.42	79.87	48.35	**25.77**	12.28	**55.41**	**26.73**	13.66	**81.77**	81.26	79.97	**48.72**	17.35	13.16	**50.28**	22.63	14.09

calculate the metrics with respect to distance ranges. Following Simonelli et al. [54], average precision (AP) is based on 40 recall positions to provide a fair comparison. We consider *Pedestrian* and *Car* as our target detection classes.

The 3D metrics are based on intersection over union (IoU) between cuboids [11], which has the disadvantage of equally penalizing completely wrong detections and detections with IoU below the threshold. Due to the emphasis on challenging scenarios in the dataset, as well as imperfect sensor synchronization, the dataset has notably more label noise than typical public 3D object datasets. This problem is mitigated by using lower IoU thresholds than in KITTI: 0.2 for *Car* and 0.1 for *Pedestrian*. To focus on detection at different depth ranges, metrics based on difficulty as defined in KITTI are provided in the Supplemental Document.

Baselines. We compare our approach to monocular, stereo, lidar, and pseudo-lidar methods. As monocular baseline, we evaluate M3D-RPN [4], which performs 3D object detection from a single RGB image by "depth-aware" convolution, where weights in one branch of the network are shared across rows only, assuming objects higher up in the image tend to be further away. As stereo method, we evaluate STEREO-RCNN [36], which utilizes stereo image pairs to predict left-right 2D bounding boxes and keypoints that are then used to infer 3D bounding boxes using geometric constraints. Recent pseudo-lidar methods allow us to compare our method with recent state-of-the-art methods using the depth map as input, and therefore more directly assess the effectiveness of our model architecture in extracting information from gated images. To this end, we use the method from Gruber et. al. [21] to first generate dense depth maps from gated images, back-project all the pixels of the depth maps into 3D coordinates, and follow [61] to perform 3D object detection using Frustum PointNet [45]. We also evaluate Pseudo-Lidar ++ [65] depth correction method from sparse lidar, downsampled from our 64 layered lidar to four lidar rays. Furthermore, we evaluate PatchNet [42], which implements a pseudo-lidar approach based on 2D image-based representation. As a lidar reference method for reference with known (measured) depth, we evaluate POINTPILLARS [33].

We use the corresponding open source repositories and tune the hyperparameters of each baseline model during training over our dataset.

Experimental Validation. As described in Section 4, we also experiment with the use of depth maps as a training signal to our Gated3D model. In this experiment, we train the *Gated2Depth* model in our dataset, extract and feed the estimated depth maps from these trained models to our Gated3D network. The Gated3D network then crops the regions of interest from the depth maps, and fuses the features with the gated and backbone features through an attention mechanism, as illustrated in Figure 3.

Tables 1a and 1b, respectively, show *Car* and *Pedestrian* AP for 2D, 3D and BEV detection on the test set. Overall, our Gated3D model by itself obtains more robust performance over the different category and daytime evaluation settings, and using depth maps as an additional training signal slightly improves accuracy at near distances. Consistent with prior work [36] both the monocular and stereo baselines show a drop in performance with distance. Monocular and stereo cues for a small automotive baseline of 10 - 30cm are challenging to find with increasing range.

The proposed GATED3D method offers a new im-

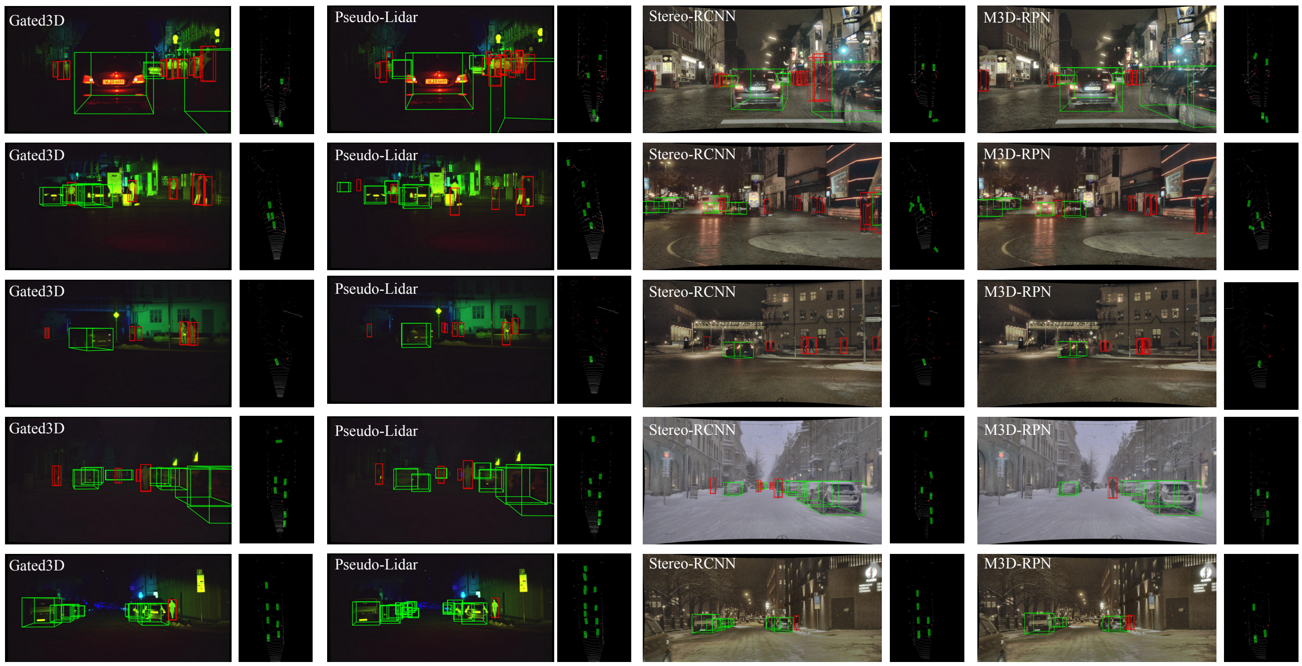


Figure 6: Qualitative comparisons on the test dataset. Bounding boxes from the proposed method are tighter and more accurate than the baseline methods. This is seen in the second image with the other methods showing large errors in pedestrian bounding box heights. The BEV lidar overlays show our method offers more accurate depth and orientation than the baselines. For example, the car in the intersection of the fourth image has a 90 degree orientation error in the pseudo-lidar and stereo baselines, and is missed in the monocular baseline. The advantages of our method are most noticeable for pedestrians, as cars are easier for other methods due to being large and specular (please zoom in on the electronic version for details).

age modality between monocular, stereo and lidar measurements. The results demonstrate improvement over intensity-only methods, especially for pedestrians and at night. GATED3D excels at detecting objects at long distances or in low-visibility situations. Note that pseudo-lidar and stereo methods can be readily combined with the proposed method — a gated stereo pair may capture stereo cues orthogonal to the gated cues exploited by the proposed method. For additional ablation studies on the components of Gated3D, please refer to the Supplemental Document.

Figure 6 shows qualitative examples of our proposed method and state-of-the-art methods. The color-coded gated images illustrate the semantic and space information of the gated data (red tones for closer objects and blue for farther away ones). Our method accurately detects objects at both close and large distances, whereas other methods struggle, particularly in the safety-critical application of detecting pedestrians at night or in adverse weather.

7. Conclusions and Future Work

This work presented the first 3D object detection method for gated images. As a low-cost alternative to lidar, *Gated3D* outperforms recent stereo and monocular detection methods, including state-of-the-art pseudo-lidar approaches. We expand on CMOS sensor arrays used in passive imaging approaches by flood-illuminating the scene and capture the temporal intensity variation in coarse temporal gates. Gated images allow us to leverage existing 2D feature-extraction architectures. We distribute the resulting features in the camera frustum along the corresponding gate – a representation that naturally encodes geometric constraints between the gates. The proposed method runs at real-time rates and we validate the method experimentally, demonstrating *higher 3D object detection accuracy than existing monocular or stereo detection methods*, including recent stereo and monocular pseudo-lidar methods with similar cost to the proposed system.

We envision our work as a first step towards gated imaging as a new sensing modality, beyond lidar, radar and camera, for a broad range of tasks in robotics and autonomous driving, including tracking, motion planning, SLAM, visual odometry, and large-scale scene understanding.

Acknowledgements

Felix Heide was supported by NSF CAREER Award (2047359) and a Sony Faculty Innovation Award. The work received funding under the AI-SEE project which is a co-labeled PENTA and EURIPIDES[2] project endorsed by EUREKA. National Funding Authorities: Austrian Research Promotion Agency (FFG), Business Finland, Federal Ministry of Education and Research (BMBF), National Research Council of Canada Industrial Research Assistance Program (NRC-IRAP).

References

[1] Amit Adam, Christoph Dann, Omer Yair, Shai Mazor, and Sebastian Nowozin. Bayesian time-of-flight for realtime shape, illumination and albedo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(5):851–864, 2017. 1

[2] Pierre Andersson. Long-range three-dimensional imaging using range-gated laser radar images. *Optical Engineering*, 45(3):034301, 2006. 1

[3] 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. 1, 6

[4] 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. 1, 2, 3, 7

[5] Jens Busck. Underwater 3-D optical imaging with a gated viewing laser radar. *Optical Engineering*, 2005. 1

[6] Yingjie Cai, Buyu Li, Zeyu Jiao, Hongsheng Li, Xingyu Zeng, and Xiaogang Wang. Monocular 3d object detection with decoupled structured polygon estimation and height-guided depth estimation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(07):10478–10485, Apr. 2020. 3

[7] Jia-Ren Chang and Yong-Sheng Chen. Pyramid stereo matching network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5410–5418, 2018. 1, 2

[8] Richard Chen, Faisal Mahmood, Alan Yuille, and Nicholas J Durr. Rethinking monocular depth estimation with adversarial training. *arXiv preprint arXiv:1808.07528*, 2018. 2

[9] Xiaozhi Chen, Kaustav Kundu, Ziyu Zhang, Huimin Ma, Sanja Fidler, and Raquel Urtasun. Monocular 3d object detection for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2147–2156, 2016. 1, 3

[10] Xiaozhi Chen, Kaustav Kundu, Yukun Zhu, Huimin Ma, Sanja Fidler, and Raquel Urtasun. 3d object proposals using stereo imagery for accurate object class detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(5):1259–1272, 2017. 1

[11] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1907–1915, 2017. 1, 7

[12] Yilun Chen, Shu Liu, Xiaoyong Shen, and Jiaya Jia. Fast point r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9775–9784, 2019. 3

[13] 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*, 2016. 6

[14] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Advances in Neural Information Processing Systems*, pages 2366–2374, 2014. 1, 2

[15] Martin Engelcke, Dushyant Rao, Dominic Zeng Wang, Chi Hay Tong, and Ingmar Posner. Vote3deep: Fast object detection in 3d point clouds using efficient convolutional neural networks. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1355–1361. IEEE, 2017. 1, 3

[16] 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. 3

[17] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361, 2012. 6

[18] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1440–1448, 2015. 2

[19] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587, 2014. 2

[20] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 1, 2

[21] Tobias Gruber, Frank D. Julca-Aguilar, Mario Bijelic, Werner Ritter, Klaus Dietmayer, and Felix Heide. Gated2depth: Real-time dense lidar from gated images. *CoRR*, abs/1902.04997, 2019. 1, 2, 3, 5, 7

[22] Miles Hansard, Seungkyu Lee, Ouk Choi, and Radu Patrice Horaud. *Time-of-flight cameras: principles, methods and applications*. Springer Science & Business Media, 2012. 1

[23] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. 2

[24] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017. 3, 4

[25] Paul Heckman and Robert T. Hodgson. Underwater optical range gating. *IEEE Journal of Quantum Electronics*, 3(11):445–448, 1967. 1

[26] Lichao Huang, Yi Yang, Yafeng Deng, and Yinan Yu. Densebox: Unifying landmark localization with end to end object detection. *arXiv preprint arXiv:1509.04874*, 2015. 2

[27] Alex Kendall, Hayk Martirosyan, Saumitro Dasgupta, Peter Henry, Ryan Kennedy, Abraham Bachrach, and Adam Bry. End-to-end learning of geometry and context for deep stereo regression. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017. 1

[28] Jan J. Koenderink and Andrea J. Van Doorn. Affine structure from motion. *Journal of the Optical Society of America A*, 8(2):377–385, Feb 1991. 2

[29] Andreas Kolb, Erhardt Barth, Reinhard Koch, and Rasmus Larsen. Time-of-flight cameras in computer graphics. In *Computer Graphics Forum*, volume 29, pages 141–159. Wiley Online Library, 2010. 1

[30] Jason Ku, Melissa Mozifian, Jungwook Lee, Ali Harakeh, and Steven L Waslander. Joint 3d proposal generation and object detection from view aggregation. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 1–8. IEEE, 2018. 1

[31] Jason Ku, Alex D. Pon, and Steven L. Waslander. Monocular 3d object detection leveraging accurate proposals and shape reconstruction. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11859–11868, 2019. 3

[32] Yevhen Kuznietsov, Jörg Stückler, and Bastian Leibe. Semi-supervised deep learning for monocular depth map prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2215–2223, 2017. 2

[33] 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. 1, 3, 7

[34] Robert Lange. 3D time-of-flight distance measurement with custom solid-state image sensors in CMOS/CCD-technology. 2000. 1

[35] Bo Li. 3d fully convolutional network for vehicle detection in point cloud. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 1513–1518. IEEE, 2017. 1

[36] 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. 1, 2, 3, 7

[37] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 936–944, 2017. 4

[38] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2980–2988, 2017. 2

[39] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *Proceedings of the IEEE European Conf. on Computer Vision*, pages 21–37. Springer, 2016. 2

[40] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015. 2

[41] Wenjie Luo, Bin Yang, and Raquel Urtasun. Fast and furious: Real time end-to-end 3d detection, tracking and motion forecasting with a single convolutional net. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3569–3577, 2018. 3

[42] Xinzhu Ma, Shinan Liu, Zhiyi Xia, Hongwen Zhang, Xingyu Zeng, and Wanli Ouyang. Rethinking pseudo-lidar representation. *arXiv preprint arXiv:2008.04582*, 2020. 3, 7

[43] Xinzhu Ma, Zhihui Wang, Haojie Li, Pengbo Zhang, Wanli Ouyang, and Xin Fan. Accurate monocular 3d object detection via color-embedded 3d reconstruction for autonomous driving. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6851–6860, 2019. 3

[44] Andrea Pilzer, Dan Xu, Mihai Puscas, Elisa Ricci, and Nicu Sebe. Unsupervised adversarial depth estimation using cycled generative networks. In *International Conference on 3D Vision (3DV)*, pages 587–595, 2018. 2

[45] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3d object detection from rgb-d data. pages 918–927, 2018. 4, 7

[46] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 779–788, 2016. 2

[47] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, pages 91–99, 2015. 2

[48] Ashutosh Saxena, Sung H. Chung, and Andrew Y. Ng. Learning depth from single monocular images. In *Advances in Neural Information Processing Systems*, pages 1161–1168, 2006. 2

[49] Michael Schober, Amit Adam, Omer Yair, Shai Mazor, and Sebastian Nowozin. Dynamic time-of-flight. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6109–6118, 2017. 1

[50] Brent Schwarz. Lidar: Mapping the world in 3D. *Nature Photonics*, 4(7):429, 2010. 1, 2

[51] Steven M. Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 519–528, 2006. 2

[52] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013. 2

[53] Shaoshuai Shi, Chaoxu Guo, Li Jiang, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. *arXiv preprint arXiv:1912.13192*, 2019. 3

[54] Andrea Simonelli, Samuel Rota Rota Bulò, Lorenzo Porzi, Manuel López-Antequera, and Peter Kontschieder. Disentangling monocular 3d object detection. *arXiv preprint arXiv:1905.12365*, 2019. 1, 3, 7

[55] Shuran Song and Jianxiong Xiao. Sliding shapes for 3d object detection in depth images. In *Proceedings of the IEEE European Conf. on Computer Vision*, pages 634–651. Springer, 2014. 1

[56] Shuran Song and Jianxiong Xiao. Deep sliding shapes for amodal 3d object detection in rgb-d images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 808–816, 2016. 1

[57] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou,

Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset, 2020. 6

[58] Philip H.S. Torr and Andrew Zisserman. Feature based methods for structure and motion estimation. In *International workshop on vision algorithms*, pages 278–294. Springer, 1999. 2

[59] Benjamin Ummenhofer, Huizhong Zhou, Jonas Uhrig, Nikolaus Mayer, Eddy Ilg, Alexey Dosovitskiy, and Thomas Brox. DeMoN: Depth and motion network for learning monocular stereo. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2

[60] Dominic Zeng Wang and Ingmar Posner. Voting for voting in online point cloud object detection. In *Robotics: Science and Systems*, volume 1, pages 10–15607, 2015. 1, 3

[61] Yan Wang, Wei-Lun Chao, Divyansh Garg, Bharath Hariharan, Mark Campbell, and Kilian Q Weinberger. Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8445–8453, 2019. 1, 3, 7

[62] Xinshuo Weng and Kris Kitani. Monocular 3d object detection with pseudo-lidar point cloud. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019. 3

[63] Wang Xinwei, Li Youfu, and Zhou Yan. Triangular-range-intensity profile spatial-correlation method for 3D super-resolution range-gated imaging. *Applied Optics*, 52(30):7399–406, 2013. 1

[64] Bin Yang, Wenjie Luo, and Raquel Urtasun. Pixor: Real-time 3d object detection from point clouds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7652–7660, 2018. 1, 3, 6

[65] 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, 7

[66] Fisher Yu, Wenqi Xian, Yingying Chen, Fangchen Liu, Mike Liao, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving video database with scalable annotation tooling. *arXiv preprint arXiv:1805.04687*, 2018. 6

[67] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G. Lowe. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2

[68] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4490–4499, 2018. 1, 3