Aerial Monocular 3D Object Detection

Yue Hu, Shaoheng Fang, Weidi Xie and Siheng Chen

Abstract— **Drones equipped with cameras can significantly enhance human's ability to perceive the world because of their remarkable maneuverability in 3D space. Ironically, object detection for drones has always been conducted in the 2D image space, which fundamentally limits their ability to understand 3D scenes. Furthermore, existing 3D object detection methods developed for autonomous driving cannot be directly applied to drones due to the lack of deformation modeling, which is essential for the distant aerial perspective with sensitive distortion and small objects. To fill the gap, this work proposes a dual-view detection system named DVDET to achieve aerial monocular object detection in both the 2D image space and the 3D physical space. To address the severe view deformation issue, we propose a novel trainable geo-deformable transformation module that can properly warp information from the drone's perspective to the BEV. Compared to the monocular methods for cars, our transformation includes a learnable deformable network for explicitly revising the severe deviation. To address the dataset challenge, we propose a new large-scale simulation dataset named AM3D-Sim, generated by the co-simulation of AirSIM and CARLA, and a new real-world aerial dataset named AM3D-Real, collected by DJI Matrice 300 RTK, in both datasets, high-quality annotations for 3D object detection are provided. Extensive experiments show that i) aerial monocular 3D object detection is feasible; ii) the model pre-trained on the simulation dataset benefits real-world performance; and iii) DVDET also benefits monocular 3D object detection for cars. To encourage more researchers to investigate this area, we will release the dataset and related code in here.**

I. INTRODUCTION

Drones equipped with cameras provide remarkable flexibility to perceive the world and have been actively used in a wide range of applications, including agricultural, aerial photography, fast delivery, and surveillance [1]. A unique advantage of drones is their maneuverability in the 3D space, enabling a vast potential in 3D scene understanding. However, the current drones' object detection is only limited to the 2D image space and the resulting 2D boxes with no 3D physical meaning [2]. This paper considers the problem of 3D object detection for images captured by drones.

In practice, developing 3D object detection systems for images captured by drones faces three critical challenges: lack of well-organized dataset, developing suitable 3D representation from drone's view, and a well-designed detection method. First, the existing drone's perception datasets [2], [3] only have 2D annotations in image coordinate, which cannot be used for 3D object detection. Second, common 3D bounding box representation for autonomous driving is not suitable for drones, as in the aerial view, the object height
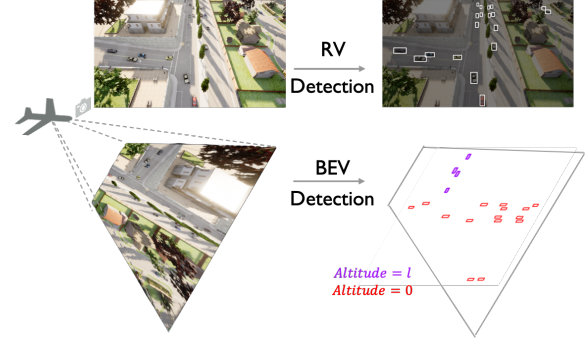
Yue Hu, Shaoheng Fang, Weidi Xie, and Siheng Chen are with Cooperative Medianet Innovation Center (CMIC), the School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai, 200240, China (e-mail:18671129361, shfang, weidi, sihengc@sjtu.edu.cn)

Fig. 1. Our dual-view object detection system simultaneously detects the objects in both 2D range view (RV) and 3D birds' eye view (BEV), given a 2D aerial image. Colors denote the BEV detections at various altitudes.

is negligible compared to the flying height of the drones so that is almost impossible to estimate. Third, the intuitive method that directly transforms 2D boxes to 3D boxes fails the aerial view due to the severe deformation issue, including aerial view variation and distant imaging, see Fig. 2. For the same reason, the emerging monocular 3D object detection methods for autonomous driving are not suitable for drones.

To resolve the dataset limitation, we first propose a comprehensive and well-organized dataset, including a simulation data version, AM3D-Sim, and a real-world data version, AM3D-Real. AM3D-Sim is based on the co-simulation of AirSIM [4] and Carla [5], where AirSIM simulates the flying drones and Carla simulates the complex background scenes and dynamic foreground objects. AM3D-Real is collected with DJI drones to validate the capability of the 3D measurement of the real world. Similar to the 3D perception dataset for autonomous driving, our dataset contains the aerial image data with both 2D and 3D annotations.

To address the object representation issue, we propose a novel representation for drones: the BEV bounding box and the categorical altitude level, which simultaneously localizes an object on the ground and reflects its altitude. Categorical altitude estimation is essential in aerial view, as i) unlike cars, the observed ground from drones is usually much broader and non-flat, especially overpass and ramps, so altitude is critical for 3D detection; ii) estimated altitudes allow to place 2D image information at the correct 3D locations, so altitude aids in precise warping. Note that we estimate the categorical altitude, instead of a continuous value. This can relieve the difficulty of altitude learning.

To address severe deformation issues caused by view variation and distant imaging in aerial view, which are rarely considered in autonomous driving, we propose a novel geo-deformable transformation, leveraging both the stability of the geometric transformation and the flexibility of the learnable deformable transformation. In the geometric trans-
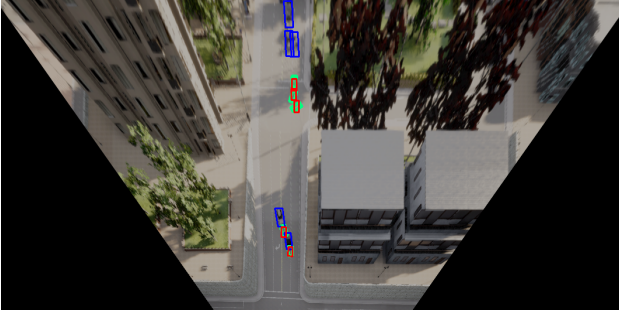
Fig. 2. Directly transforming 2D detections from RV to BEV (*Late-GeoT*) fails due to the small object size and severe deformation issue. Green, blue and red denote the ground-truth, detections of *Late-GeoT* and DVDET, respectively. Note that the BEV image is for reference only. Its pixel values are inaccurate due to deformation and loss of altitude information.

formation branch, the range-view (RV) feature is warped to the BEV based on camera pose information and the estimated altitude level. In the deformable transformation branch, we adjust the BEV features in a local region through a distance-aware deformable convolutional network. A residual structure combines the outputs of two branches. Compared to methods for autonomous driving, we additionally use a learnable DCN to adjust the geometrically warped feature, thus mitigating the severe deviation in the aerial view.

Based on the above designs, we propose a dual-view, aerial monocular 3D object detection system; termed as **DVDET**, which jointly localizes the objects in the image and the 3D space. We utilize the domain transfer technique to help handle the real-world data with the knowledge learned on the large-scale simulation dataset. We conduct comprehensive experiments to validate the effectiveness of DVDET.

To summarize, our contributions are as follows:

• We propose a novel task of aerial monocular 3D object detection to promote 3D scene understanding for drones from an aerial perspective. Our final system can simultaneously achieve 2D object detection in RV and 3D object detection in BEV given one 2D aerial image.

• We propose two core techniques specifically designed for aerial monocular 3D object detection, including trainable geo-deformable transformation, which warps the features from the RV to the BEV by leveraging camera pose parameters, geometric prior and learning ability, as well as categorical altitude estimation, which estimates the altitude level of each image pixel through classification.

• We build novel simulation and real-world benchmarks for the task of aerial monocular 3D object detection. We conduct extensive experiments to validate the proposed methods. We apply the domain transfer technique to benefit the real-world performance with the large-scale simulation data.

II. RELATED WORK

Aerial object detection. Since objects in aerial perspective have massive variations in scale and orientation, existing detection datasets and algorithms could not be directly applied. To mitigate the dataset problem, a large number of aerial object detection datasets [2], [3] with large quantities of arbitrarily oriented instances in complicated scenes are proposed. To handle the large scale and orientation variations

along with aerial perspective, a large amount of aerial object detection algorithms are proposed, e.g. feature pyramids and deformable modules are designed to handle the scale variations and the orientation variants [6], [7].

Unfortunately, the current aerial object detection datasets and algorithms only focus on the 2D range-view space and could not directly achieve 3D scene understanding. Recently, 3D object detection in driving scenarios is emerging [8], [9], [10]. To take advantage of the maneuverability of drones and fill the huge scientific blank in aerial 3D object detection, this work proposes a new task of aerial monocular 3D object detection and a well-organized dataset.

Monocular 3D object detection. Monocular 3D object detection aims to detect objects in the 3D space given the 2D image. It has three types: direct, depth-based, and grid-based. The direct methods first detect the 2D boxes and then use the geometric relation to regress the 2D boxes to 3D boxes [11], which perform inferior without explicit depth information. The depth-based methods [12] first estimate the depth map, which is combined with the image to generate the pseudo-3D point clouds. Then, 3D object detection could be performed. However, here depth estimation and detection are not trained in an end-to-end manner. The grid-based methods [13] predict BEV grid representation and conduct the 3D detection on the grid. However, equal contribution of the image features along the projection ray causes repeated grid features. Recently, [14], [15] jointly performs depth estimation and detection, and uses the estimated depth to weight the contribution of image features to the grids.

The current monocular 3D object detection methods are designed for driving scenarios, mainly about depth learning. However, drones' aerial perspective encounters severe view variation and deformation. To tackle this issue, we introduce a novel geo-deformable transformation, leveraging geometric prior and learning ability, to achieve a more precise transformation from the range view to BEV.

View transformation. View transformation is a common module of many tasks, such as the view synthesis [16], multi-view pedestrian detection, and tracking [17]. It has two types: geometric and parametric transformation. The non-parametric geometric transformation [17] explicitly transforms the source view to the target view based on the camera projection, easy to deploy, and performs stably, while unable to estimate the unseen areas and tolerate the view deformation. The parametric transformation implicitly implements the view transformation with neural networks [18] trained with GAN [16]. It could infer unseen regions based on the context and flexibly adjust itself to cope with the deformation, but hard to fit the diverse and changing views.

Taking advantage of the stability of the geometric transformation and the flexibility of the learnable parametric transformation, we propose the hybrid geo-deformable transformation to address the view variation and distortion issue.

III. PROBLEM FORMULATION

Aerial monocular 3D object detection aims to localize the objects in the 3D space given a single 2D aerial image. To
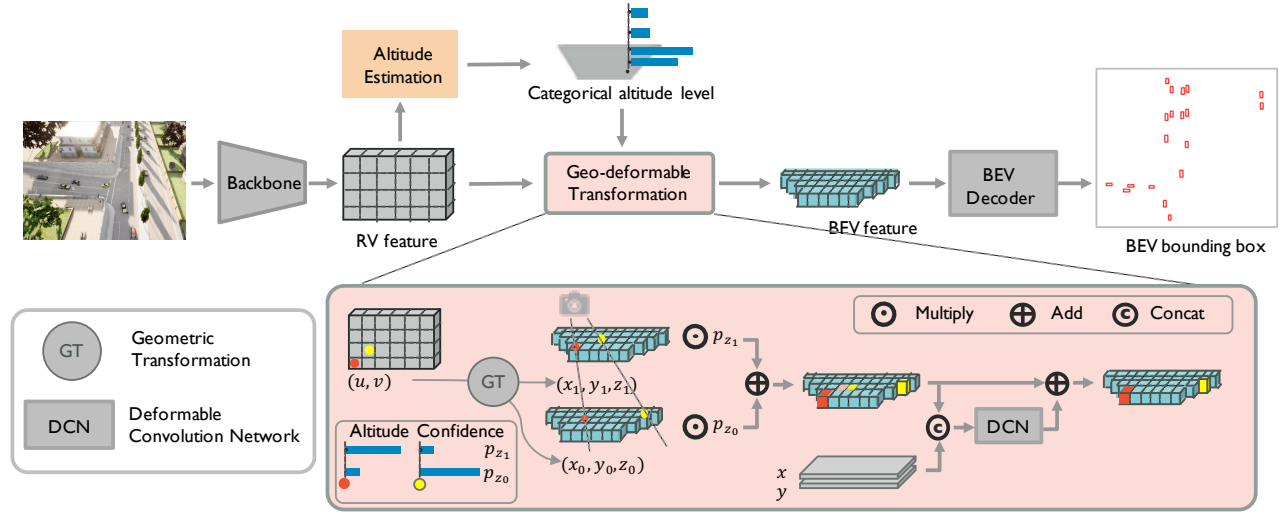
Fig. 3. The overall framework of aerial monocular 3D object detection. First, a backbone is utilized to extract the RV feature from the image data. Second, the altitude estimation module predicts the categorical altitude level for each RV feature point, afterwards, a geometric transformation is performed to get the categorical altitude level for each coordinate in BEV. Third, the RV feature and the estimated altitudes are output to a geo-deformable transformation module to generate the BEV feature. Finally, the BEV feature is decoded to the object bounding boxes with orientation.

mathematically formulate this task, we represent a 3D object from an aerial perspective. Traditionally, an object's 3D bounding box is represented in two ways: eight box corners or box parameters, including the box center, the box size (length, width, height), and the box orientation. However, neither representation is suitable for the drone's cases. As the drone's altitude is usually tens or even hundreds of times higher than that of the objects, making it almost impossible to perceive the object's height or the precise altitude.

To address such a 3D object representation issue, we use a bird's-eye-view (BEV) bounding box to reflect the object occupation on the ground and a categorical altitude to reflect the discretized altitude level. Specifically, a BEV bounding box is parameterized by (x, y, w, l, θ, c), where (x, y) is the center of the object on the ground, w is the width, l is the length, θ denotes the azimuth angle and c is the object category. We categorize the altitudes into 9 bins ranging from -1m to 8m in a space-increasing way. The center of the categorical altitude bins a are $[-1.0, -0.5, 0, 0.5, 0.75, 1.0, 1.5, 2.0, 8.0]$.

Since the aerial view is significantly different from car driving, most existing monocular 3D object detection methods designed for cars cannot be directly applied to drones. To fill this gap, we propose a series of techniques: categorical altitude estimation and geo-deformable transformation in Section V. We further integrate the proposed methods and propose an overall perception system that jointly localizes objects in the 2D image and 3D space; see Sec. VI.

IV. DATASET PREPARATION

To enable aerial monocular 3D object detection, we develop the first datasets for 3D object detection for aerial image. The previous aerial object detection datasets only provide the 2D bounding boxes in the image coordinate system. Besides, the existing 3D object detection datasets for autonomous driving could not be easily transferred to drones due to the large domain gap, for example, perspectives.

TABLE I

DATASET STATISTICS. FH IS FLYING HEIGHT IN METER. *(*/*) DENOTES TOTAL(TRAIN/TEST).

Dataset	Scenes	FH(m)	Images	Boxes
AM3D-Sim	3	40-80	48,250 (41,500/6,750)	397,984 (347,588/5,0396)
AM3D-Real	10	30-40	1,012 (919/93)	33,083 (31,668/1,415)

We propose both simulation and real-world datasets, named AM3D-Sim and AM3D-Real. The datasets include RGB images with well-annotated 2D & 3D bounding boxes of vehicles and precise camera pose information; see Tab. I. Our dataset organization refers to the database schema of NuScenes[9], an open-source autonomous driving benchmark. Our goal is to motivate monocular 3D object detection from aerial view by providing challenging benchmarks with novel difficulties to the 3D perception community.

AM3D-Sim is collected by the co-simulation of CARLA and AirSIM. CARLA [5] simulates complex scenes and traffic flow, and AirSIM [4] simulates drones flying in the scene. To promote data diversity, the flying height is set ranging from 40m to 80m, covering an area of 200m×200m. In the simulation, the annotations could be produced autonomously, so we provide a large and diverse simulation benchmark.

AM3D-Real is collected with DJI Matrice 300 RTK flying over the campus. The drone is equipped with a well-aligned LiDAR and an RGB camera. We annotate the 3D bounding boxes in the 3D point clouds collected by the LiDAR and get the 2D boxes by projecting the 3D boxes back to the image according to the calibrated camera project matrix. Due to challenging and costly data collecting and labeling, the flying height is set lower and the dataset size is relatively smaller.

V. METHODOLOGY

A. Motivation

The aerial setting has severer view deformation than the driving setting for two reasons. First, in autonomous driving, objects in the scene share almost the same altitude with the camera and the distortion along the height axis is less sensitive, so the ratio of height to depth is mostly constant.

The object height could be leveraged as a reliable proxy to estimate the depth information; while in the aerial view, there is no reliable proxy and we have to work with a pure 3D space problem. Second, autonomous driving usually considers detecting objects within 100 meters; while a drone usually considers more distant objects.

To tackle the 3D space problem in the aerial setting, we consider solutions from two aspects: i) achieve better altitude estimation; ii) compensate for the inaccurate altitude estimation. Accordingly, we propose two techniques: i) categorical altitude estimation, simplifies the challenging altitude learning by categorizing the altitude into multiple bins and substituting the strict regression task with an easygoing classification task; ii) geo-deformable transformation, corrects severe view deviation using a learnable deformable network with trainable offsets to compensate for the geometric spatial sampling generated based on the estimated altitudes.

B. Framework in Mathematics

As discussed in Sec. III, the goal is to provide the bird's-eye-view (BEV) bounding box and the categorical altitude for each object in a range-view (RV) aerial image. Fig. 3 illustrates the proposed framework of monocular 3D object detection for drones in four steps.

First, we extract the RV features from the RGB image by a backbone network. Given an image $\mathbf{I} \in \mathbb{R}^{H_I \times W_I \times 3}$ with H_I, W_I the image height and weight, the RV feature map is

$$\mathbf{F}^{(\mathrm{rv})} = f_{\mathrm{backbone}}(\mathbf{I}) \in \mathbb{R}^{H_R \times W_R \times C},$$

where $f_{\mathrm{backbone}}(\cdot)$ is a DLA-based backbone [19], and H_R, W_R, C are the height, weight and channel dimension.

Second, we estimate the categorical altitude level for each coordinate in BEV by leveraging the proposed altitude estimation module followed by the geometric transformation. The BEV map of categorical altitude $\mathbf{A}^{(\mathrm{bev})}$ is

$$\mathbf{A}^{(\mathrm{bev})} = f_{\mathrm{altitude}}(\mathbf{F}^{(\mathrm{rv})}) \in \mathbb{R}^{X \times Y \times Z}, \quad (1)$$

where $f_{\mathrm{altitude}}(\cdot)$ is the proposed altitude estimation module, X, Y denote the length of the perception field along x-axis and y-axis, Z is the amount of categorical altitude bins along z-axis. Each element $\mathbf{A}^{(\mathrm{bev})}(x, y, z)$ reflects the confidence score of the zth altitude level at the location (x, y) in BEV.

Third, we get the BEV features by warping the RV features based on the proposed geo-deformable transformation and the estimated categorical altitude. The BEV feature map is

$$\mathbf{F}^{(\mathrm{bev})} = f_{\mathrm{deform}}(\mathbf{F}^{(\mathrm{rv})}, \mathbf{A}^{(\mathrm{bev})}) \in \mathbb{R}^{X \times Y \times C}, \quad (2)$$

where $f_{\mathrm{deform}}(\cdot)$ is the proposed geo-deformable transformation module that leverages the camera pose information, geometric prior and the learning ability.

Fourth, we detect the BEV bounding boxes by decoding the BEV features. The detected BEV bounding boxes are

$$\{\mathbf{o}_i\} = f_{\mathrm{decoder}}(\mathbf{F}^{(\mathrm{bev})}, \mathbf{A}^{(\mathrm{bev})})),$$

where $\mathbf{o}_i = (x_i, y_i, w_i, l_i, \theta_i, a_i, c_i)$ is the ith box with (x_i, y_i) the center, w_i the width, l_i the length, θ_i the azimuth angle, a_i the altitude level and c_i the object category. Our decoder follows the established detector CenterNet [20].

C. Categorical altitude estimation

Here we elaborate on the details of the proposed categorical altitude estimation module (1). The specific goal is to localize the object in z-axis. Categorical altitude estimation is critical for 3D detection in aerial view as the observed ground is broader and not flat, and allows to place 2D image information at the correct 3D locations, providing side information for precise warping. It is challenging because the aerial perspective encounters severe long-range issues, where the altitude difference across objects is relatively minor compared to the distance between objects and the drone. It is almost impossible to accurately localize the objects in a continuous manner. To alleviate the difficulty of altitude estimation, we categorize the altitude into multiple bins and substitute the regression task with a classification task.

To realize this, we first estimate the altitude level at each RV pixel. Intuitively, image context can provide rich altitude hints. For example, since most vehicles have similar sizes, they look bigger when they are closer to the drone and smaller when far away from the drone. The RV map of altitude confidence is obtained as

$$\mathbf{A}^{(\mathrm{rv})} = \mathrm{conv}(\mathbf{F}^{(\mathrm{rv})}) \in \mathbb{R}^{H_R \times W_R \times Z},$$

where $\mathrm{conv}(\cdot)$ is a 1×1 convolution layer. Each element in $\mathbf{A}^{(\mathrm{rv})}$ reflects the probabilities over all the altitude levels at each pixel in the RV. We next geometrically transform the RV map of altitude confidence to the BEV coordinate by

$$\mathbf{A}^{(\mathrm{bev})} = t(\mathbf{A}^{(\mathrm{rv})}) \in \mathbb{R}^{X \times Y \times Z},$$

where the geometric transformation $t(\cdot)$ is fully derived from the camera pose provided with the input image. The output $\mathbf{A}^{(\mathrm{bev})}$ reflects the altitude information at each BEV location.

D. Geo-deformable transformation

The proposed geo-deformable transformation aims to learn the BEV feature for the 3D object detection given the RV feature; see (2). This is required to infer the 3D scene given the planar 2D image; however, the altitude information is not available without an extra depth sensor. Theoretically, each image pixel is the projection of a line across all the altitudes in the 3D space, causing ambiguities.

To compensate for the missing altitude information, we consider solutions from two aspects: 1) weighting features along the z axis; 2) deforming features along with the x, y axes. First, we leverage the geometric transformation derived from the camera pose to generate BEV representations at all possible altitudes; and then, weight them with the estimated altitude confidence. The weighted BEV representations are averaged along the altitude axis, collapsing to a flat BEV feature. Second, we use a trainable deformable convolutional network (DCN) to adaptively revise the distortion in the BEV feature caused by the imprecise altitude estimation, promoting flexibility in this view transformation phase. DCN is capable of augmenting the spatial sampling locations with additional offsets, which could help to finetune the geometrically transformed feature. A residual structure is

applied to combine information from both the geometric transformation and the adaptively deformable transformation.

Geometric transformation. The geometric transformation is a non-parametric approach for view transformation. The camera projection matrix $\mathbf{P}$ defines the mapping between the global coordinate $(x, y, z) \in \mathbb{R}^3$ to the local image pixel coordinate $(u, v) \in \mathbb{R}^2$. The geometric transformation of altitude z could be denoted as the following mapping:

$$z \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{P} \begin{bmatrix} x \\ y \\ z \end{bmatrix}.$$

The RV feature $\mathbf{F}^{(\mathrm{rv})} \in \mathbb{R}^{H_R \times W_R \times C}$ is transformed to all the Z possible altitudes, which are stacked along the z-axis and produces the 3D feature $\mathbf{G}^{(\mathrm{bev})} \in \mathbb{R}^{X \times Y \times Z \times C}$. Given the feature $\mathbf{G}$ and the altitude confidence $\mathbf{A}^{(\mathrm{bev})}$ in BEV, via weighting and collapsing along the altitude, we can get the flatten BEV feature $\mathbf{F}_g^{(\mathrm{bev})} \in \mathbb{R}^{X \times Y \times C}$ as follows,

$$\mathbf{F}_g^{(\mathrm{bev})}(x,y) = \frac{1}{Z} \sum_{z=0,1,\ldots,Z-1} \mathbf{G}^{(\mathrm{bev})}(x,y,z) \cdot \mathbf{A}^{(\mathrm{bev})}(x,y,z),$$

where z is the index of the altitude level, $\mathbf{G}^{(\mathrm{bev})}(x,y,z)$ is possible feature representation at the coordinate (x,y,z). $\mathbf{A}^{(\mathrm{bev})}(x,y,z)$ is the confidence that the altitude value of the feature point at the coordinate (x,y) ranges in the z-th altitude level. Here, the 3D feature $\mathbf{G}^{(\mathrm{bev})}$ contains the BEV feature across all the possible altitudes. As stated in [21], BEV grids greatly reduce the computational overhead while offering similar performance to 3D voxel grids. So we use the flatten BEV feature while keeping the relative importance in the altitude levels to perform 3D object detection.

Deformable transformation. Through geometric transformation across all the possible altitudes, we get flat yet 'stereo' BEV feature. The non-parametric geometric transformation lacks learnable flexibility. Ideally, if the altitude is precisely estimated, the BEV feature could exactly represent the real world. However, the severe long-range issue along with the aerial view makes the altitude estimation especially difficult. Therefore, the geometrically transformed BEV feature $\mathbf{F}_g^{(\mathrm{bev})}$ is expected to encounter spatial sampling noise.

To promote better transformation, a DCN layer is cascaded to augment the geometric spatial sampling with trainable offsets. We further concatenate the coordinates with the BEV feature to guide offset learning. Since the coordinates could hint at the network with the geometric prior that the perception field is increasing with the distance between objects and the camera, which means that the perturbation area is large at far distance, and vice versa. Mathematically, the BEV feature map after the deformable convolution is

$$\mathbf{F}_d^{(\mathrm{bev})} = \mathrm{DCN}([\mathbf{F}_g^{(\mathrm{bev})}; \mathbf{X}; \mathbf{Y}]),$$

where $\mathrm{DCN}(\cdot)$ is the trainable DCN layer, $[;]$ denotes concatenation, $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{X \times Y \times 1}$ refer to the x, y coordinates of the feature points, reflecting the geometric prior.

Finally, we use a residual structure that combines the geometrically transformed feature $\mathbf{F}_g^{(\mathrm{bev})}$ and the adaptively
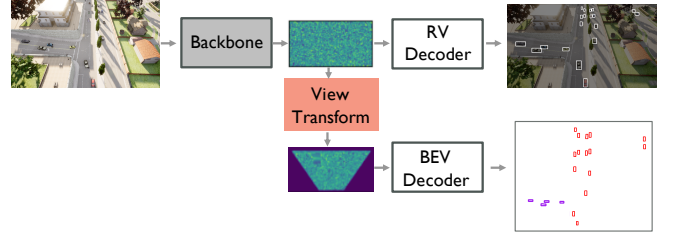

Fig. 4. DVDET simultaneously localizes objects in image and 3D space.

deformable feature $\mathbf{F}_d^{(\mathrm{bev})}$ to get the final BEV feature map, $\mathbf{F}^{(\mathrm{bev})} = \mathbf{F}_g^{(\mathrm{bev})} + \mathbf{F}_d^{(\mathrm{bev})}$.

E. Loss function

To train the overall system, we consider two loss functions to supervise two tasks: the altitude-level classification and the BEV-based object detection. For altitude-level classification, let $\mathbf{A}^{(\mathrm{bev})}$ be the estimated altitude category, $\widetilde{\mathbf{A}}^{(\mathrm{bev})}$ is the ground-truth altitude category, $\mathbf{M}$ is the objectiveness mask, only the foreground objects are supervised, the classification loss is then $L_{\mathrm{altitude}} = \mathbf{M} \odot \mathrm{Focal}(\mathbf{A}^{(\mathrm{bev})}, \widetilde{\mathbf{A}}^{(\mathrm{bev})})$. Focal loss [22] is used to alleviate the class imbalance issues. For BEV object detection, we follow CenterNet [20] and jointly optimize the classification and regression losses. Let $\mathbf{H}$ be the estimated category heatmap, $\widetilde{\mathbf{H}}$ be the ground-truth heatmap, the classification loss is $L_{\mathrm{cls}} = \mathrm{Focal}(\mathbf{H}, \widetilde{\mathbf{H}})$. Let (x, y, w, l, θ) be a detected box and $(\widetilde{x}, \widetilde{y}, \widetilde{w}, \widetilde{l}, \widetilde{\theta})$ be the ground-truth box, ℓ_1 loss $\|\cdot\|_1$ is used for the regression loss

$$\begin{aligned} L_{\mathrm{box}} &= \|x - \widetilde{x}\|_1 + \|y - \widetilde{y}\|_1 + \|w - \widetilde{w}\|_1 + \left\|l - \widetilde{l}\right\|_1 \\ &+ \left\|\sin\theta - \sin\widetilde{\theta}\right\|_1 + \left\|\cos\theta - \cos\widetilde{\theta}\right\|_1. \end{aligned}$$

The overall loss is the addition of L_{altitude}, L_{cls} and L_{box}.

F. Simulation to real-world transfer

To alleviate expensive and laborious real-world data collection/annotation, we adopt a Sim2Real training regime, where the model is pre-trained on simulation data (AM3D-Sim), and followed by fine-tuning on real data (AM3D-Real). Specifically, to minimize the domain gap, which can be triggered by the inconsistency between physical parameters, *e.g.* illumination, reflection, etc, we take inspiration from the idea of **domain randomization** [23], conducting aggressive color augmentations on the simulation data, and train the model to be invariant to all of them. Likely this model can adapt to the real-world environment, as the real visual scene is expected to be one sample in that rich distribution of training variations.

VI. DUAL-VIEW OBJECT DETECTION SYSTEM

We further propose a dual-view object detection system, termed as *DVDET*, which simultaneously perceive the objects in the 2D image space and the 3D physical space, based on the intuition that the two views could potentially promote each other. Specifically, the 2D image space can provide object details, such as color and shape, and help object understanding, while the 3D space can provide more accurate spatial information. The implicit consistency from

Method	Trans-Location			Trans-Type		Altitude Estimate	BEV Object Detection				Altitude Classification			
	Early	Inter	Late	Geo	Def	CAE	Fullset	Town1	Town2	Town3	Fullset	Town1	Town2	Town3
Late-GeoT	-	-	✓	✓	-	-	0.62/2.50/0	1.13/5.89/0.05	0.37/2.38/0	0.07/0.48/0	-	-	-	-
Early-GeoT	✓	-	-	✓	-	-	37.36/81.75/27.88	37.43/78.49/30.71	42.62/90.11/33.40	32.27/75.84/20.72	-	-	-	-
Inter-GeoT	-	✓	-	✓	-	-	38.64/80.52/32.19	38.85/77.06/34.94	43.19/88.20/36.58	34.17/75.42/26.15	-	-	-	-
Inter-GeoDT	-	✓	-	✓	✓	-	40.84/81.85/36.37	43.39/80.32/42.67	44.11/88.32/38.46	36.02/76.79/30.05	-	-	-	-
Inter-GeoT-CAE	-	✓	-	✓	-	✓	41.54/82.94/37.31	43.27/80.56/42.35	44.37/88.11/39.26	37.65/79.61/31.96	88.56	74.55	93.54	83.54
DVDET	-	✓	-	✓	✓	✓	**42.70/84.57/38.37**	**45.31/82.94/44.93**	**45.06/89.43/39.41**	**38.78/81.07/33.17**	**90.36**	**79.02**	**94.24**	**86.36**

the supervision of the two views, including RV and BEV, can thus help reduce the error of each other and promote more precise detection. Fig. 4 illustrates *DVDET*. The detectors for two views share the same backbone. The RV decoder localizes objects in the 2D image space and the BEV decoder localizes objects in the 3D space by using the proposed categorical altitude estimation and geo-deformable transformation methods.

VII. EXPERIMENTAL RESULTS

A. Implementation details

Our detector follows the CenterNet [20] with DLA-34 [19] backbone. The RV aerial image size is $(800, 450)$ and $(720, 480)$ in the simulation and real-world dataset. The resolution of the BEV is 0.25m/pixel. We transform the pyramid RV feature maps to BEV. The size of the BEV feature map is $(192, 352)$ and $(96, 128)$ in the simulation and real-world datasets respectively. We employ the generic detection evaluation metric: Average Precision (AP) at Intersection-over-Union (IoU) thresholds of 0.5 and 0.75.

B. Overall performance

Evaluation on AM3D-Sim. We first compare baselines that directly transform the three different information sources: detected objects, input image, or intermediate feature from RV to BEV according to the given camera pose information, as shown in Tab. II. *Early-GeoT*, *Inter-GeoT* and *Late-GeoT* denotes transforming the raw RV image, intermediate feature and the detection output, respectively. Here *GeoT* is short for geometric transformation, fully derived from the camera pose. We see that: i) *Inter-GeoT* performs the best; ii) *Early-GeoT* performs slightly inferior to *Inter-GeoT*. Since compared to *Inter-GeoT*, the input of *Early-GeoT* has higher resolution, however, it encounters irreparable deformation, which could be alleviated in *Inter-GeoT*. Overall, *Inter-GeoT* has more flexibility and performs the best. iii) *Late-GeoT* performs poorly. Directly transforming the RV detection fails the BEV object detection. This might be caused by two reasons: first, objects in the RV are mostly represented with an axis-aligned bounding box, which could not precisely represent the objects in the local coordinate; second, there are many tiny objects from the aerial perspective, occupying only 0.146% of the image on average. So, the view deformation along with the aerial perspective severely degrades the transformed BEV object detection performance.

System	BEV			RV		
	AP	AP@50	AP@75	AP	AP@50	AP@75
RV	-	-	-	56.70	93.40	61.70
BEV	42.70	84.57	38.37	-	-	-
Dual-View	**43.27**	**84.83**	**39.76**	**57.80**	**93.50**	**65.10**

We next validate the other two proposed modules, namely, geo-deformable transformation (*GeoDT*) and categorical altitude estimation (*CAE*). Building on *Inter-GeoT*, *Inter-GeoDT* integrates *deformable* convolutions into the geometric transformation and *DVDET* further considers all the possible altitudes. As shown in Tab. II: i) *Inter-GeoDT* consistently performs better than *Inter-GeoT* and improves by 12.99% on AP@75 on the fullset, reflecting the effectiveness of the proposed geo-deformable transformation; ii) *Inter-GeoT-CAE* improves *Inter-GeoT* by 7.50% on AP on the fullset and *DVDET* improves *Inter-GeoDT* by 4.54%, reflecting the effectiveness of the proposed categorical altitude estimation. Fig. 5 shows that i) BEV detection performance degrades with the altitude, which means the detection difficulty is increasing along with the altitude; ii) the improvement of *DVDET* over *Inter-GeoDT* is stable across all the altitudes, which means that *DVDET* is robust and alleviates the severe deformation issues at high altitude.

We further validate the feasibility of the dual view object detection in Tab. III. We see that: i) dual-view outperforms the individual RV and BEV by 1.1 and 0.57 on AP, respectively. It means that the two views can provide complementary information and promote each other: BEV can alleviate the occlusion in the RV, while RV can provide more object details and more smooth image context to help alleviate the deformation in BEV. Fig 7 presents quantitative results on AM3D-Sim. We see that: i) the proposed system accurately detects most of the objects in dual-views; ii) the occlusion and overlapping between objects are alleviated in BEV; iii) as the right sample shows, the objects on the ramp can be accurately detected. Note that the presented BEV images are the RV images transformed to the ground plane. It only provides a rough idea about BEV object detection and cannot accurately represent the real situation, especially for the areas with varying altitudes.

Evaluation on AM3D-Real. We further validate the proposed modules and dual-view system on the real-world dataset with abundant scenes and high-quality annotations, which is relatively smaller than the simulation one due to the

TABLE IV

OVERALL PERFORMANCE ON AM3D-REAL. * DENOTES WITH DOMAIN TRANSFER FROM SIMULATION TO REAL DATA.

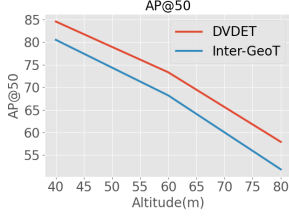| System	Method	BEV			RV		
		AP	AP@50	AP@75	AP	AP@50	AP@75
RV	-	-	-	-	39.90	84.70	30.00
BEV	Inter-GeoT	22.67	62.26	9.76	-	-	-
	Inter-GeoDT	23.89	65.04	11.50	-	-	-
	Inter-GeoT-CAE	24.13	66.41	10.71	-	-	-
	DVDET	26.79	69.08	13.89	-	-	-
	Inter-GeoT*	25.14	60.52	17.29	-	-	-
	Inter-GeoDT*	27.11	65.95	17.05	-	-	-
	Inter-GeoT-CAE*	28.19	68.79	16.30	-	-	-
	DVDET*	29.04	**72.66**	15.09	-	-	-
Dual-View	DVDET	27.39	68.46	14.43	41.60	84.80	34.10
	DVDET*	**31.82**	72.60	**21.40**	**43.50**	**85.90**	**35.90**

Fig. 5. *DVDET* is robust and could alleviate the severe deformation issues at high altitude.
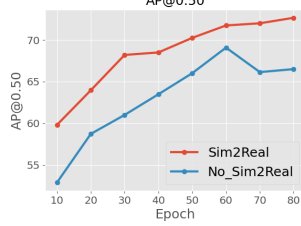

Fig. 6. Simulation data benefits BEV object detection on real data with domain transfer.

TABLE V

ABLATION STUDIES ON THE TRANSFORMATION TYPES. THE GEO-DEFORMABLE VARIANT *Geo-DADCN* PERFORMS THE BEST.

Transformation		AP	AP@50	AP@75
Geometric	*Geo*	38.64	80.52	32.19
Deformable	*MLP*	23.56	58.16	14.97
Geo-Deformable	*Geo-MLP*	24.45	59.16	16.26
	Geo-DCN	40.23	81.83	34.83
	Geo-DADCN	**42.70**	**84.57**	**38.37**

TABLE VI

ABLATION STUDIES ON THE CATEGORICAL ALTITUDE ESTIMATION. CATEGORICAL ESTIMATION IMPROVES 5.50% ON AP@75.

Altitudes	Estimation	Supervison	AP	AP@50	AP@75
Single	-	-	40.84	81.85	36.37
Multiple	Categorical	-	40.94	81.7	36.85
	Categorical	✓	**42.70**	**84.57**	**38.37**
	Continuous	✓	40.69	83.21	35.08

TABLE VII

DVDET OUTPERFORMS 3D DETECTION SOTAS FOR CARS AND 2D DETECTION SOTAS FOR FLAT BEV IMAGE ON AERIAL 3D DETECTION.

AP for method	3D detection for cars		2D detection		DVDET
	MonoRCNN	CaDDN	Faster-RCNN	SwinT	
AM3D-Sim	0.41	41.54	20.05	24.00	**43.27**
AM3D-Real	0.00	24.13	12.40	8.20	**31.82**

TABLE VIII

DVDET OUTPERFORMS PREVIOUS SOTAS BY 15.32% ON KITTI, A WELL-KNOWN AUTONOMOUS DRIVING BENCHMARK.

Method	$AP|_{R_{40}}$ [Easy / Mod / Hard] ↑	
	AP_{3D}	AP_{BEV}
FQNet(CVPR19) [24]	2.77/1.51/1.01	5.40/3.23/2.46
M3D-RPN(ICCV19) [25]	14.76/9.71/7.42	21.02/13.67/10.23
MonoPair(CVPR20) [26]	13.04/9.99/8.65	19.28/14.83/12.89
MoVi-3D(ECCV20) [27]	15.19/10.90/9.26	22.76/17.03/14.85
RTM3D(ECCV20) [28]	14.41/10.34/8.77	19.17/14.20/11.99
MonoRCNN(ICCV21) [11]	18.36/12.65/10.03	25.48/18.11/14.10
CaDDN(CVPR21) [14]	19.17/13.41/11.46	-/-/-
GUP Net(ICCV21)[29]	20.11/14.20/11.77	-/-/-
DVDET	**23.19/15.44/13.07**	**32.05/22.15/19.32**

costly collection and annotation. To mitigate this, we apply the domain adaptation technique to transfer the pre-trained system on the simulation data to handle real-world data. From Tab. IV, we see that: i) both the proposed module: geo-deformable transformation (GeoDT), and categorical altitude estimation (CAE) achieve improvements, the in-line results with cognition validate that the proposed system is robust and the proposed dataset is trustworthy; ii) dual-view outperforms the individual RV and BEV; iii) the simulation data could effectively help to improve the real-world detection performance. Fig. 6 shows that the detector pre-trained on simulation data consistently enables a better initialization for the real-world data. Fig. 8 presents qualitative results. We see: i) the proposed system could accurately detect most of the objects in dual-views; ii) the occlusion and overlapping between objects are alleviated in BEV.

C. Ablation studies

We provide ablation studies to validate our design choices, and all experiments are conducted on AM3D-Sim.

Effect of geo-deformable transformation. Tab. V explores the properties of geometric and deformable transformation with multiple variants. We see that: i) the geometric transformation *Geo* achieves stable but inferior performance; ii) the purely learnable transformation without any geometric guidance *MLP* fails; iii) the improperly introduced deformable flexibility *Geo-MLP* severely harms the transformation; iv) a well-designed deformable module *Geo-DCN* and *Geo-DADCN* boost the detection performance; v) the distance prior could help DCN learn the offset, and improve the AP from 40.23 to 42.70. To sum up, the proposed geo-deformable transformation enjoys both stability of geometric transformation and the flexibility of learnable transformation.

Effect of categorical altitude estimation. Tab. VI assesses the effectiveness of categorical altitude estimation. We

see that: i) without supervision, the BEV representation with multiple altitudes shows similar performance as its single version. The minor difference among altitudes is difficult to catch; and ii) with the proper altitude guidance, the augmented BEV representation across multiple altitudes could achieve superior performance, the overall AP could achieve 42.70; iii) in the same setting, continuously regressing is clearly worse than categorically binning (40.69 vs. 42.70).

D. Generalization to autonomous driving

We further validate the proposed modules by comparing with previous SOTAs on AM3D-Sim and the KITTI 3D object detection benchmark [8]. Tab. VII shows *DVDET* clearly outperforms 3D autonomous driving and 2D detection SOTAs on both simulation and real datasets. Note: i) MonoRCNN [11] and CaDDN [14] are worse than *DVDET* as they do not consider severe view deformation; while we alleviate this by the proposed geo-deformable transformation; ii) Faster-RCNN [30] and SwinT [31] on 2D BEV images are worse than *DVDET* as they consider 3D scenes with flat images; while *DVDET* considers 3D features with the proposed categorical altitude estimation technique. Tab. VIII shows *DVDET* clearly outperforms previous 3D autonomous
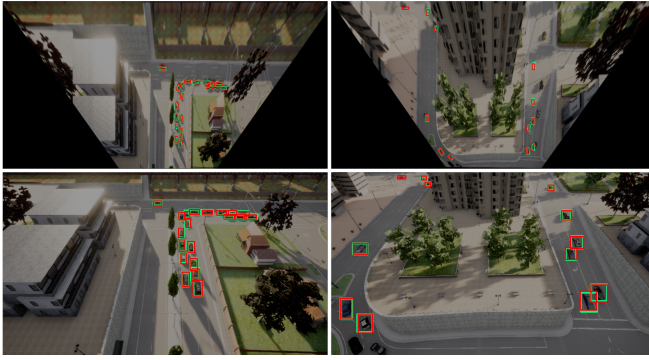
Fig. 7. The qualitative results of DVDET on AM3D-Sim. The upper row shows the BEV results and the bottom row shows the RV results. The ground-truth are colored green and the predictions are colored red.

Fig. 8. The quantitative results of DVDET on AM3D-Real. The upper row shows the BEV results and the bottom row shows the RV results.

driving SOTAs on KITTI. Note: i) *DVDET* benefits monocular 3D object detection for autonomous driving; ii) our improvement (15.32%) is significantly superior to previous SOTA GUP Net(ICCV21)[29] (4.90%).

VIII. CONCLUSION

To address the problem of aerial monocular 3D object detection, this paper proposes a new dataset, including both simulation (AM3D-Sim) and real-world (AM3D-Real) dataset, as well as a novel monocular 3D object detection system, DVDET, with two core techniques: categorical altitude estimation and geo-deformable transformation. Extensive experiments show that i) DVDET significantly outperforms baseline methods on AM3D-Sim and AM3D-Real, reflecting the effectiveness of the 3D scene understanding from aerial perspective; ii) the model pre-trained on the simulation dataset benefits real-world performance; and iii) DVDET achieves the leading performance on KITTI, reflecting that the proposed method also benefits autonomous driving.

REFERENCES

[1] S. Manfreda, M. F. McCabe, P. E. Miller, R. Lucas, V. Pajuelo Madrigal, G. Mallinis, E. Ben Dor, D. Helman, L. Estes, G. Ciraolo, *et al.*, "On the use of unmanned aerial systems for environmental monitoring," *Remote sensing*, vol. 10, no. 4, p. 641, 2018.

[2] J. Ding, N. Xue, G.-S. Xia, X. Bai, W. Yang, M. Y. Yang, S. Belongie, J. Luo, M. Datcu, M. Pelillo, *et al.*, "Object detection in aerial images: A large-scale benchmark and challenges," *arXiv:2102.12219*, 2021.

[3] D. Du, Y. Qi, H. Yu, Y. Yang, K. Duan, G. Li, W. Zhang, Q. Huang, and Q. Tian, "The unmanned aerial vehicle benchmark: Object detection and tracking," in *ECCV*, 2018, pp. 370–386.

[4] S. Shah, D. Dey, C. Lovett, and A. Kapoor, "Airsim: High-fidelity visual and physical simulation for autonomous vehicles," in *Field and service robotics*. Springer, 2018, pp. 621–635.

[5] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "Carla: An open urban driving simulator," in *Conference on robot learning*. PMLR, 2017, pp. 1–16.

[6] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *CVPR*, 2017.

[7] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *ICCV*, 2017, pp. 764–773.

[8] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *CVPR*. IEEE, 2012.

[9] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *CVPR*, 2020.

[10] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, *et al.*, "Scalability in perception for autonomous driving: Waymo open dataset," in *CVPR*, 2020.

[11] Q. Y. Xuepeng Shi, Z. C. Xiaozhi Chen, Chuangrong Chen, and T.-K. Kim, "Geometry-based distance decomposition for monocular 3d object detection," in *ICCV*, 2021.

[12] Y. Wang, W.-L. Chao, D. Garg, B. Hariharan, M. Campbell, and K. Q. Weinberger, "Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving," in *CVPR*, 2019.

[13] X. Chen, K. Kundu, Z. Zhang, H. Ma, S. Fidler, and R. Urtasun, "Monocular 3d object detection for autonomous driving," in *Proceedings of the CVPR*, 2016, pp. 2147–2156.

[14] C. Reading, A. Harakeh, J. Chae, and S. L. Waslander, "Categorical depth distribution network for monocular 3d object detection," in *CVPR*, 2021, pp. 8555–8564.

[15] J. Philion and S. Fidler, "Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d," in *ECCV*. Springer, 2020, pp. 194–210.

[16] R. Rombach, P. Esser, and B. Ommer, "Geometry-free view synthesis: Transformers and no 3d priors," *arXiv:2104.07652*, 2021.

[17] Y. Hou, L. Zheng, and S. Gould, "Multiview detection with feature perspective transformation," in *ECCV*, 2020.

[18] Q. Li, Y. Wang, Y. Wang, and H. Zhao, "Hdmapnet: A local semantic map learning and evaluation framework," *arXiv:2107.06307*, 2021.

[19] F. Yu, D. Wang, E. Shelhamer, and T. Darrell, "Deep layer aggregation," in *Proceedings of the CVPR*, 2018, pp. 2403–2412.

[20] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," in *arXiv:1904.07850*, 2019.

[21] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *CVPR*, 2019, pp. 12 697–12 705.

[22] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *ICCV*, 2017.

[23] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," *IROS*, pp. 23–30, 2017.

[24] L. Liu, J. Lu, C. Xu, Q. Tian, and J. Zhou, "Deep fitting degree scoring network for monocular 3d object detection," in *CVPR*, 2019.

[25] G. Brazil and X. Liu, "M3d-rpn: Monocular 3d region proposal network for object detection," in *ICCV*, 2019, pp. 9287–9296.

[26] Y. Chen, L. Tai, K. Sun, and M. Li, "Monopair: Monocular 3d object detection using pairwise spatial relationships," in *CVPR*, 2020.

[27] A. Simonelli, S. R. Bul, L. Porzi, E. Ricci, and P. Kontschieder, "Towards generalization across depth for monocular 3d object detection," in *ECCV*, 2020.

[28] P. Li, H. Zhao, P. Liu, and F. Cao, "Rtm3d: Real-time monocular 3d detection from object keypoints for autonomous driving," in *ECCV*, 2020.

[29] Y. Lu, X. Ma, L. Yang, T. Zhang, Y. Liu, Q. Chu, J. Yan, and W. Ouyang, "Geometry uncertainty projection network for monocular 3d object detection," in *ICCV*, 2021.

[30] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Neurips*, 2015.

[31] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," *arXiv:2103.14030*, 2021.