Exploring 2D Data Augmentation for 3D Monocular Object Detection

Sugirtha T\textsuperscript{1}, Sridevi M\textsuperscript{1}, Khailash Santhakumar\textsuperscript{2}, B Ravi Kiran\textsuperscript{3}, Thomas Gauthier\textsuperscript{3} and Senthil Yogamani\textsuperscript{4}

\textsuperscript{1}NIT Tiruchirappalli, India \quad \textsuperscript{2}SASTRA University, India \quad \textsuperscript{3}Navya, France \quad \textsuperscript{4}Valeo, Ireland

Fig. 1: Illustration of different 2D data augmentations for Monocular 3D Object Detection. Red boxes highlight cutout regions and blue boxes denote object annotations which are used to mix or paste.

Abstract—Data augmentation is a key component of CNN based image recognition tasks like object detection. However, it is relatively less explored for 3D object detection. Many standard 2D object detection data augmentation techniques do not extend to 3D box. Extension of these data augmentations for 3D object detection requires adaptation of the 3D geometry of the input scene and synthesis of new viewpoints. This requires accurate depth information of the scene which may not be always available. In this paper, we evaluate existing 2D data augmentations and propose two novel augmentations for monocular 3D detection without a requirement for novel view synthesis. We evaluate these augmentations on the RTM3D detection model firstly due to the shorter training times.

We obtain a consistent improvement by 4\% in the 3D AP (@IoU=0.7) for cars, 1.8\% scores 3D AP (@IoU=0.25) for pedestrians & cyclists, over the baseline on KITTI car detection dataset. We also demonstrate a rigorous evaluation of the mAP scores by re-weighting them to take into account the class imbalance in the KITTI validation dataset.

I. INTRODUCTION

3D object detection is crucial perception task in modern autonomous driving applications, used upstream for scene understanding, object tracking and trajectory prediction and decision making. Initially, autonomous cars are equipped with LiDAR sensors and most 3D detectors rely on LiDAR data to perform 3D object detection. LiDAR provides precise distance measure which makes it feasible to detect accurate 3D bounding boxes. But, they are expensive to be deployed in autonomous cars. Recent autonomous cars use single monocular camera and hence monocular 3D object detection (3D OD) became a research focus in computer vision community. 3D OD methods are usually part of the 4 following categories: (1) 2D proposal generation [1], (2) Geometric constraints [2], (3) Keypoints detection [3] or (4) Direct 3D proposals generation [4].

Convolutional Neural Networks (CNNs) learn complex hierarchical features from images. However, the models are susceptible to overfitting and require proper regularization techniques to be applied. Usual regularization techniques include data augmentation, i.e., image transformations such as scaling, translation, random flipping etc. In computer vision, data augmentation is pervasive due to its simplicity to implement and efficiency. In recent years, few regularization methods have been proposed to reduce overfitting of the model while training. This also improves the robustness of CNNs to learn complex features.

RTM3D[5] is a monocular 3D object detector that only applies flipping to augment the training samples. In this paper, we propose variants of 2D data augmentations such as Cutout, CutPaste and Mixup and try all possible combinations to find out the best suited combination in order to improve the detection accuracy over the baseline RTM3D detector model. We define a family of 2D data augmentations and their transformations on monocular images for 3D object detection. Also, we perform an ablation study on which our proposed augmentation methods improve the performance over the RTM3D baseline. 3D monocular detection requires an expensive annotation process. 3D detection methods have mostly been trained on large-scale datasets, while data augmentation methods do exist but have not been explored further as in case of YOLOv3[6] or modern object detectors. Our new data augmentation methods improve performance greatly on 3D object detection for occlusion, pixel corruption and better localization.
This paper evaluates an initial set of 2D data augmentation techniques for monocular 3D detection, without changing the 3D geometry of bounding boxes in the scene and avoiding the requirement to synthesize new viewpoints of the input camera view. In summary, the contributions of our paper are as follows:

- We evaluate the performance of photometric as well as region based 2D data augmentations for the monocular 3D object detection task.
- We propose three new 2D data augmentation strategies that significantly improve the detection accuracy of the baseline RTM3D detector namely Box-MixUp, Box-Cut-Paste, and Mosaic-Tile.

Extensive analysis on KITTI dataset demonstrates that our proposed augmentations improve performance under various conditions: occlusions, contrasted/shadowed pixels, changing the diversity of viewpoints of objects seen in the dataset.

II. RELATED WORK

2D OD on image plane is inadequate for reliable autonomous driving scenario because it does not provide an accurate estimation of 3D objects sizes and space localization. In other words, 2D OD methods have limited performance in following scenarios namely Occlusion, Object pose estimation and 3D position information. A 3D bounding box provides precise information about size of the object and its position in 3D space.

3D Object Detection: Recently, many researchers proposed various techniques for 3D Monocular Object Detection. RTM3D and its extension KM3D use CenterNet to regress a set of 9 projected keypoints corresponding to a 3D cuboid in image space. They also perform direct regression for the object’s distance, size and orientation. These values are then used for offline initialization of an optimizer to estimate 3D bounding boxes under geometric constraints. CenterNet provides basic data augmentation such as affine transformations (shifting, scaling) and random horizontal flipping. Over these, KM3D adds coordinate independent augmentation via random color jittering.

SMOKE regresses 3D bounding box directly from image plane which eliminates 2D bounding box regression. It represents an object by a single keypoint and these keypoints are projected as 3D center of each object. Mono3D is a region proposal based method that uses semantics, object contours and location priors to generate 3D anchors. It generates proposal by performing exhaustive search on 3D space and use Non-maximal suppression for filtering. SMOKE augments the training samples with horizontal flipping, scaling and shifting. GS3D predicts the guidance of cuboid and performs feature extraction by projecting region of guidance. GS3D performs monocular 3D detection without augmenting the training data. We observe that few detectors applied basic augmentations and few others applied none. In this paper, we propose data augmentations that can be deployed on the 3D detectors to improve their performances.

Advanced 2D Data Augmentations: Motivated by dropout regularization technique, few 2D data augmentations were proposed in recent years for object classification task. CutOut removes a square region from the input image and fills it with any of the following: (i) black pixels (ii) grey pixels (iii) Gaussian noise. In other words, it performs dropout but in the input space. This drives the model to spotlight on the whole image instead of focusing on a few key features. Still, there is a chance to loose informative pixels during training. Mixup performs linear interpolation of two randomly drawn images. Yet, the resultant images sometimes look unusual and it may confuse the model during localization task.

CutMix overcomes the limitations in CutOut and Mixup by replacing the square region with a patch of identical dimensions from another image. This results in no loss of informative pixels while training, besides preserving the advantages of regional dropout in input space. Also, the images formed are legitimate compared to Mixup. Smoothmix is proposed to minimize the “strong-edge” problem caused due to regional dropout. The model takes two images from training set and generates augmented images by performing element-wise addition of two images after an element-wise multiplication with a smoothly transitioning mask $G$.

All 2D data augmentations discussed above when applied to 3D space require synthesizing new view points of the input view and orientation of the bounding boxes will get distorted. In this paper, we extend existing 2D data augmentations without modifying the 3D geometry of bounding box and avoid synthesizing new view points for the input camera view.

YOLOv4 is a 2D object detector which which has improved object detection accuracy mainly thanks to adequate data augmentation such as Cutout, Mixup and CutMix. Additionally, they introduced the mosaic data augmentation to detect small objects. Mosaic data augmentation in YOLOv4 merges 4 training samples directly and applies random cropping to produce augmented image. The recent developments in 2D data augmentations suggest their potential utility for the monocular 3D detection task, while assuming some geometrical constraints.

3D augmentations with view synthesis: Both traditional and advanced data augmentations discussed above can be applied only in 2D image space. To handle data augmentations for objects in 3D, performing in varying viewpoints a view synthesis problem is required to be solved. Rol-10D lifts 2D RoIs to 3D for 6 DoF pose regression. Authors show that 3D synthetic data augmentation was achieved by inpainting 3D meshes directly onto the 2D scenes. MoCa - multi modality cut and paste was proposed to overcome the limitations of cut-paste that is applicable for single modality 3D detectors. MoCa is readily applicable for multi modality 3D object detection while preserving consistency...
TABLE I: List of data augmentations studied for 3D-MOD. * indicates proposed 2D transforms.

| Data augmentation Transforms | Description and Utilization |
|-----------------------------|-----------------------------|
| Cutout (holes, W)           | Cutout removes a randomly chosen patch from the image and those pixels are zeroed. Cutout parameters include the number of square holes and their side. |
| Photometric pixel-wise DA   | Standard pixel level data augmentations which apply motion blur, RGB shifts and Random contrast from the albu augmentations library [16]. |
| Box - MixUp*                | Generates an augmented (image, bounding box set) pairs by averaging the pixel values between the two images (reference & random samples) only within masks defined by the bounding box. |
| Box - Cut - Paste*          | A variation of Box-MixUp is local-cut-paste where regions of the 2D bounding box from the second image is pasted on to the reference image. |
| Mosaic-Tile*                | Augmented image is created by tiling/concatenating the 4 corresponding quadrants of four different training samples into one image. |

III. PROPOSED METHOD

Generally there are multiple families of 2D data augmentations that are employed in literature discussed below.

Pixel level data augmentation: Photometric augmentation includes changes to the pixel values such as contrast, blurring, brightness or color changes.

Regional data augmentation: We refer to data augmentations that mask, transform or change multiple locations in the image domain at the same time as regional data augmentations. Coarsedrop [16] is an example of region transform. Occlusion occurs because of regional augmentation and our proposed Cutout helps the detector to detect occluded objects more accurately. Hence it overcomes the effects due to region based augmentations and thus increasing the robustness of the detector. Motion blur is simulated by the blurring transformation which is regional augmentation.

Geometric data augmentation: transfers image pixels to new positions and hence changes the image geometry. This can include reflection, rotation, cropping, translation, scaling, flipping, etc.

A. Data augmentation (DA) methods

The input training image is represented by tensor $x \in \mathbb{R}^{W \times H \times 3}$ while the training bounding boxes coordinates by $y$ along with its class label. The data augmented samples are represented by $(\tilde{x}, \tilde{y})$. They are generated from reference images designated by $(x_A, y_A)$ and $(x_B, y_B)$ (image, bounding box set). The data augmentations we proposed in this paper are under one of the two categories:

- Data augmentations that transform only the input training image $x_A$ without modifying the ground truth boxes $y_B$. Ex. Cutout and Photometric transformations.
- Augmentations that modify both image and ground truth boxes. Ex. Box-Mixup

Finally, let $M \in \{0, 1\}^{W \times H}$ be a binary mask which is 1 for pixels included by the set of bounding boxes in ground truth $y$, and 0 otherwise. The list of data augmentations evaluated in this study are described in Table I and illustrated in Fig. 1.

Box-Mixup: Motivated by the work on Mixup data augmentation [12] which was applied to image level classification problems, we aim here to propose an equivalent for 2D object detection. Box-MixUp is thus proposed to augment an image with object patches from other images, thus providing the same advantages of MixUp but localized over multiple regions in the image. Box-MixUp generates an augmented sample pair by averaging pixels of two input images under a mask containing all bounding boxes from the second image, while concatenating both ground truth bounding box sets.

$$
\tilde{x} = (0.5x_A + 0.5x_B) \cdot M_B + x_A \cdot (1 - M_B)
$$

$$
\tilde{y} = y_A \cup y_B
$$

Box-Cut-Paste: Cut-Paste data augmentation [19] uses pre-existing masks and background scenes to artificially create an augmented dataset to train an object detection model. In Box-Cut-Paste, we propose not to use a separate mask. The transformation directly pastes pixels under a bounding box from a reference image onto another image. This further ensures the object context is preserved and avoids unrealistic scenarios possible in Cut-Paste, such as cars in the sky. This is because the cut paste is only done over bounding box masks already existing in source images from the dataset. There might be cases where objects could get pasted on walls or other unusual backgrounds, though the perspective of the object remains natural.

$$
\tilde{x} = x_A \cdot (1 - M_B) + x_B \cdot M_B
$$

$$
\tilde{y} = y_A \cup y_B
$$

Mosaic-Tile: YOLOV4 proposed by authors [15] introduces the Mosaic augmentation, which takes 4 training images in different contexts and tiles them into one. It then performs random cropping to produce an augmented image of size equal to the original training image size. The batch normalization evaluates activation values across 4 different images reducing the requirement for a large mini-batch size. Mosaic increases the scale diversity of objects by tiling images at different scales.

Our proposed Mosaic-Tile augmentation takes distinct parts from four different training images and combines them into one as shown in Fig. 2. To ensure that we do not
select bounding boxes from an input sample, which is outside its corresponding tile’s extent in the augmented image, we ensure that at least 40% of boxes area is within the corresponding tile. Ground truth boxes under this threshold and outside their tiles are eliminated from training to avoid overfitting (insufficient image pixels available for these boxes). In future work this threshold hyper-parameter shall be studied to obtain the best detection results. The advantage of our proposed Mosaic-Tile are quite similar to the original Mosaic augmentation. In our current evaluation we do not randomize the image coordinates at which we split the image and keep this to be the center of the image across all training. Here again, this a way to somehow preserve the scene geometry while combining different contexts.

IoU check : The Box-Mixup & Box Cut-Paste augmentations were evaluated under two different conditions.

- In the first case we enforce that no two boxes between the two input samples overlap by a large margin. This was achieved by evaluating the 2D-IoU between pairs of boxes coming from two different samples and ensuring the intersection was smaller than 0.4. That is \( \text{IoU}(B_i, B_j), \forall B_i \in y_A, B_j \in y_B \). If the IoU is too high, the incoming bounding box is rejected from the augmentation and the mixup/paste operation is not performed in the image domain.

- In the second case, we do not filter boxes using any IoU threshold and permit all possibilities.

In Table III the presence of IoU condition is designated by B+Box-Mixup (w iou) and its absence B+Box-Mixup (w/o iou).

IV. RESULTS

A. Baseline Model

RTM3D[5] is a real-time network that is based on the CenterNet architecture, which enables both fast training cycles and small inference times. Authors have already provided a set of baseline data augmentations which include flip, affine transformations, stereo dataset augmentation using the left/right images in the KITTI dataset. RTM3D architecture enables a faster evaluation of data augmentation strategies. It is important to note that the data augmentation methods proposed in this study are quite generic and can be employed to any other Monocular 3D object detection model. RTM3D and few other models had begun evaluating a simple set of 2D data augmentations, while there was no systematic study yet done, and was the motivation for this study.

B. Dataset and Implementation Details

We evaluate our data augmentations on the KITTI 3D detection benchmark which consists of 7,481 labeled training samples and 7518 unlabeled testing samples. Since the ground truth labels for the test set are not available, we evaluated our model by splitting the training set into 3711 training samples and 3768 validation samples. We experiment with ResNet-18 as backbone. We implemented our deep neural network in Pytorch and trained using Adam optimizer with learning rate of 1.25*1e-4 for 200 epochs. We trained our network with a batch size of 16. Our model achieved best speed with 33 FPS on a NVIDIA GTX 2080Ti GPU.

C. Results

The KITTI benchmark evaluates the models by Average Precision(AP) of each class (Car, Pedestrian and Cyclist) under easy, moderate and hard conditions. We report two official evaluation metrics for “Test set” for comprehensive evaluation: AP for 3D bounding boxes \( \text{AP}_{3D} \) and AP for Birds Eye View \( \text{AP}_{bev} \). We tabulate the results under two IoU thresholds for each class. IoU = 0.7 and IoU = 0.5 for Car category. IoU = 0.5 and IoU = 0.25 for Pedestrian and Cyclist categories. Comparison of baseline results with our proposed data augmentations is shown in Table II. IoU thresholds for Car are 0.7 while 0.25 for pedestrian and cyclists.

To further describe the results we use two other evaluation metrics already known in literature: First, the Mean Average Precision (mAP) The mean value of the Average Precision (AP) over all classes :

\[
\text{mAP}_{3D} = \frac{1}{|C|} \sum_{c \in C} \text{AP}_c
\]

where \( C = \{\text{car, pedestrian, cyclist}\} \). Second we introduce the, Inverse Class Frequency Weighted (ICFW) mAP metric. This new metric was introduced to demonstrate the class imbalance inherent in the KITTI dataset. The relative frequency (denoted by \( f_c \) and in blue) of car, pedestrian and cyclist classes in the validation classes for the different difficulty
TABLE III: mAP and ICFW mAP scores for both 3D and BEV detection bounding boxes at various IoUs. The values of different data augmentations below the baseline correspond to the difference. Green refers to positive gains, while red refers to negative drops in performance.

| IoU = 0.25 | mAP<sub>3D</sub> | mAP<sub>BEV</sub> | ICFW mAP<sub>3D</sub> | ICFW mAP<sub>BEV</sub> |
|------------|-----------------|-----------------|-----------------|-----------------|
| Baseline (B) | 32.60 | 25.63 | 24.57 | 40.87 | 32.74 | 29.19 | 15.16 | 19.63 | 16.59 | 27.01 | 19.63 | 16.59 |
| B+Box-Mixup (w/o iou) | -0.50 | 0.80 | -0.83 | 1.43 | 0.56 | 0.78 | 1.05 | 1.80 | 1.85 | 0.84 | 1.07 | 1.04 |
| B+Box-Mixup (w/o iou) | -0.10 | 1.89 | 0.29 | -0.26 | -0.32 | 0.50 | 0.09 | -1.74 | 0.44 | -0.88 | -1.74 | 0.44 |
| B+Box-CutPaste (w/o iou) | -6.17 | -3.60 | -5.32 | -8.19 | -5.50 | 0.84 | -2.84 | 0.31 | 1.60 | 1.53 | 0.31 | 1.60 |
| B+Box-CutPaste (w/ iou) | -0.50 | 1.84 | 0.22 | 0.88 | 0.54 | 0.91 | 0.15 | 0.65 | 1.60 | 2.33 | 0.65 | 1.60 |
| IoU = 0.5 | mAP<sub>3D</sub> | mAP<sub>BEV</sub> | ICFW mAP<sub>3D</sub> | ICFW mAP<sub>BEV</sub> |
|------------|-----------------|-----------------|-----------------|-----------------|
| Baseline (B) | 32.78 | 25.63 | 24.57 | 40.87 | 32.74 | 29.19 | 15.16 | 19.63 | 16.59 | 27.01 | 19.63 | 16.59 |
| B+Box-Mixup (w/o iou) | -0.04 | -0.59 | -0.78 | 1.80 | 0.60 | 0.71 | 1.17 | 1.85 | 1.85 | 0.84 | 1.07 | 1.04 |
| B+Box-Mixup (w/ iou) | -0.04 | 0.41 | 0.21 | -0.11 | -0.31 | 0.44 | 0.12 | -1.74 | 0.43 | -0.85 | -1.74 | 0.43 |
| B+Box-CutPaste (w/o iou) | -6.52 | -5.29 | -5.00 | -9.26 | -5.75 | 0.69 | -2.91 | 0.26 | 1.58 | 1.30 | 0.26 | 1.58 |
| B+Box-CutPaste (w/ iou) | -0.39 | 0.39 | 0.23 | 0.45 | 0.42 | 0.71 | 0.17 | 0.62 | 1.58 | 2.24 | 0.62 | 1.58 |
| IoU = 0.7 | mAP<sub>3D</sub> | mAP<sub>BEV</sub> | ICFW mAP<sub>3D</sub> | ICFW mAP<sub>BEV</sub> |
|------------|-----------------|-----------------|-----------------|-----------------|
| Baseline (B) | 32.78 | 25.63 | 24.57 | 40.87 | 32.74 | 29.19 | 15.16 | 19.63 | 16.59 | 27.01 | 19.63 | 16.59 |
| B+Box-Mixup (w/o iou) | -1.16 | 1.10 | 1.18 | 1.37 | 1.48 | 0.96 | -0.26 | -0.27 | -0.10 | -0.04 | -0.09 | -0.20 |
| B+Box-Mixup (w/ iou) | -1.49 | 1.52 | 1.63 | 1.36 | 1.25 | 1.29 | 0.51 | 0.52 | 0.69 | 0.40 | 0.37 | 0.38 |
| B+Box-CutPaste (w/o iou) | 0.92 | 1.06 | 1.23 | 1.14 | 1.01 | 1.02 | -0.08 | -0.06 | 0.12 | 0.61 | 0.44 | 0.44 |
| B+Box-CutPaste (w/ iou) | 0.76 | 1.12 | 1.29 | 0.62 | 0.87 | 0.92 | 0.15 | -0.07 | 0.12 | 0.41 | 0.08 | 0.09 |

levels (based on thresholds over object occlusion levels) are shown in Table IV. To reduce the effect of the dominant class (here the car) and observe which data augmentation method produces gains in detection in least represented class, we introduce the inverse class weighting. This is evaluated by the following formula:

\[
 w_c := \frac{1}{\sum_{c \in C} f_c} \quad [0, 1] \quad \text{and} \quad \sum_{c \in C} w_c = 1
\]

The values of \( w_c \) are shown in Table IV in red. Now the ICFW mAP is evaluated as:

\[
 \text{ICFW mAP}_{3D} = \sum_{c \in C} w_c \cdot \text{mAP}_{c}
\]

**D. Analysis**

From Table II, it is clear that Box - Mixup outperforms all other augmentations for “Car” class in terms of \( \text{AP}_{3D} \) and \( \text{AP}_{BEV} \) for both IoU = 0.7 and 0.5. Cutout-2 holes is best suited augmentation for “Pedestrian” class for \( \text{AP}_{3D}(\text{IoU} = 0.5) \) and \( \text{AP}_{BEV}(\text{IoU} = 0.5 \& 0.25) \). For “Cyclist” class, different augmentations suit different conditions.

Table III shows different combinations of the proposed augmentations experimented on KITTI dataset. It tabulates the average mAP for each difficulty level (E, M and H) over all classes for IoU = 0.5. The table shows in red cases where the data augmentation performed poorly, and in green where it performed better, with each entry showing the difference w.r.t the baseline.

We observe that even though the mAP scores improvements might be negligible or negative in for some data augmentations the ICFW mAP scores reflect the improvement in performance reweighted due to reflect poor pedestrian and cyclist class frequencies.

The mAP of the RTM3D detector is improved reasonably with our proposed data augmentations. We find that our proposed augmentations are applicable over any baseline independent of the method. The RTM3D has two main data augmentation schemes already implemented which we consider as our baseline model : 1. Affine transformations, 2. image transforms. We do not include the stereo image pair from the KITTI dataset to extend the size of the dataset, or cited by [5] as stereo dataset augmentation. This produces a different baseline result in our case.

Fig. 3 shows the detection results of our proposed Box-Mixup data augmentation on KITTI in different scenarios. Ex. Occluded objects, missed detections and mis-classification by baseline. It shows baseline, Box-Mixup predictions on left panel and their corresponding BEV
E. Analysis of Results

From the comparison in Table III, the key data augmentation where observed improvement were the Box-Mixup and Box-CutPaste with an IoU check. When used without the IoU check we observe the drop in ICFW mAP scores denoting that the car AP score dominated the mAP scores. Ensuring that no two boxes from different images have a large intersection enables the data augmentation to provide a more diverse augmented images.

Effect of IoU: Box-MixUp augmentations generated with low IoU constraint between training images, provide better performance for pedestrian and cyclists classes.
Effect of Class imbalance: The data augmentations that performed well on the majority class are not optimal for the low frequency pedestrian and cyclist classes.

Table II shows the class-wise performance of all data augmentation methods. Global evaluation of all proposed data augmentations is shown in Table III. Almost all data augmentations have increased performance over "Car" and "Cyclists" classes. But, performance gain across data augmentation methods on "Pedestrian" class is lower. We hypothesize that this is due to the symmetry in width and height of pedestrians (which is not the case with cyclists where there are 3 distinct dimensions). This makes the network uncertain in localizing "Pedestrian" class.

V. CONCLUSION

We have evaluated a family of 2D data augmentations for monocular 3D object detection in images, along with a set of novel 2D data augmentations without changing the 3D geometry and avoids synthesizing new viewports. The data augmentations were applied over the RTM3D detector as baseline. In spite of these 2D data pixel/region level augmentations, improvements were observed in (mAP_{3D}) and (mAP_{bev}) scores. This was due to improved invariance to conditions like occlusion, corrupted pixels, novel orientations of object masks in 3D with varying background. As seen in Fig. 3 the Box-MixUp augmentation provides better separation of occluded vehicles, as well as a farther distance detection. To achieve a better performance across all classes we aim to study a better sampling mechanism in future work that takes into account the class imbalance inherent in the KITTI dataset.

REFERENCES

[1] H. Law and J. Deng, “Cornernet: Detecting objects as paired keypoints,” in The European Conference on Computer Vision (ECCV), September 2018.

[2] X. Zhou, J. Zhuo, and P. Krahenbuhl, “Bottom-up object detection by grouping extreme and center points,” in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2019.

[3] X. Zhou, D. Wang, and P. Krahenbuhl, “Objects as points,” ArXiv, 2019.

[4] Z. Liu, Z. Wu, and R. Töth, “Smoke: single-stage monocular 3d object detection via keypoint estimation,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2020, pp. 996–997.

[5] J.-H. Lee, M. Z. Zaheer, M. Astrid, and S.-I. Lee, “Smoothmix: a simple yet effective data augmentation to train robust classifiers,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2020, pp. 756–757.

[6] J. Redmon and A. Farhadi, “Yolov3: An incremental improvement,” ArXiv, 2018.

[7] D. Dwibedi, I. Misra, and M. Hebert, “Cut, paste and learn: Surprisingly easy synthesis for instance detection,” 2017 IEEE International Conference on Computer Vision (ICCV), pp. 1310–1319, 2017.

[8] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, and A. A. Kalinin, “Albumentations: Fast and flexible image augmentations,” Information, vol. 11, no. 2, 2020. [Online]. Available: https://www.mdpi.com/2078-2489/11/2/125

[9] X. Zhou, J. Zhuo, and P. Krahenbuhl, “Objects as points,” in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2019.

[10] B. Li, W. Ouyang, L. Sheng, X. Zeng, and X. Wang, “Gs3d: An efficient 3d object detection framework for autonomous driving,” in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2019.

[11] T. Devries and G. W. Taylor, “Improved regularization of convolutional neural networks with cutout,” ArXiv, vol. abs/1708.04552, 2017.

[12] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, “mixup: Beyond empirical risk minimization,” International Conference on Learning Representations, 2018.

[13] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, “Cutmix: Regularization strategy to train strong classifiers with localizable features,” in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 6023–6032.

[14] A. Geiger, P. Lenz, and R. Urtasun, “Are we ready for autonomous driving? the kitti vision benchmark suite,” in 2012 IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 3354–3361.

[15] X. Chen, K. Kundu, Z. Zhang, H. Ma, S. Fidler, and R. Urtasun, “Monocular 3d object detection for autonomous driving,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016.