AugPOD: Augmentation-oriented Probabilistic Object Detection

Chuan-Wei Wang*, Chin-An Cheng*, Ching-Ju Cheng*, Hou-Ning Hu, Hung-Kuo Chu, Min Sun
National Tsing Hua University
{kobe.wangddhl, chinancheng0811}@gmail.com, {nthuee4032@gapp, eborboihuc@gapp, hkchu@cs, sunmin@ee}.nthu.edu.tw

Abstract

The Probability Object Detection (POD) aims to measure spatial and label uncertainty of an object detector. The uncertainty measurement is important in robotic applications where actions triggered by erroneous but high-confidence perception can lead to catastrophic results. In this work, we propose the AugPOD, which augments the state-of-the-art models using several approaches, including i) MC Dropout ii) Gamma Correction iii) Virtual Dataset Collection. The experimental studies demonstrate that our method outperforms all the models involved in the competition with the score of 22.563*, which is* 2.72 *times improvement on the original Mask R-CNN.*

1. Introduction

Robotic Vision is an essential technology to equip robots with higher functionality. The surrounding information in various application fields, such as a factory or household, can provide robots the important clues to complete their tasks more accurately and efficiently. With the recent success of deep neural networks, lots of works applied Object Detection algorithm on Robotic Vision. Even though the Object Detection model could achieve high Average Precision (AP) on several datasets, such as MS COCO [7] and Pascal VOC [4], it may still encounter many failure cases in the real-world scenarios. Besides, robot vision systems also need to generalize well in various environments under different brightness and surrounding conditions.

The Probabilistic Object Detection Challenge [12] proposed a new metric and dataset which corresponded to previous problems. Compared with the standard AP-based measures, the Probability-based Detection Quality (PDQ) score [5] measures the spatial and label uncertainty. Besides, the dataset contains a wide variety of brightness and different surrounding conditions.

In this work, we present AugPOD that consists of 3 different techniques to address previously mentioned issues. Based on Probabilistic Object Detection, we apply MC Dropout [9] on different detection models, including Faster R-CNN [11], Mask R-CNN [6], Cascade Mask R-CNN [3] and Hybrid Task Cascade [3] to measure uncertainty of bounding box. For the generalization, we apply gamma correction, and data augmentation to deal with the large variation of brightness in day and night. Besides, we also collected our virtual dataset based on UnrealCV Engine [10] to increase the richness of surrounding conditions.

As a result, we demonstrate that the AugPOD achieves the score of 22.563, which is the top-1 rank in the competition and improves the original Mask R-CNN by the magnitude of 2.72.

*These authors contributed equally to this work

2. Methods

In this section, we described the method we used for the Robotic Vision Challenge[12]. It's divided into four subsections: Detection Model, Gamma Correction, Monte Carlo Dropout Estimation, and External Data Collection.

2.1. Detection Model

For the task of object detection, there are two types of frameworks, including the single-stage detectors and multi-stage detectors. Although the inference speed of multi-stage detectors is much slower than single-stage detectors, they often achieve higher accuracy than the others. Moreover, some works leverage segmentation information to improve the performance of object detection. According to previous observations, we chose the following four models. The first one is Faster R-CNN [11], the model regress the bounding boxes of objects by Region Proposal Network and ROI pooling. The second one is Mask R-CNN [6], they proposed ROI Align and the mechanism of estimating segmentation and bounding boxes simultaneously to get better performance. The third is Cascade Mask R-CNN [3] which is based on the idea of the work [1]. It cascades three modules after the original Mask R-CNN, including feature extraction, bounding box regression, and classification. The last one is the Hybrid Task Cascade [3] model, it interleaves bounding box regression and mask estimation, fusing additional semantic information to bounding box branch and mask branches. We made use of the above models in our experiment section.

Figure 1. Examples of gamma correction processing. The original images are listed in the first row and the images applying gamma correction are listed in the second row.

2.2. Gamma Correction

After inspecting the data set, we found the brightness of images is diverse, which causes object detection to fail. It's because half of the images in validation and testing sets are collected in the night time scene. In contrast, the images in MS COCO Dataset [7] are mostly captured in a bright condition. One way to solve this problem is augmenting randomized brightness data during training. However, it's still hard for a model to learn a large difference in the distribution of object appearance. The other way is to increase the brightness of the testing images. Gamma correction is a nonlinear operation to increase the value of each pixel, defined by the following formula.

$$V_{out} = AV_{in}^{\gamma} \tag{1}$$

where V_{in} and V_{out} are the input image and the corresponding output result. A and γ are tuning parameters. To prevent the over bright results after applying gamma correction, we used the following procedure. First, the input image is transformed from RGB format to HSV format, and then we average image pixel values in the V channel. Second, if the average of V is smaller than δ, the image will be enhanced by above formula 1 of gamma correction. We combined both of data augmentation and gamma correction procedure in our final result. Here we set $A = 1$, $\gamma = 0.4$ and $\delta = 60$. Figure. 1 displays the enhanced results after applying gamma correction procedure.

2.3. Monte Carlo Dropout Estimation

For Probabilistic Object Detection, it's crucial to estimate the probability distribution of predicted bounding boxes. In the work [5], they presented a bounding box by $B = (N_0, N_1) = (N(\mu_0, \Sigma_0)), N(\mu_1, \Sigma_1))$, where μ_i and Σ_i are the mean and covariances for the multivariate Gaussian describing the top-left and bottom-right corners of the box. However, most of the current popular Object Detection methods are for non-probabilistic, such as Faster R-CNN [11], Mask R-CNN [6], Cascade R-CNN [1] and Hybrid Task Cascade [3]. A straightforward way is applying fixed covariance estimation to the corners predicted by one of the previous methods. However, using fixed covariances is unrealistic for model generalization and cross-domain evaluation. Instead, Monte Carlo (MC) Dropout SSD [9] compared different merging strategies to measure the uncertainty of Object Detection model (SSD). Based on one of their clustering methods called Basic Sequential Algorithmic Scheme (BSAS), we modified it and employed the following algorithm 1. Here we set $K = 20$, $\alpha = 0.75$ and $\beta = 0.005$. Figure. 2 displays an example after applying the algorithm 1 on Mask R-CNN.

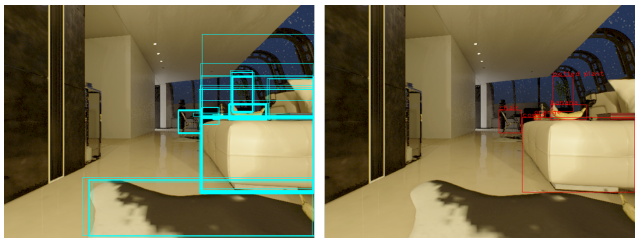

Figure 2. The left image is the output of MC Dropout on Mask R-CNN before clustering. The right image is the result of bounding boxes after applying the algorithm 1 on Mask R-CNN.

2.4. External Data Collection

Since this challenge[12] didn't provide any training data, and there is no restriction on external data usage, we collected an external virtual dataset based on UnrealCV Engine [10]. We build the indoor scene with the existing environment and 30 sub-classes of MS COCO [7] categories. Following the varied camera height settings in the dataset of this challenge[12], we collected our own training data with three different heights including tall, medium and short viewpoints. The examples of our external data are shown in Figure. 4. Our dataset includes $10K$ images with the ground truth of object detection and segmentation mask, which is available at: https://drive.google.com/drive/folders/13GBbYsEXu3SOAjVMv6UxK7_xxAvzSOuB.

3. Experiments

3.1. Dataset and Metric

Here we use the MS COCO Dataset [7], validation set provided by the organizer of CVPR 2019 The Probabilistic Object Detection Challenge [12], and the external dataset mentioned in the section. 2.4. The evaluation metric is following Probability-based Detection Quality (PDQ) score [5]. The PDQ score contains spatial quality and labels quality to accurately estimated spatial and label uncertainties.

3.2. Implementation Details

The Faster R-CNN [11] and Mask R-CNN [6] models are modified from this code [8], and the Cascade Mask R-

Algorithm 1 Framework of MC Dropout Object Detection

$K = Sampling\ times$
$I = Input\ image$
$C = Clusters\ of\ bounding\ boxes$
$\theta = The\ weight\ of\ detection\ bounding\ box\ head$
for each $k \in [0, K]$ **do**
  $\theta_k = Dropout(\theta, probabilty = 0.5)$;
  $D_k = Detection(I, \theta_k)$;
  for each $d_i \in D_k$ **do**
    for each $c_j \in C$ **do**
      if $Cls(d_i) = Cls(c_j)\ and\ IoU(d_i, c_j) \geq 0.5$ **then** $d_i\ joins\ c_j$;
      else $d_i\ becomes\ new\ cluster\ and\ joins\ C$;
      end if
    end for
  end for
end for
$Preds = []$
for each $c_i \in C$ **do**
  if $Number(c_i) \geq K \times \alpha$ **then**
    $\mu = Mean(c_i)$
    $cov = Covariance(c_i)$;
    $normcov = NormalizeCovariance(c_i)$;
    if $normcov < \beta$ **then**
      $Append\ \{(\mu, cov, Cls(c_i)\}\ into\ Preds$
    end if
  end if
end for
return $Preds$

CNN [3] and Hybrid Task Cascade [3] models are modified from this code [2]. We replaced the dimension of 80 categories with the new 30 ones on the last fully connected layer. The model was trained on single NVIDIA GeForce GTX 1080Ti with batch size of 2 and a learning rate of 0.001 which is decreased by 10 at the $140K$ iterations. And we adopted a weight decay of 0.0001 and momentum of 0.9. Besides, we joined MS COCO Dataset[7] and our own dataset, and applied data augmentation during the training process, including Gaussian noise and brightness. During inference, we set the confidence threshold at 0.3. If the predicted probability is greater than the confidence threshold, the output probability will be set to 1.0 for increasing label quality score.

3.3. Ablation Study

3.3.1 Detection Model

As shown in Table. 1, we compared four popular object detection models on validation set provided by the challenge organizer. All of them are pre-trained on MS COCO [7] without any fine-tuning and without covariance matrix. We found that the trend of the performance on the validation set is similar to the trend of the performance on MS COCO Dataset[7]. The state-of-the-art Hybrid Task Cascade [3] with Deformable Convolution Network as backbone still outperforms other models. But as mentioned in Hybrid Task Cascade [3] paper, they use 16 GPUs and train 20 epochs. Due to limited time and resource, we choose Mask R-CNN to be our final result.

3.3.2 Gamma Correction

To evaluate the effect of gamma correction procedure mentioned in previous section 2.2, we compared Cascade Mask R-CNN [3] and Hybrid Task Cascade [3] on the validation set without fine-tuning and using zero covariance matrix. The results are shown in Table. 2. After applying gamma correction procedure, there is significant performance gain on both of two models.

3.3.3 MC Dropout

We compared Mask R-CNN on the validation set by 3 different covariance matrix settings, including without covariance matrix, fixed covariance matrix and MC Dropout[9]. First, the fixed covariance matrix greatly improves the overall score and provides nearly twice the Average Spatial Quality compare to the model without it. Unfortunately, it is hard to manually search for the best covariance matrix. So, we applied MC Dropout[9] procedure mentioned in section 1. As a result, MC Dropout provides a significant performance gain without manually hyperparameter search. The results are shown in Table. 3. However, the main drawback of MC Dropout is that the computation time will increase dramatically, as sampling times increase.

3.4. Final Results

Table. 4 is the highest scores in our submissions. AugPOD is based on Mask R-CNN [6] architecture jointly trained $87,500$ steps on MS COCO Dataset[7] and our virtual dataset, then inference on the testing set with gamma correction procedure. The final score is 22.56. Due to limited time, our final results do not apply MC Dropout[9], but using fixed covariance matrix.

4. Conclusion

In summary, we presented AugPOD with MC Dropout, Gamma Correction and Virtual Dataset collection on the several object detection models in the Probabilistic Object Detection Challenge. Our AugPOD is successful to measure the uncertainty of object bounding box and generalize in a large variation of environment.

In order to teach robots to better estimate spatial and semantic uncertainty, our future work will focus on the object-level SLAM to establish 3D map and locating the position of objects for further manipulation tasks.

Model	Score	Avg. Overall Quality	Avg. Spatial Quality	Avg. Label Quality	COCO AP.
Faster R-CNN [11]	5.479	0.281	0.160	1.000	41.2%
Mask R-CNN [6]	6.056	0.268	0.148	1.000	42.2%
Cascade Mask R-CNN [3]	9.007	0.357	0.234	1.000	45.7%
Hybrid Task Cascade [3]	10.247	0.370	0.246	1.000	50.7%

Table 1. The validation results of different Detection Models which are pre-trained on MS COCO without fine-tuning and covariance matrix.

Model	Gamma	Score	Avg. Overall Quality	Avg. Spatial Quality	Avg. Label Quality
Cascade Mask R-CNN [3]	X	9.007	0.357	0.234	1.000
	V	10.698	0.380	0.255	1.000
Hybrid Task Cascade [3]	X	10.247	0.370	0.246	1.000
	V	10.918	0.390	0.263	1.000

Table 2. The validation results of applying gamma correction on Cascade Mask R-CNN and Hybrid Task Cascade which are pre-trained on MS COCO without fine-tuning and covariance matrix.

Model	Score	Avg. Overall Quality	Avg. Spatial Quality	Avg. Label Quality
Without Covariance Matrix	8.392	0.332	0.213	1.000
Fixed Covariance Matrix	14.787	0.545	0.399	1.000
MC Dropout	15.378	0.601	0.452	1.000

Table 3. The validation results of Mask R-CNN which are pre-trained on MS COCO and fine-tuned on validation set. Note: Fixed Covariance Matrix = [[1, 0], [0, 1]]

Model	G	ED	Score	Avg.OQ	Avg.SQ	Avg.LQ	TP	FP	FN
AugPOD	V	V	22.563	0.605	0.454	1.000	152967	113620	143400

Table 4. Our final results on the testing set. Our model applied Gamma correction procedure and trained on our virtual dataset. Due to limited time, instead of using MC Dropout, we used fixed Covariance Matrix = [[3, 0], [0, 3]]. Legend:X=Not implement, V=Implement, G=Gamma correction, ED=External data, Avg.OQ=Avg. Overall Quality, Avg.SQ=Avg. Spatial Quality, Avg.LQ=Avg. Label Quality, TP=True Positives, FP=False Positives, FN=False Negatives

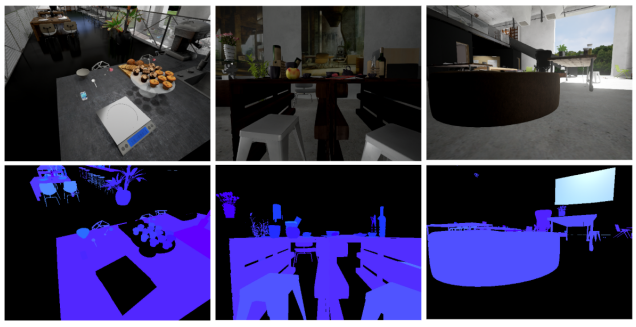

Figure 3. The examples of our external dataset. The images are in the first row and the corresponding ground truth segmentation masks are in the second row.

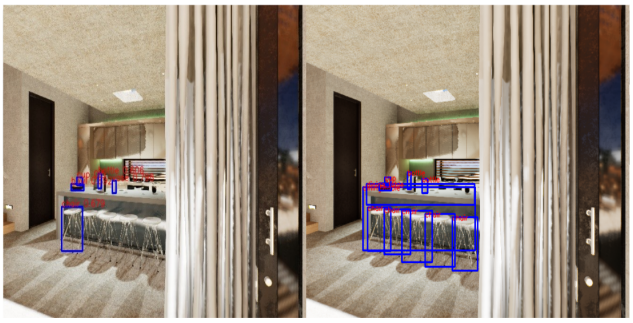

Figure 4. The left image is the output bounding box of original Mask R-CNN model. The right image is output bounding box of our AugPOD model.

References

[1] Z. Cai and N. Vasconcelos. Cascade r-cnn: Delving into high quality object detection. *CVPR*, 2018. 1, 2

[2] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. mmdetection. https://github.com/open-mmlab/mmdetection, 2018. 3

[3] K. Chen, J. Pang, J. Wang, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Shi, W. Ouyang, C. C. Loy, and D. Lin. Hybrid task cascade for instance segmentation. *CVPR*, 2019. 1, 2, 3, 4

[4] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 2010. 1

[5] D. Hall, F. Dayoub, J. Skinner, H. Zhang, D. Miller, P. Corke, G. Carneiro, A. Angelova, and N. Snderhauf. Probabilistic object detection: Definition and evaluation. *arxiv*, 2018. 1, 2

[6] K. He, G. Gkioxari, P. Dollr, and R. Girshick. Mask r-cnn. *CVPR*, 2018. 1, 2, 3, 4

[7] T. Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollr. Microsoft coco: Common objects in context. *ECCV*, 2014. 1, 2, 3

[8] Massa, Francisco, Girshick, and Ross. maskrcnn-benchmark: Fast, modular reference implementation of Instance Segmentation and Object Detection algorithms in PyTorch. https://github.com/facebookresearch/maskrcnn-benchmark, 2018. Accessed: [Insert date here]. 2

[9] D. Miller, F. Dayoub, M. Milford, and N. Sunderhauf. Evaluating merging strategies for sampling-based uncertainty techniques in object detection. *ICRA*, 2019. 1, 2, 3

[10] W. Qiu, F. Zhong, Y. Zhang, S. Qiao, Z. Xiao, T. S. Kim, Y. Wang, and A. Yuille. Unrealcv: Virtual worlds for computer vision. *ACM Multimedia Open Source Software Competition*, 2017. 1, 2

[11] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *NIPS*, 2015. 1, 2, 4

[12] J. Skinner, D. Hall, H. Zhang, F. Dayoub, and N. Snderhauf. The probabilistic object detection challenge. *arxiv*, 2019. 1, 2