B-H-Deformable-DETR: H-Deformable-DETR model based on Bayesian neural network optimization in Few-Shot Object Detection

Yijun Chen*[a], Shenglin Zeng[a], Muyang Li[a], Yizhe Guo[a], Fayang Zhao[a]
[a]School of Computer and Artificial Intelligence, Zhengzhou University, Zhengzhou, China
*Corresponding author's e-mail: ie_yijunchen@stu.zzu.edu.cn

ABSTRACT

In this paper, we propose an improved B-H-Deformable-DETR (Bayesian H-Deformable-DETR) model for the problem of insufficient accuracy and generalization ability in the task of Few-Shot Object Detection. In our study, we adopt Bayesian MLP (Multi-Layer Perceptron based on Bayesian Linear Layers) as the prediction layer of bounding box regression. By introducing prior distribution and uncertainty estimation, we transform the neural network with deterministic parameters into a probabilistic neural network with stochastic properties. This probabilistic neural network can handle the variability and sparsity of data more effectively with the expectation of improving the model's performance in dealing with uncertainty and generalization. As a result, under the setting of using ResNet50 and Swin-Transformer as backbone and training on the first 5000 images of the COCO dataset, the AP (average precision) of our model improves by +5.4 AP and +4.6 AP compared to the original H-Deformable-DETR model. These results indicate that our model has significantly improved prediction accuracy and enhanced generalization ability on small sample datasets.

Keywords: Few-Shot Object Detection, B-H-Deformable-DETR, Bayesian MLP, Average Precision (AP), Generalization Capability

1. INTRODUCTION

Object detection is a fundamental task in computer vision aimed at predicting the bounding boxes and categories of objects in a single image. Traditional convolutional neural networks have dominated the field of object detection for many years. However, Transformer-based object detectors (e.g., DETR (DEtection TRansformer) and its variants) have received much attention for their end-to-end characteristics and excellent performance compared to traditional object detectors. These detectors extract features from the Transformer's encoder and avoid the tedious operation of traditional NMS (non-maximal suppression) through one-to-one ensemble matching, greatly simplifying the detection process. However, in the task of Few-Shot Object Detection, such detectors are often unable to efficiently train robust models due to the sparsity and variability of the data, which leads to degraded detection accuracy and insufficient generalization ability. In addition, the original one-to-one matching method has only a small number of queries assigned as positive samples during the training process, which leads to low efficiency of positive sample training, especially on small sample datasets.

H-Deformable-DETR (Deformable DETR with hybrid matching) [1] is an improved version of DETR. It introduces a hybrid matching strategy that integrates the traditional one-to-one matching strategy with an auxiliary one-to-many matching strategy. By adding an auxiliary one-to-many matching branch during training, it addresses the issue of low positive sample training efficiency in DETR, increasing the utilization of positive samples and improving detection accuracy. However, it still has deficiencies in coping with data variability and sparsity. The generalization capability of the model and its ability to handle uncertainty needs to be further improved as well. We mainly focus on model performance on small sample datasets. However, improving the detection accuracy and enhancing the generalization capability of the model on small sample datasets remains a challenge, as we describe below:

(1) In small sample datasets, due to sparse data and high variability, it is difficult for existing models to be efficiently trained with robustness, resulting in decreased detection accuracy and insufficient generalization ability.

(2) Existing models show certain limitations in coping with data variability and sparsity, which limits their generalization ability to some extent. In practical applications, the diversity and complexity of data increase the uncertainty of model predictions, such as data noise, incompleteness, and unpredictable environmental changes,

which may lead to unstable and inaccurate prediction results if the model is unable to deal with such uncertainty effectively.

(3) There have been many types of research for Few-Shot Object Detection tasks, such as methods based on data augmentation, transfer learning, and few-shot learning. These methods have shown some effectiveness, but they often require complex preprocessing or additional data, which may limit their practical applications.

To overcome the above challenges, we have introduced Bayesian methods into the H-Deformable-DETR model by integrating uncertainty estimation and prior distributions in the expectation of improving the model's performance in dealing with uncertainty and generalization. We summarize our contributions as follows:

(1) We introduce the method of prior distribution and uncertainty estimation to transform the neural network with deterministic parameters into a probabilistic neural network with stochastic properties, which can deal with the variability and sparsity of data more effectively.

(2) We added KL divergence (Kullback–Leibler divergence) to the loss function of the model to measure the difference between the predictive distribution and the prior distribution to better regularize the model, prevent overfitting, and improve generalization.

(3) During the training process of the improved model, we use a transfer learning approach to further enhance the performance of the model on small sample datasets.

In this study, we use ResNet-50 and Swin-Transformer as backbone and train on the first 5000 images of the COCO dataset, respectively, followed by experimental evaluation of the original H-Deformable-DETR model and our model. The experimental results show that our model has significant improvement in terms of AP on small sample datasets. Specifically, the improved model trained on the first 5000 images of the COCO dataset achieves AP values of 44.6 and 46.6. Compared to the original model, these results show an increase of +5.4AP and +4.6AP, respectively.

2. RELATED WORK

2.1 Classical CNN for Object Detection

Convolutional neural networks (CNNs) have made substantial contributions to the advancement of object detection. Since the introduction of R-CNN (Region-CNN) [2], a variety of CNN-based object detection frameworks have been developed, including Fast R-CNN [3], Faster R-CNN [4], YOLO (You Only Look Once) [5], and SSD (Single Shot MultiBox Detector) [6]. These models have improved both detection speed and accuracy through various strategies. For instance, Faster R-CNN incorporates a Region Proposal Network (RPN) that significantly enhances detection efficiency, while models like YOLO and SSD transform the object detection process into a regression task, enabling real-time detection. However, these CNN-based methods still have limitations in handling small sample datasets and dealing with uncertainty. Their generalization ability needs to be improved as well.

2.2 DETR for Object Detection

In recent years, Transformer-based architectures have achieved remarkable advancements in computer vision. DETR [7] stands as the first model to apply the Transformer framework to object detection tasks. DETR achieves end-to-end object detection by transforming the object detection problem into a set prediction problem and combining it with the Self-Attention Mechanism. This approach can avoid the region proposal step in traditional methods as well. Compared with traditional CNN methods, DETR performs well in dealing with long-range dependencies and multi-scale feature fusion. However, DETR requires larger computational resources and longer convergence time during training and has insufficient generalization ability on small sample datasets. H-Deformable-DETR [1,7], as an improved version of DETR, further enhances the performance of DETR through the introduction of a hybrid matching mechanism and a multi-scale deformable Attention Mechanism. However, when the data sample size is small, the generalization ability and prediction accuracy of the H-Deformable-DETR model can still be further improved.

2.3 Bayesian Methods

Bayesian methods provide a probabilistic framework for modeling uncertainty in machine learning, which is particularly effective in handling tasks with limited data and uncertain conditions. In the context of object detection, Bayesian methods

provide a way to quantify and manage the uncertainty inherent in predictions, presenting a notable improvement over deterministic models.

Bayesian neural networks (BNN) [8] introduce uncertainty by distributing the network weights rather than using fixed values. This allows BNNs to provide not only point estimates but also uncertainty intervals for their predictions. In the case of small or noisy datasets, this property is particularly useful as models can express confidence in their prediction. BNNs have been used in object detection tasks to improve detection performance by increasing robustness and providing reliable confidence scores.

Recent advances in Bayesian deep learning have also introduced hybrid models that combine Bayesian principles with modern deep learning architectures. For example, Bayesian versions based on popular object detection models such as Faster R-CNN [9] and YOLO [10] have been developed to take advantage of the strengths of both approaches. These models aim to improve detection accuracy and reliability, especially in applications with scarce data and high uncertainty.

Despite these advantages of Bayesian methods in object detection, they still face some challenges such as computational complexity and scalability issues. Training Bayesian models usually requires more computational resources and time than their deterministic counterparts. In addition, implementing and adapting Bayesian methods can be complex, which may limit their practical application.

3. B-H-DEFORMABLE-DETR

Aiming at the problem of insufficient accuracy and generalization ability in the task of Few-Shot Object Detection, we propose an improved B-H-Deformable-DETR model. This model is centered on the introduction of a multi-layer perceptron based on Bayesian linear layers as the prediction layer of bounding box regression. It transforms the neural network with deterministic parameters into a probabilistic neural network with stochastic properties. This change enhances the model's ability to handle data variability and sparsity, improving its performance in dealing with uncertainty and generalization. Our model consists of two modules: Two-stage Deformable-DETR and Bayesian MLP. During training, we use a hybrid branching scheme as a hybrid matching strategy and train the model through transfer learning, as shown in Figure 1 and Figure 2.

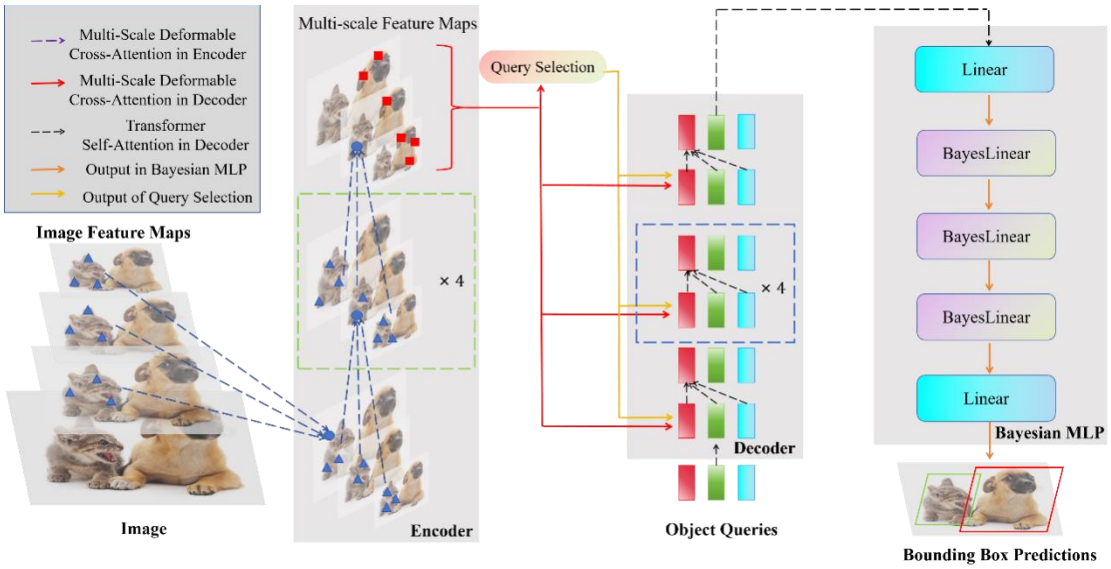


Figure 1. Illustration of our B-H-Deformable-DETR.

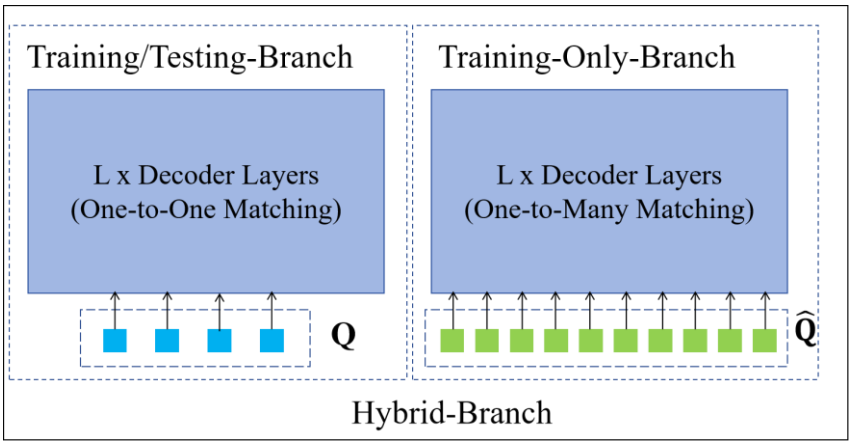


Figure 2. Illustration of our Hybrid Branch Scheme.

3.1 Two-stage Deformable-DETR

Two-stage Deformable-DETR [1,11] is an improved DETR model designed to improve the accuracy and efficiency of object detection. It consists of two main phases. In the first phase, the model generates region proposals, and in the second phase, it refines and classifies these proposals. In the two-stage Deformable-DETR, during the first stage, the model generates preliminary bounding boxes through a rough predictor that cover locations that may contain targets. During the second stage, it uses these preliminary bounding boxes as a reference for further optimization through a more refined predictor to improve the accuracy of the bounding boxes and the classification accuracy. This approach combines the advantages of traditional region proposal-based methods (e.g., Faster R-CNN) with those of DETR. As a result, our model can significantly improve detection accuracy while ensuring efficient training and inference at the same time.

3.2 Bayesian MLP Module

In the B-H-Deformable-DETR model, we introduce Bayesian MLP (Multi-Layer Perceptron based on Bayesian Linear Layers) as the predictive layer used for bounding box regression. This enhancement aims to improve the model's performance in handling uncertainty and generalization. Traditional Feed-Forward Neural Networks are often prone to overfitting and poor generalization performance when dealing with small sample datasets. Bayesian MLP effectively overcomes these problems by introducing uncertainty estimation and prior information.

Structure and Definition. The architecture of Bayesian MLP includes an input layer, multiple Bayesian linear layers, and an output layer. Given the input feature's dimension as d_{in}, the hidden layer's dimension as d_{hidden}, the output's dimension as d_{out}, as well as the number of hidden layers as L, the structure of Bayesian MLP can be expressed as:

$$\mathbf{y} = F_L\left(F_{L-1}(\dots F_0(\mathbf{x})\dots)\right), \qquad (F_i(\mathrm{x}) = \mathbf{W_i}\cdot\sigma(\mathrm{x}) + \mathbf{b_i}) \tag{1}$$

where $\sigma(\cdot)$ denotes the ReLU activation function. W_i and b_i are denoted as the weights and biases of layer i, respectively. These parameters are all modeled as Gaussian distributions, which enables the modeling of uncertainty in the model parameters. Specifically, the prior distributions of the weights and biases are set as:

$$W_i \sim \mathcal{N}\left(\mu_{W_i}, {\sigma^2}_{W_i}\right), b_i \sim \mathcal{N}\left(\mu_{b_i}, {\sigma^2}_{b_i}\right), \tag{2}$$

The Bayesian MLP infers by integrating prior knowledge with observational data evidence. And it employs probability to quantify the uncertainty in reasoning. The outcome is a probability distribution that expresses the model's confidence in the likelihood of various predictions.

Bayesian Learning Process. The Bayesian learning process initiates by establishing a model M and assigning a prior distribution $p(h)$ over the model's parameters h. The prior distribution reflects our initial confidence regarding the values of the parameters before observing any data. Upon observing fresh data $\mathrm{D} = \{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots (x^{(n)}, y^{(n)})\}$, the prior distribution is adjusted to form a posterior distribution that uses Bayes' rule:

$$p(h|D) = \frac{p(D|h)p(h)}{p(D)} \; \alpha \; L(h|D)p(h), \tag{3}$$

where the likelihood function $L(h|D)$ represents the probability of the observed data as a function of the model's unknown parameters. To predict the new output $y^{(n+1)}$ for the new input $x^{(n+1)}$, the predictive distribution is derived by integrating the posterior distribution over the model parameters:

$$p\left(y^{(n+1)}\middle|x^{(n+1)}, D\right) = \int p\left(y^{(n+1)}\middle|x^{(n+1)}, h\right)p(h|D)dh\,, \tag{4}$$

This is equivalent to taking the average prediction of all models and weighting them by their merits.

Regularization. We use KL divergence to measure its difference from the prior distribution, thus introducing a regularization term in the training process. The KL divergence for each layer is calculated as:

$$\mathrm{KL}(\boldsymbol{W}_i) = \frac{1}{2}\left(\frac{\sigma^2_{\boldsymbol{W}_i}}{\sigma^2_{prior}} + \frac{\left(\mu_{\boldsymbol{W}_i} - \mu_{prior}\right)^2}{\sigma^2_{prior}} - 1 + log\,\frac{\sigma^2_{prior}}{\sigma^2_{\boldsymbol{W}_i}}\right), \tag{5}$$

The total KL loss is the sum of the KL divergence of each layer:

$$\mathrm{KL}_{\mathrm{total}} = \sum_{i=0}^{L}(\mathrm{KL}(\mathbf{W}_i) + \mathrm{KL}(\mathbf{b}_i)), \tag{6}$$

The final loss function consists of the bounding box regression loss and the KL loss together:

$$L = L_{regression} + \beta \cdot \mathrm{KL}_{total}, \tag{7}$$

where β is a trade-off coefficient to balance the regression loss and the KL loss.

In summary, by introducing Bayesian linear layers, we can model the uncertainty of the model parameters. This uncertainty modeling enables our model to better cope with the variability and sparsity of the data when dealing with small sample datasets, thus improving the generalization ability of our model. At the same time, we introduce the KL divergence regularization term so that we can effectively control the complexity of our model to prevent overfitting. This makes our model perform more robustly when dealing with new data. Compared to traditional Feed-Forward Neural Networks, this Bayesian MLP not only has stronger explanatory power in theory but also shows better performance in practical applications.

4. EXPERIMENT

4.1 Experimental Setup

Dataset

In this experiment, we use the COCO2017 [12] dataset for training and evaluation. COCO2017 is a large-scale image recognition, segmentation, and object detection dataset consisting of 118,000 training images and 5,000 validation images. These images cover 80 categories of common objects, such as people, animals, transportation, etc. Due to its extensive annotations and diverse scenarios, the COCO2017 dataset has become a widely used benchmark for object detection.

For our experiments, we selected the first 5000 images from the COCO2017 training set to simulate a Few-Shot Object Detection scenario. This approach reduces computational costs and enables faster iteration, which is crucial for efficiently testing and refining improvements in our B-H-Deformable-DETR model. Similar strategies have been employed in previous studies to evaluate model performance in Few-Shot Object Detection [13].

Evaluation Metrics

Average Precision (AP): Average precision is the most commonly evaluated metric in object detection tasks. AP calculates the average precision of the model over all categories for different IoU (Intersection over Union) thresholds. Commonly used AP metrics include AP50 (IoU=0.50), AP75 (IoU=0.75), and AP@[0.50:0.95], which is the average of AP over IoUs ranging from 0.50 to 0.95 at 0.05 intervals.

4.2 Baseline Methods

Baseline Models

H-Deformable-DETR. H-Deformable-DETR is based on the two-stage Deformable-DETR model and uses a hybrid matching strategy. Two-stage Deformable-DETR is an improved DETR model designed to improve the accuracy and efficiency of object detection. It consists of two main phases. In the first phase, the model generates region proposals, and in the second phase, it refines and categorizes these proposals. The hybrid matching strategy, on the other hand, adds an additional one-to-many matching branch to the two-stage Deformable-DETR model. During the training process, the model calculates the corresponding loss according to the matching results and optimizes the parameters of the one-to-one and one-to-many matching branches at the same time.

Implementation Details

We used ResNet-50 and Swin-Transformer as the backbone respectively to train the model by transfer learning, and we froze the parameters of the Transformer module (excluding the prediction and regression layers) for the first 5 epochs of training to stabilize the feature extraction and prevent the pre-training weights from being corrupted, and then thawed these parameters for the co-training to gradually adjust the pre-trained features and optimize the whole model. With this staged training strategy, the integrity of the pre-trained features can be protected at the initial stage, and the potential of the pre-trained network can be fully utilized at the later stage to achieve the joint optimization of feature extraction and object detection. In the bounding box regression part of the model, Bayesian MLP is introduced, which consists of two linear layers plus three Bayesian linear layers, to better handle uncertainty through the Bayesian linear layers, thus improving the generalization ability of the model and its performance on small datasets. And we employ a two-stage mechanism to enhance the accuracy of bounding boxes and the precision of classification. In addition, we perform both one-to-one and one-to-many matching during training to enhance the model's learning ability. We adopt the AdamW optimizer with a learning rate of 2e-4 and a weight decay of 1e-4 to train our model. Training and evaluation are conducted on the COCO dataset.

4.3 Experimental Results

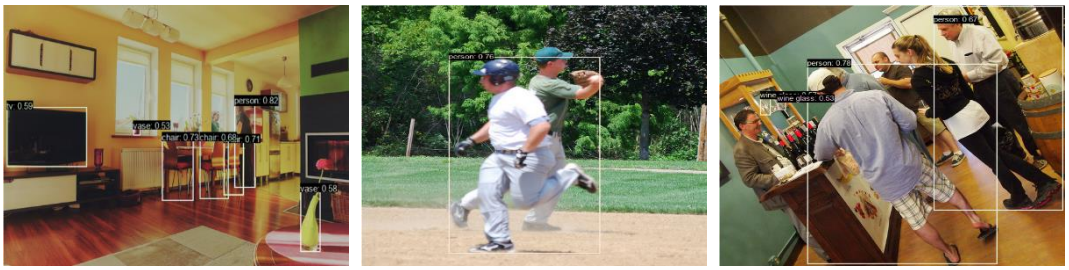
Object detection results on COCO. Table 1 presents a comparison of object detection results on the COCO dataset. Our B-H-Deformable-DETR model consistently outperforms the baseline across various backbones, such as ResNet-50 and Swin-Tiny. These results were achieved by training on the first 5000 images from the COCO dataset over 12 epochs. Specifically, when Swin-Tiny is utilized as the backbone and trained for 12 epochs, our B-H-Deformable-DETR model enhances the original model's performance from 42.0% to 46.6%.

Table1. Object detection results on COCO.

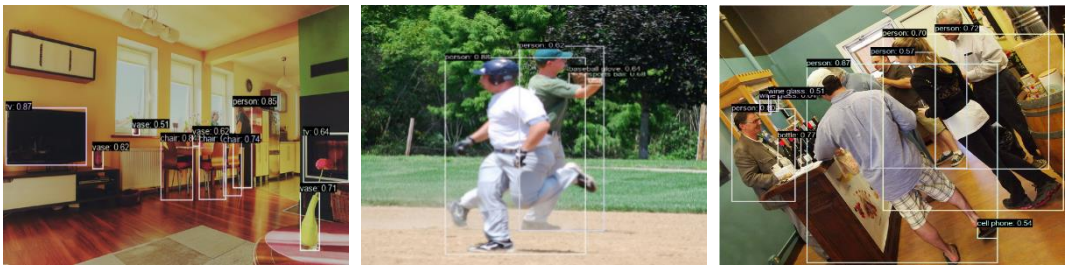
method	dataset	backbone	#epochs	AP	AP_S	AP_M	AP_L
H-Deformable-DETR	COCO (First 5000 Samples)	ResNet-50	12	39.2	22.5	42.2	51.9
H-Deformable-DETR[1]	COCO (Full Dataset)	ResNet-50	12	48.7	31.2	51.5	63.5
B-H-Deformable-DETR	COCO (First 5000 Samples)	ResNet-50	12	$\mathbf{44.6}^{+5.4}$	28.1	47.5	58.1
H-Deformable-DETR	COCO (First 5000 Samples)	Swin-Tiny	12	42.0	24.8	45.0	55.7
H-Deformable-DETR[1]	COCO(Full Dataset)	Swin-Tiny	12	50.6	33.4	53.7	65.9
B-H-Deformable-DETR	COCO (First 5000 Samples)	Swin-Tiny	12	$\mathbf{46.6}^{+4.6}$	29.5	49.7	60.6

Visualization of object detection results. Figure 3 depicts the prediction results on the COCO object detection set. It can be seen that our B-H-Deformable-DETR model can accurately identify more objects than the baseline model. Additionally,

it achieves higher confidence levels when trained with different backbones (including ResNet-50 and Swin-Tiny). Especially, it performs better when dealing with scenes with multiple target objects. These results were achieved by training on the first 5000 images of the COCO dataset over 12 epochs. These results show that our improved model can significantly enhance the detection accuracy and generalization capability of the model.

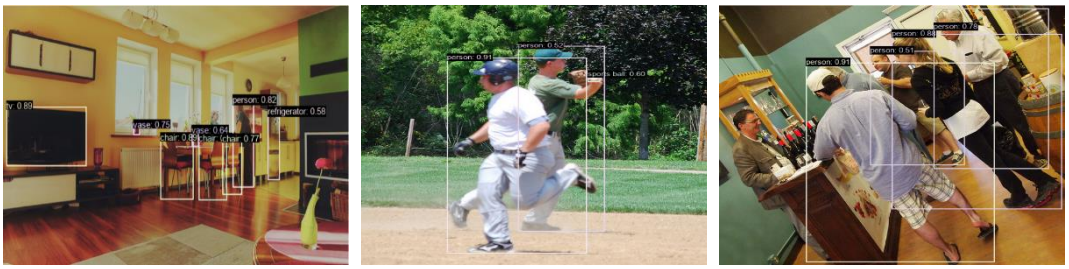


(a) H-Deformable-DETR with Swin-Tiny



(b) B-H-Deformable-DETR with Swin-Tiny

(c) H-Deformable-DETR with ResNet-50



(d) B-H-Deformable-DETR with ResNet-50

Figure 3. Visualization of prediction results on COCO. Each subgraph contains the original image as well as the object bounding boxes predicted by different models. For readability, these bounding boxes are given different colors and transparency to distinguish different object classes, and the confidence value of the model's recognition of the object is also labeled inside each bounding box. By observing these subgraphs, we can visualize the difference in performance between the original H-Deformable-DETR model and our B-H-Deformable-DETR model in real-world application scenarios.

4.4 Ablation Study

Effectiveness of each component. Table 2 shows the results of the ablation experiments for the B-H-Deformable-DETR model, comparing the effect of using or not using a linear layer before and after the Bayesian linear layer in the case of using the Swin-Transformer as a backbone. The results show that the full configuration (i.e., using a linear layer both before and after the Bayesian linear layer) obtained the highest AP of 46.6, while removing the subsequent linear layer and using only the Bayesian linear layer decreased the AP to 30.1 and 24.3, respectively. It can be seen that using a linear layer before and after the Bayesian linear layer is crucial for improving the performance of the model.

Table2. Ablation results for B-H-Deformable-DETR.

Linear layer before Bayesian linear layers	Bayesian linear layers	Linear layer after Bayesian linear layers	AP
√	√	√	46.6
√	√		30.1
	√		24.3

5. CONCLUSION

In this paper, we propose an improved B-H-Deformable-DETR object detection model to enhance the model's prediction accuracy and generalization ability on small datasets. The experimental results show that under the setting of using ResNet50 and Swin-Transformer as backbone and training on the first 5000 images of the COCO dataset, the AP of our model improves by +5.4 AP and +4.6 AP compared to the original H-Deformable-DETR model. Furthermore, Bayesian networks also have their unique advantages in dealing with complex data scenarios. In our future work, we will continue to optimize the model structure and algorithms for a wider range of application scenarios, including complex data conditions such as noisy images, environmental variations, occlusions, and weakly supervised or unsupervised learning situations. We hope that our efforts will promote the further development of object detection techniques in uncertainty and data scarcity conditions, improving their performance and reliability in various real-world application scenarios.

REFERENCES

[1] Jia, D., Yuan, Y., He, H., Wu, X., Yu, H., Lin, W., Sun, L., Zhang, C. and Hu, H., "DETRs with Hybrid Matching," arXiv.org, 2022, <https://arxiv.org/abs/2207.13080> (14 September 2024).

[2] Girshick, R., Donahue, J., Darrell, T. and Malik, J., "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," 2014 IEEE Conference on Computer Vision and Pattern Recognition, 580–587 (2014). Van Derlofske, J. F., "Computer modeling of LED light pipe systems for uniform display illumination," Proc. SPIE 4445, 119-129 (2001).

[3] Girshick, R., "Fast R-CNN," 2015 IEEE International Conference on Computer Vision (ICCV), 1440–1448 (2015).

[4] Ren, S., He, K., Girshick, R. and Sun, J., "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," IEEE Transactions on Pattern Analysis and Machine Intelligence **39**(6), 1137–1149 (2017).

[5] Redmon, J., Divvala, S., Girshick, R. and Farhadi, A., "You Only Look Once: Unified, Real-Time Object Detection," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 779–788 (2016).

[6] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y. and Berg, A. C., "SSD: Single Shot MultiBox Detector," Computer Vision – ECCV 2016 **9905**, 21–37 (2016).

[7] Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A. and Zagoruyko, S., "End-to-End Object Detection with Transformers," Computer Vision – ECCV 2020 **12346**, 213–229 (2020).

[8] Blundell, C., Cornebise, J., Kavukcuoglu, K. and Wierstra, D., “Weight Uncertainty in Neural Networks,” arXiv.org, 2015, <https://arxiv.org/abs/1505.05424>.
[9] Sagar, Tanveer, J., Chen, Y., Jun Hoong Chan, Hyung Seok Kim, Karam Dad Kallu and Ahmed, S., “Bayes R-CNN: An Uncertainty-Aware Bayesian Approach to Object Detection in Remote Sensing Imagery for Enhanced Scene Interpretation,” Remote Sensing **16**(13), 2405–2405 (2024).
[10] Das, D. and Miura, J., “Camera Motion Compensation and Person Detection in Construction Site Using Yolo-Bayes Model,” 2022 26th International Conference on Pattern Recognition (ICPR) (2022).
[11] Zhu, X., Su, W., Lu, L., Li, B., Wang, X. and Dai, J., “Deformable DETR: Deformable Transformers for End-to-End Object Detection,” arXiv:2010.04159 [cs] (2021).
[12] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P. and Zitnick, C. L., “Microsoft COCO: Common Objects in Context,” Computer Vision – ECCV 2014 **8693**, 740–755 (2014).
[13] Xie, G., Wang, J., Liu, J., Jin, Y. and Zheng, F., “Pushing the Limits of Fewshot Anomaly Detection in Industry Vision: Graphcore,” Iclr.cc, 2023, <https://iclr.cc/virtual/2023/poster/12228>.