VotingNet: Adaptive Hough Voting Based Compositional Model for X-Ray Prohibited Item Detection Under Occlusion

Kaitao Huang, Yan Yan*
Fujian Key Laboratory of Sensing and Computing for Smart City, School of Informatics, Xiamen University, Xiamen, China

Abstract

Recently, X-ray prohibited item detection has been widely used for security inspection. In practical applications, the items in the luggage are severely overlapped, leading to the problem of occlusion. In this paper, we address prohibited item detection under occlusion from the perspective of the compositional model. To this end, we propose a novel VotingNet for occluded prohibited item detection. VotingNet incorporates an Adaptive Hough Voting Module (AHVM) based on the generalized Hough transform into the widely-used detector. AHVM consists of an Attention Block (AB) and a Voting Block (VB). AB divides the voting area into multiple regions and leverages an extended Convolutional Block Attention Module (CBAM) to learn adaptive weights for inter-region features and intra-region features. In this way, the information from unoccluded areas of the prohibited items is fully exploited. VB collects votes from the feature maps of different regions given by AB. To improve the performance in the presence of occlusion, we combine AHVM with the original convolutional branches, taking full advantage of the robustness of the compositional model and the powerful representation capability of convolution. Experimental results on OPIXray and PIDray datasets show the superiority of VotingNet on widely used detectors (including representative anchor-based and anchor-free detectors).

Introduction

With the population growth in cities and increasing crowd density at public transportation hubs, security inspection has become very important in protecting public safety. X-ray scanners, which can generate X-ray images to determine the existence of prohibited items in passengers' luggage, are widely used for security inspection. In many cases, the items in the luggage are randomly stacked. As a result, it is challenging for security personnel to accurately identify all prohibited items after prolonged observation of complex X-ray images. Although frequent shift changes can somewhat alleviate this problem, they consume significant manpower, which is undesirable in real-world applications.

Fortunately, with recent advances in deep learning, automatic X-ray prohibited item detection has become possible. However, different from traditional object detection tasks, X-ray prohibited item detection often suffers from heavy occlusion, since the items in the luggage are severely overlapped. Therefore, how to effectively detect prohibited items under occlusion needs further study.

*Corresponding Author

To address the occlusion, a common solution is to use a data-driven strategy, which trains a model based on a large-scale dataset containing diverse samples captured under different occlusion conditions. In this way, the model can learn occlusion-invariant features from these samples. Nevertheless, collecting the dataset covering diverse occlusion patterns in quantity, appearance, and position is not trivial.

Recently, compositional models (such as [1, 2, 3]), which represent a whole object by the spatial composition of parts, have shown great robustness for natural object detection under occlusion. These models often adopt two stages to identify objects (i.e., the parts are first detected, and then the spatial relationship between the detected parts is modeled to detect objects by aggregating the information from different parts). Both stages are designed to achieve robustness to occlusion. Thus, these models can detect the object under occlusion as long as visible parts satisfy reasonable spatial constraints.

The prohibited items in X-ray images significantly differ from objects in natural images. X-ray possesses remarkable penetrating power, where different materials within an object exhibit varying degrees of absorption, showing distinct colors in X-ray images. Moreover, the contours of prohibited items and safety items are usually mixed. In this way, conventional compositional models are difficult to directly apply for prohibited item detection because these models are required to not only detect parts but also learn the spatial constraints among parts.

To address X-ray prohibited item detection under occlusion, in this paper, we develop VotingNet by adaptively collecting the votes from different pre-defined regions in a log-polar vote field [4]. In this way, the spatial constraints between parts can be explicitly modeled, where part detection can be performed in each region. As a result, the learning capability of the model to detect occluded prohibited items is greatly improved, benefiting the final detection performance.

Specifically, VotingNet performs prohibited item detection by designing an Adaptive Hough Voting Module (AHVM) based on the widely-used detector. AHVM consists of two blocks: an Attention Block (AB) and a Voting Block (VB). On the one hand, AB first divides the voting area into multiple regions, corresponding to different parts of a prohibited item. Then, it leverages an extended Convolutional Block Attention Module (extended CBAM) to obtain adaptive weights for inter-region features and intra-region features. On the other hand, based on the generalized Hough transform [5], VB takes the weighted feature maps obtained from AB as the input and performs voting. Unlike existing methods (which directly perform detection by using the output of compositional models), we combine the output of the compositional model with the original convolutional structure of

the detection head in a parallel way. Such a way combines the advantages of the convolution neural network and compositional model for improving the robustness against occlusion, allowing us to achieve a model with strong discriminative capability and robustness to occlusion.

In summary, the contributions of this paper are as follows:

- We propose VotingNet, which involves a novel AHVM, for X-ray prohibited item detection under occlusion. To the best of our knowledge, we are the first to introduce generalized Hough transform to the compositional model for handling occlusion in X-ray images.
- We develop an extended CBAM to adaptively learn weights for both inter-region and inter-region features, explicitly considering the influence of different regions for voting.
- We validate the effectiveness of VotingNet on two types of object detectors (including anchor-based and anchor-free object detectors). Experiments on the OPIXray [6] and PIDray [7] datasets show the superiority of our proposed method for addressing occlusion in X-ray prohibited item detection.

Related Work

Object Detection. Existing object detection methods can be divided into two categories: anchor-based and anchor-free methods. Anchor-based methods include two-stage and one-stage methods. The representative two-stage methods are RCNN series, including Faster-RCNN [8], Cascade R-CNN [9], etc. The representative one-stage methods are SSD [10], RetinaNet [11], etc. Anchor-free methods include dense prediction-based methods (such as YOLOv1 [12], DenseBox [13], FCOS [14]) and keypoint-based methods (such as CornerNet [15], ExtremeNet [16], OAP [17], CenterNet [18]).

Object Detection under Occlusion. Yan et al. [19] propose a boosted cascade framework to detect partially visible objects. Recently, several deep learning methods [20, 21] have been proposed for detecting occluded objects. Note that these methods require detailed part-level annotations to reconstruct the occluded objects. Xiang et al. [22] propose to use 3D models and formulate object detection under occlusion as a multi-label classification task. However, in the X-ray prohibited item detection, the classes of occluders are hardly modeled in 3D and are often not known as a priori. The most related methods to our work are part-based voting methods, which have been proven to work reliably for object detection under occlusion. However, some methods [2, 23] adopt a fixed-size bounding box, limiting their applicability to real-world object detection. Wang et al. [1] develop a method to robustly estimate the bounding box of the object even under very strong partial occlusion. But such a method requires pre-training. In this paper, we propose VotingNet, which introduces AHVM as the compositional model. VotingNet collects the votes from different pre-defined regions, then the occluded prohibited items can be detected based on votes from unoccluded regions.

Prohibited Item Detection in X-Ray Images. Existing X-ray prohibited item detection methods are extended from conventional object detectors considering the characteristics of X-ray images. DOAM [6] leverages the different appearance information of the prohibited items to generate the attention maps, which can be used to refine feature maps for the detectors. LIM [24] suppresses the noisy information while activating the most identifiable features from the four directions. SDANet [7] consists of a dense attention module and a dependency refinement module to learn discriminative features and exploit the dependencies of multi-scale features, respectively. In this paper, we perform prohibited item detection from the perspective of the compositional model.

Attention Mechanism. Recently, attention mechanisms have been widely used for various computer vision tasks, such as image classification, object detection, and image segmentation. SENet [25] proposes a squeeze-and-excitation module to increase the weights of important channels and decrease the weights of unimportant channels. CBAM [26] models the inter-channel relation and the inter-spatial relation of features. Non-local network [27] captures long-range dependencies of any two locations.

Methodology

Overview

In VotingNet, a novel AHVM is designed based on widely-used detectors. We use a representative anchor-based method ATSS [28] and an anchor-free method FCOS [14], both of which have three prediction branches (a classification branch, a bounding box branch, and a centerness branch) and use FPN [29], as base detectors.

An overview of our proposed VotingNet is given in Fig. 1. First, the input image is fed into the backbone and FPN to obtain multi-scale feature maps (denoted as $\{\mathbf{F}^l \in \mathbb{R}^{D\times H_l\times W_l}\}_{l=1}^{L}$, where L is the number of feature maps; D, H_l and W_l are the channel number, height and width of $\mathbf{F}^l$, respectively). Each feature map is sent to a classification branch and a regression branch, where the regression branch involves a bounding box branch and a centerness branch.

For the centerness branch, AHVM converts the input feature map with the size of $D\times H_l\times W_l$ to $1\times H_l\times W_l$ and predicts the centerness of each location on the feature map. For the classification branch, the convolutional sub-branch and AHVM sub-branch generate two feature maps with the size $C\times H_l\times W_l$ (C is the number of categories), where these feature maps are combined to give the output (with the size of $C\times H_l\times W_l$) of the branch, predicting the category scores at each position. For the bounding box branch, the convolutional sub-branch and AHVM sub-branch give two feature maps with the size $4\times H_l\times W_l$, where the two feature maps are combined to give the regression result (with the size of $4\times H_l\times W_l$).

Adaptive Hough Voting Module (AHVM)

AHVM contains an Attention Block and a Voting Block.

Attention Block

Existing object detection methods [2, 4] leverage either the average votes as the voting result or a large convolutional layer to collect visual concepts. As a result, if an item is heavily occluded, most of the votes at a point come from occluders, while the votes from the unoccluded parts are insignificant, resulting in inaccurate detection of the occluded item. Therefore, although existing methods achieve good results in natural object detection, they have difficulty alleviating the negative influence of occluded areas of X-ray prohibited items.

To address the above problem, we introduce the Attention

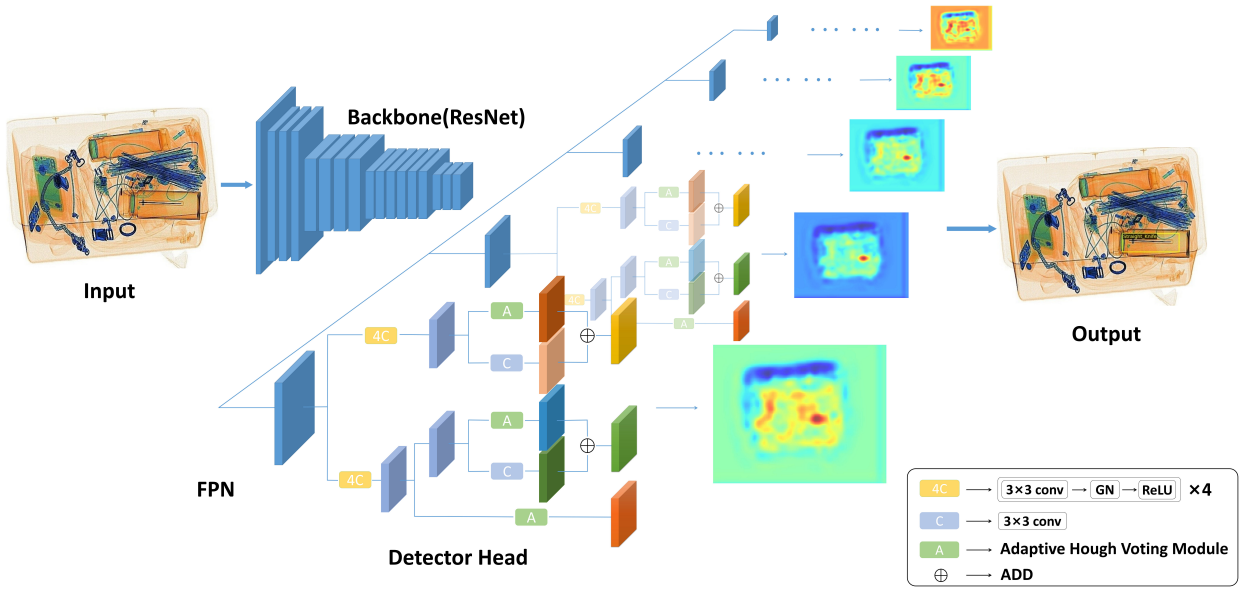

Figure 1. *Overview of VotingNet.*

Block to assign adaptive weights to the votes from different regions so that the voting result can be mainly determined by the unoccluded regions. Specifically, we first adopt a convolutional layer to transform the channel number of the input feature map to $N \times R$, where N is the number of predicted values ($N = C$ for the classification branch, $N = 4$ for the bounding box branch and $N = 1$ for the centerness branch) and R is the number of regions in the vote field. Then, we divide the transformed feature map into N part feature maps $\{\mathbf{P_1}, \mathbf{P_2}, \ldots, \mathbf{P_N}\}$, where $\mathbf{P_i} \in \mathbb{R}^{R \times H \times W}$, $i = 1, 2, \ldots, N$. Next, an extended CBAM is introduced to adaptively generate weights for inter-region and intra-region features for N tensors. In the extended CBAM, we use the channel attention module to calculate the weights at different channels of the feature map, which correspond to different voting regions. Meanwhile, we use the spatial attention module to calculate weights at different positions of the feature map which correspond to spatial positions within the same voting region. Note that we average the N tensors to obtain $\overline{\mathbf{P}} \in \mathbb{R}^{R \times H \times W}$, where the weights are calculated using $\overline{\mathbf{P}}$. Finally, we apply the obtained weights to all $\mathbf{P_i}$.

Channel Attention Module. The original channel attention in CBAM only exploits the maximum and average values of the original maps. However, for our detection task, the output of the detection head is the category score, the bounding box, and the centerness. In this case, the maximum, minimum, and average values can all be used to determine the feature distribution of the item to some extent. Therefore, we introduce the minimum pooling (MinPool) to the original channel attention in CBAM to better exploit the feature map, obtaining adaptive weights for inter-region features along the channel dimension.

To this end, we introduce the local pooling (LocalPool) operation. Specifically, instead of directly compressing a feature map with the size of $R \times H \times W$ to $R \times 1 \times 1$ by the global pooling (GlobalPool) in CBAM, we first compress the input feature map to the size of $R \times S \times S$ using a pooling (max pooling or min pooling) operation to get local max/min values, where S is a fixed value. Next, a global average pooling (GAP) operation is used to collect the local values from the above pooling operation to obtain a feature vector with the size of $R \times 1 \times 1$. We define the above operations as local maximum pooling (LocalMaxPool) and local minimum pooling (LocalMinPool) operations.

Based on the introduced LocalPool operation, our channel attention is given as follows: The feature map $\mathbf{F} \in \mathbb{R}^{R \times H \times W}$ first passes through LocalMaxPool and LocalMinPool, respectively, to generate two feature vectors. Then, they are fed into a shared multi-layer perception (MLP) followed by a Sigmoid function to obtain the weight of the channel $\mathbf{W_c}(\mathbf{F}) \in \mathbb{R}^{R \times 1 \times 1}$. Mathematically, the channel attention is computed as:

$$\mathbf{W_c}(\mathbf{F}) = \text{Sigmoid}(\text{MLP}(\text{LocalMaxPool}(\mathbf{F})) + \text{MLP}(\text{LocalMinPool}(\mathbf{F}))) \tag{1}$$

where $\text{Sigmoid}(\cdot)$, $\text{MLP}(\cdot)$, $\text{LocalMaxPool}(\cdot)$ and $\text{LocalMinPool}(\cdot)$ denote the Sigmoid function, the MLP operation, the LocalMaxPool operation and the LocalMinPool operation, respectively.

The above calculation of channel weights $\mathbf{W_c}(\mathbf{F})$ can be applied to part feature maps to obtain the weights of inter-region features, which are calculated as:

$$\mathbf{P_i'} = \text{REPEAT}(\mathbf{W_R}(\overline{\mathbf{P}})) \odot \mathbf{P_i}, i = 1, 2, \ldots, N \tag{2}$$

where $\text{REPEAT}(\cdot)$ means extending the number of channels from R to N by replication, $\odot$ denotes element-wise multiplication, which uses the operation of broadcast on space.

Spatial Attention Module. The original spatial attention module in CBAM first performs maximum pooling and average pooling operations, respectively, along the channel dimension and concatenates the two $1 \times H \times W$ feature maps along the channel dimension to obtain a $2 \times H \times W$ feature map, which is then compressed to $1 \times H \times W$ by a convolution operation. Finally, spatial weights are obtained by a Sigmoid activation function.

We also introduce the min pooling operation by concatenating the feature maps through MaxPool, AvgPool, and MinPool to

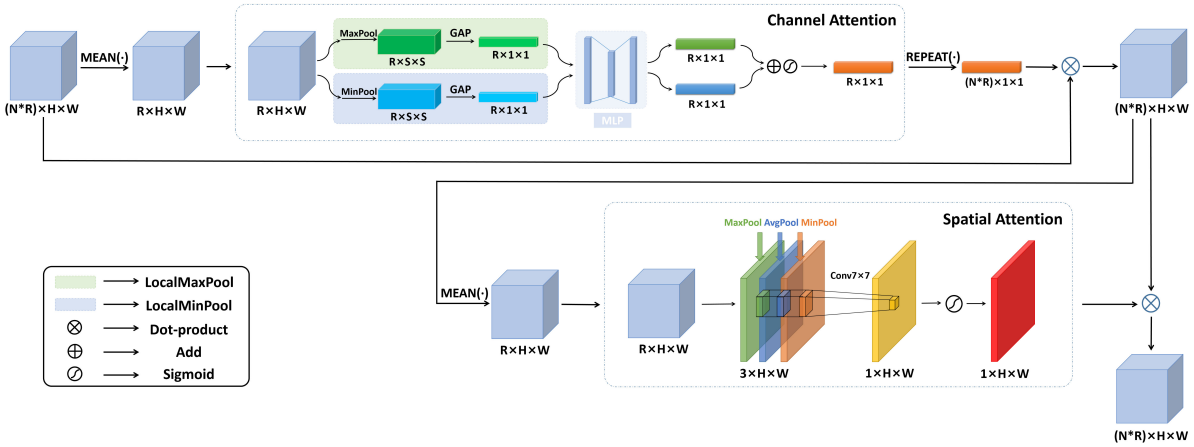

Figure 2. *The network of Attention Block.*

obtain a feature map, of size $3 \times H \times W$ and then perform the convolution and activation operations to calculate the spatial weights $\mathbf{W_s}(\mathbf{F}) \in \mathbb{R}^{H \times W}$. Mathematically, the spatial attention is computed as:

$$\mathbf{W_s}(\mathbf{F}) = \text{Sigmoid}(f^{7\times7}([\text{MaxPool}(\mathbf{F}); \text{AvgPool}(\mathbf{F}); \text{MinPool}(\mathbf{F})])) \tag{3}$$

where $f^{7\times7}$ represents a convolution operation with the kernel size of 7×7. Sigmoid$(\cdot)$, AvgPool$(\cdot)$ and MinPool$(\cdot)$ denote the Sigmoid, AvgPool, and MinPool operations, respectively.

The above calculation of spatial weights $\mathbf{W_s}(\mathbf{F})$ can be applied to part feature maps to obtain the weights of intra-region features, which are calculated as:

$$\mathbf{P_i''} = \mathbf{W_R}(\overline{\mathbf{P'}}) \odot \mathbf{P_i'}, i = 1, 2, \ldots, N \tag{4}$$

where $\mathbf{P_i'}$ can be calculated from Equation 2, $\odot$ requires the use of broadcast on channel.

The flow chart of the Attention Block is shown in Figure 2.

Voting Block

Since the spatial relationship between the parts of the item is represented visually by the Attention Block, the Voting Block collects the votes from the corresponding positions. Inspired by the voting module in [4], the Voting Block is as follows.

The output of the Attention Block is N part feature maps $\{\mathbf{P_1''}, \mathbf{P_2''}, \ldots, \mathbf{P_N''}\}$, where $\mathbf{P_i''} \in \mathbb{R}^{H \times W \times R}$, $i = 1, 2, \ldots, N$ (calculated from Equation 4), which is also the input to the Voting Block.

The output of the Voting Block is the voting result maps $\mathbf{V}$, where $\mathbf{V} = \{\mathbf{V_i} \in \mathbb{R}^{H \times W \times 1} | i = 1, 2, \ldots, N\}$, and the outputs of the three prediction branches correspond to the output $\mathbf{V_{cls}}$, $\mathbf{V_{bbox}}$ and $\mathbf{V_{centerness}}$. Then, the peaks in the $\mathbf{V_{cls}}$ / $\mathbf{V_{centerness}}$ indicate the presence of object instances. At the same time, the value corresponding to the same position on the $\mathbf{V_{bbox}}$ is the bounding box regression value of the object.

The voting process converts the part feature maps (e.g., $\mathbf{P_n}$) to voting result maps (e.g., $\mathbf{V_n}$), which takes in a part feature map as the input and generates a voting result map. Specifically, for feature layers of different sizes, there are different receptive fields responsible for detecting prohibited items of different sizes. Therefore, we need to adjust the size of the vote field of the Voting Block according to the size of each feature map's receptive fields, so that the vote results come from the various parts that make up the object and context.

Joint Loss Function

The joint loss function for the model is given as:

$$\begin{aligned} \mathcal{L}_{Joint} &= \gamma_1 \mathcal{L}_{cls} + \gamma_2 \mathcal{L}_{bbox} + \gamma_3 \mathcal{L}_{centerness} \\ &= \sum_{i=1}^{3} \gamma_i (\lambda_{conv_i} \mathcal{L}_{conv_i} + \lambda_{ahvm_i} \mathcal{L}_{ahvm_i}) \end{aligned} \tag{5}$$

where i=1,2,3 correspond to the classification branch, the bounding box branch, and the centerness branch respectively, γ and λ are hyperparameters. For the classification branches, $\lambda_{conv_1} = \lambda_{ahvm_1} = 1.0$ and FocalLoss [11] is used for $\mathcal{L}_{conv_1}$ and $\mathcal{L}_{ahvm_1}$; For the bbox branches, $\lambda_{conv_2} = \lambda_{ahvm_2} = 1.0$ and GIoULoss [30] is used for $\mathcal{L}_{conv_2}$ and $\mathcal{L}_{ahvm_2}$; For the cneterness branch, $\lambda_{conv_3} = 0$, $\lambda_{ahvm_3} = 1.0$ (includes AHVM only) and CrossEntropyLoss is used for $\mathcal{L}_{conv_3}$ and $\mathcal{L}_{ahvm_3}$.

Experiments

First, we introduce the experimental settings. Next, we perform ablation studies to demonstrate the effectiveness of our AHVM. Finally, we compare our proposed method with several state-of-the-art methods on OPIXray and PIDray datasets.

Experimental Settings

Datasets All the methods are trained and tested on two datasets: OPIXray [6] and PIDray [7]. The OPIXray dataset contains a total of 8,885 X-ray images (with the size of $1,225 \times 954$) of 5 categories (i.e., folding knife, straight knife, scissor, utility knife, and multi-tool knife), where the test set can be divided into OL1, OL2, and OL3 according to the different levels of occlusion. The PIDray dataset contains 12 categories of prohibited items (i.e., gun, knife, wrench, pliers, scissors, hammer, handcuffs, baton, sprayer, powerbank, lighter, and bullet) with 47,677 X-ray images. For the PIDray dataset, we resize the image to 500×500. As our main focus is on the detection of occluded prohibited items, the hidden test set is used to evaluate the performance of the model.

To demonstrate the generalization of our method, we choose an anchor-based method ATSS and an anchor-free method FCOS

Ablation Studies for AHVM. AP(%) is used to evaluate the performance of methods on different object occlusion levels.

Method	OL1	OL2	OL3	Overall
ATSS	42.9	40.6	39.2	41.2
ATSS+VB	44.1	41.5	39.5	42.2
ATSS+CBAM+VB	44.0	41.5	40.1	42.1
ATSS+AB+VB	**44.8**	**41.8**	**40.6**	**42.8**

as base detectors and incorporate AHVM into base detectors. Both ATSS and FCOS contain three prediction branches (a classification branch, a bounding box branch, and a centerness branch), but they differ in sample matching.

Implementation details Our proposed VotingNet is implemented on the MMDetection toolkit[1]. All the results are reported on a machine with an NVIDIA RTX A5000. The whole network is trained with a stochastic gradient descent (SGD) algorithm with a momentum of 0.9 and a weight decay of 0.0001. The initial learning rate is set as 0.01 and the batch size is set as 16. Unless otherwise specified, other parameters involved in the experiments follow the settings of the MMDetection toolkit.

Evaluation Metrics We evaluate the performance using the AP metrics in [31]. The AP score is averaged across all 10 IoU (Intersection over Union) thresholds (between 0.50 and 0.95) and all the categories (5 for OPIXray and 12 for PIDray). We also give AP_{50} and AP_{75} scores, which are calculated at IoU = 0.50 and IoU = 0.75, respectively.

Ablation Studies

We conduct ablation studies to analyze the influence of the key modules on the OPIXray dataset at different levels of occlusion. The results are given in Table 1, where we add the key components one by one into the baseline ATSS.

First, by adding the Voting Block (VB) into ATSS, ATSS+VB improves the performance of the baseline ATSS method by 1.2% AP, 0.9% AP, 0.3% AP, and 1.0% AP on OL1, OL2, OL3 and overall, respectively. Then, by incorporating both Attention Block (AB) and Voting Block (VB) into the baseline, the performance of model on OL1, OL2, OL3 and Overall improved by 1.9% AP, 1.2% AP, 1.4% AP and 1.6% AP, respectively. We also replace the AB in VotingNet with CBAM for comparison. ATSS+AB+VB achieves higher accuracy than ATSS+CBAM+VB. This shows the effectiveness of our proposed extended CBAM, specifically designed for prohibited item detection. In particular, all the variants except for the ATSS leverage a combination of the convolution network and compositional model. The convolutional network can extract discriminative features, while the compositional model is robust to occlusion. Their combination can improve the final performance.

Comparison with State-of-the-Art Methods

We first compare our method with some state-of-the-art object detectors on the OPIXray and the PIDray datasets. The results are given in Table 2.

Compared with the original ATSS and FCOS methods, VotingNet gives 1.6% and 2.5% improvements in terms of AP on two datasets, respectively. Moreover, VotingNet outperforms other competing methods in most evaluation metrics. Incorporating AHVM into the base detectors can improve the detection of occluded prohibited items. Therefore, introducing the generalized Hough transform to the compositional model effectively facilitates the robustness to occlusion.

[1] https://github.com/open-mmlab/mmdetection

The evaluation results on the OPIXray and PIDray dataset. AP, AP_{50}, **and** AP_{75} **(%) are used to evaluate the performance of all methods.**

Method	OPIXray Dateset			PIDray Dateset		
	AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}
Faster-RCNN [8]	40.4	89.5	26.8	45.9	64.8	52.6
Cascade-RCNN [9]	40.6	90.1	27.0	48.2	63.7	54.2
SSD300 [10]	33.9	80.0	20.1	42.9	64.9	47.5
YOLOv3 [32]	37.1	88.2	21.5	44.6	67.6	50.6
RetinaNet [11]	40.9	90.1	26.5	44.6	63.2	50.0
free-anchor [33]	41.0	90.5	25.7	46.4	64.7	52.3
OAP [17]	38.4	88.7	23.2	45.2	63.6	50.8
FCOS [14]	40.7	90.0	27.4	44.3	62.9	49.8
VotingNet (FCOS)	41.5	**91.3**	28.9	46.7	65.5	53.7
ATSS [28]	41.2	89.8	28.1	46.1	63.4	52.3
VotingNet (ATSS)	**42.8**	90.7	**30.8**	**48.6**	**66.0**	**54.6**

Conclusion

In this paper, we propose VotingNet based on the compositional model, which contains an AHVM located at the detection head. AHVM consists of AB and VB, which assign higher weights to the unoccluded parts of the prohibited item and perform voting. Therefore, AHVM can focus more on the unoccluded parts of the object when locating prohibited items and filter out the influence of occluded items, enabling more accurate detection of prohibited items under occlusion. Experiments show the effectiveness of VotingNet.

Acknowledgments

This work was partly supported by the National Natural Science Foundation of China under Grants 62372388, 62071404, and U21A20514, and by the Fundamental Research Funds for the Central Universities under Grant 20720240076.

References

[1] A. Wang, Y. Sun, A. Kortylewski, and A. L. Yuille, "Robust object detection under occlusion with context-aware compositionalnets," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 12 645–12 654.

[2] Z. Zhang, C. Xie, J. Wang, L. Xie, and A. L. Yuille, "Deepvoting: A robust and explainable deep network for semantic part detection under partial occlusion," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1372–1380.

[3] H. Zhu, P. Tang, J. Park, S. Park, and A. Yuille, "Robustness of object recognition under extreme occlusion in humans and computational models," *arXiv preprint arXiv:1905.04598*, 2019.

[4] N. Samet, S. Hicsonmez, and E. Akbas, "Houghnet: Integrating near and long-range evidence for visual detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[5] D. H. Ballard, "Generalizing the hough transform to detect arbitrary shapes," *Pattern recognition*, vol. 13, no. 2, pp. 111–122, 1981.

[6] Y. Wei, R. Tao, Z. Wu, Y. Ma, L. Zhang, and X. Liu, "Occluded prohibited items detection: An x-ray security inspection benchmark and de-occlusion attention module," in *Proceedings of the ACM International Conference on Multimedia*, 2020, pp. 138–146.

[7] B. Wang, L. Zhang, L. Wen, X. Liu, and Y. Wu, "Towards real-world prohibited item detection: A large-scale x-ray benchmark," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5412–5421.

[8] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in Neural Information Processing S9ystems*, vol. 28, 2015.

[9] Z. Cai and N. Vasconcelos, "Cascade r-cnn: Delving into high quality object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern rRecognition*, 2018, pp. 6154–6162.

[10] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *Proceedings of European Conference on Computer Vision*, 2016.

[11] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2980–2988.

[12] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 779–788.

[13] L. Huang, Y. Yang, Y. Deng, and Y. Yu, "Densebox: Unifying landmark localization with end to end object detection," *arXiv preprint arXiv:1509.04874*, 2015.

[14] Z. Tian, C. Shen, H. Chen, and T. He, "Fcos: Fully convolutional one-stage object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 9627–9636.

[15] H. Law and J. Deng, "Cornernet: Detecting objects as paired keypoints," in *Proceedings of European Conference on Computer Vision*, 2018, pp. 734–750.

[16] X. Zhou, J. Zhuo, and P. Krahenbuhl, "Bottom-up object detection by grouping extreme and center points," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 850–859.

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

[18] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "Centernet: Keypoint triplets for object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 6569–6578.

[19] S. Yan and Q. Liu, "Inferring occluded features for fast object detection," *Signal processing*, vol. 110, pp. 188–198, 2015.

[20] N. D. Reddy, M. Vo, and S. G. Narasimhan, "Occlusionnet: 2d/3d occluded keypoint localization using graph networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7326–7335.

[21] S. Zhang, L. Wen, X. Bian, Z. Lei, and S. Z. Li, "Occlusion-aware r-cnn: detecting pedestrians in a crowd," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 637–653.

[22] Y. Xiang and S. Savarese, "Object detection by 3d aspectlets and occlusion reasoning," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2013, pp. 530–537.

[23] J. Wang, C. Xie, Z. Zhang, J. Zhu, L. Xie, and A. Yuille, "Detecting semantic parts on partially occluded objects," *arXiv preprint arXiv:1707.07819*, 2017.

[24] R. Tao, Y. Wei, X. Jiang, H. Li, H. Qin, J. Wang, Y. Ma, L. Zhang, and X. Liu, "Towards real-world x-ray security inspection: A high-quality benchmark and lateral inhibition module for prohibited items detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 923–10 932.

[25] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7132–7141.

[26] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 3–19.

[27] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7794–7803.

[28] S. Zhang, C. Chi, Y. Yao, Z. Lei, and S. Z. Li, "Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9759–9768.

[29] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2117–2125.

[30] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 658–666.

[31] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Proceedings of the European Conference on Computer Vision*, 2014, pp. 740–755.

[32] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[33] X. Zhang, F. Wan, C. Liu, R. Ji, and Q. Ye, "Freeanchor: Learning to match anchors for visual object detection," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

Author Biography

Kaitao Huang *is a master's student majoring in computer science at Xiamen University in China. He received his B.S. degree from Xiamen University in 2024. His main research area is computer vision.*

Yan Yan *is a professor at Xiamen University, China. He received his Ph.D. degree from Tsinghua University, China, in 2009. His main research area is computer vision.*