X-ray Security Inspection Based on Edge Enhancement Module

Chenyang Wang(23020221154175), Hao Yang(31520221154232), Ruikang Chen(23020221154074),
Xueting Chen(36920221153070), Yifan Zhang(31520221154236)
Department of Computer Science, Xiamen University, China
Department of Artificial Intelligence, Xiamen University, China
Institute of Artificial Intelligence, Xiamen University, China

Abstract

The use of object detection methods in x-ray security inspection can significantly reduces the consumption of human resources. However, the current mainstream object detection methods are aimed at the detection of natural objects, and the effect of these methods is often unsatisfactory for x-ray security inspection images that contain extremely serious occlusion. In x-ray images, humans generally use the edges of the images for item recognition. Inspired by this idea, we designed a object detector specifically for x-ray security inspection, which is based on the most advanced methods such as RetinaNet and an edge enhancement module that makes good use of the edge information in the pictures. The final experiment results show that the developed object detector dedicated to x-ray security inspection has good performance and excellent generalization.

Introduction

With the vigorous development of modern public transportation, security inspection has become more and more critical in protecting public safety. As an effective preventive measure for terrorist attacks and crimes worldwide, X-ray scanners usually are adopted by security inspection to find whether there is any prohibited item in passenger luggage. However, With both increased passenger throughput in the global travel network and an increasing focus on broader aspects of comprehensive border security (e.g. freight, postal), more and more inspectors are struggling to find prohibited items hidden in cluttered luggage. Therefore, a system that can automatically detect prohibited items in x-ray pictures is very useful, and it can relieve many security inspectors from the tedious task.

As the technology of deep learning and computer vision technologies (Sermanet et al. 2014; Liu et al. 2016; Tian et al. 2019; Ji et al. 2019b; 2019a; Li et al. 2020; Cai et al. 2019) develops, the recognition of occluded prohibited items from X-ray images can be regarded as an object detection problem of computer vision, which has been widely studied in the literature. However, compared with natural images, X-ray images (as Fig.1) has a quite different appearance and edges of objects and background. Most previous object detection algorithms in computer vision are designed to detect objects in natural images, which are not optimal for detection in X-ray images. In addition, the images of the objects stacked in the baggage often occlude with each other. Unlike the occlusion problem in optical images, occluded objects are still visible in the X-ray security images. Due to the occlusion of the images, the detection of the occluded object is disturbed. Therefore, directly applying methods designed for natural images will lead to decreased performance.

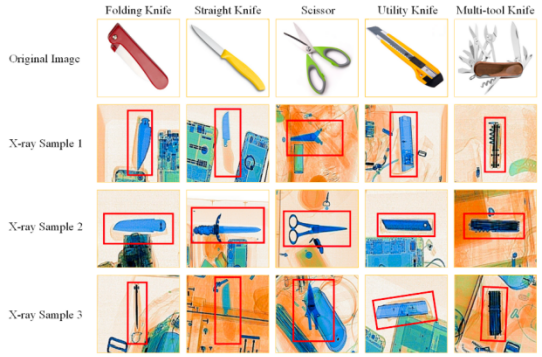

Figure 1: Samples of the five categories of cutters and corresponding X-ray images

In view of these characteristics of X-ray images, researchers have conducted in-depth exploration. (Akcay and Breckon 2020) used image augments to remove noisy and fuzzy images and evaluated the performance of CNN and Autoencoder. (AYDIN, KARAKOSE, and AKIN 2018; Cui and Oztan 2019; Liang et al. 2018; Dhiraj and Jain 2019) evaluated the performance of CNN, RetinaNet and other deep learning methods on X-ray images.

What' more, attention mechanism has been used in object detection in recent year. The essence of attention mechanism is to imitate human visual attention, which can quickly focus on useful information in a large amount of information.Previous works(Lim et al. 2021; Peng et al. 2021; Liu et al. 2022) show that attention mechanism can improve the performance of object detection model. Swim transformer is used in our work to introduce attention mechanism, and the details will be presented in later chapters.

In this project, we aim to complete the task of detecting prohibited items in X-ray images. To alleviate the occlusion problem, we propose an edge enhancement module based on RetinaNet(Lin et al. 2020), which can improve the model's performance. Moreover, we use technologies such as data enhancement to achieve higher accuracy. More details can be found in the "Plan" section.

Related Work

X-ray Image Datasets

X-rays have strong penetrating power, making occluded objects visible in the image, so they are widely used in security inspections. Many datasets for object detection in X-ray security images appear in many papers. The X-ray dataset (GDX-ray)(Mery et al. 2015) contains multi-view images, typically used for classification tasks. The luggage group is the data required for object detection in X-ray security images. The dataset contains 8150 X-ray photographs, arranged as 77 series. X-ray images are taken from containers such as backpacks, pencil cases, wallets, etc. Security X-ray (SIXray)(Miao et al. 2019) is used to study the class imbalance problem. In total, SIXray contains 1,059,231 X-ray images, of which 8,929 are labelled. The images were collected from several subway stations, with raw metadata indicating the presence or absence of prohibited items. There are six everyday prohibited items: guns, knives, wrenches, pliers, scissors, and hammers. Occlusion of Prohibited Items X-ray (OPIXray)(Wei et al. 2020) is the first high-quality object detection dataset for security inspection. OPIXray contains a total of 8885 x-ray images of 5 types of knives: folding knives, straight knives, scissors, utility knives, and multi-tool knives. The security inspection machine scans the backgrounds of all samples, and the prohibited items are synthesized into these backgrounds by professional software.

Objection Detection

Object detection is one of the fundamental tasks in the field of computer vision, and the sliding window mode, in which classifiers are applied to dense grids of images, has a long and rich history. DPMs(Felzenszwalb, Girshick, and McAllester 2010) help to extend dense detectors to more general object classes and have achieved state-of-the-art results for many years on PASCAL. While sliding window methods were the leading detection paradigm in classical computer vision, with the renaissance of deep learning(Fu et al. 2017), the two-stage detectors described next quickly dominated object detection. The dominant paradigm of modern object detection is based on a two-stage approach. As pioneered in selective search work(Redmon et al. 2016), the first stage generates a sparse set of candidate proposals that should contain all objects while filtering out most negative locations, and the second stage classifies proposals into foreground classes or backgrounds. R-CNN(Redmon and Farhadi 2017) upgraded the second stage classifier to a convolutional network, yielding considerable gains in accuracy and ushering in the modern era of object detection. The Region Proposal Network (RPN) integrates proposal generation with a second-stage classifier into a single convolutional network, resulting in a faster RCNN framework(Everingham et al. 2009). OverFeat(Krizhevsky, Sutskever, and Hinton 2012) is one of the first modern single-stage object detectors based on deep networks. Subsequently, SSD(Uijlings et al. 2013; Girshick et al. 2014) and YOLO(Ren et al. 2015) revived interest in one-stage methods. These detectors have been tuned for speed, but their accuracy lags behind two-stage methods. These detectors have been tuned for speed, but their accuracy lags behind two-stage methods. SSDs have 10-20% lower AP, while YOLO focuses on more extreme speed/accuracy tradeoffs.

Attention Mechanism

The attention mechanism has been widely used in various tasks in recent years. The essence of the attention mechanism is to imitate human visual attention, which can quickly filter out discriminative information from a large amount of information. Various attention mechanisms have been proposed. SENet(Hu et al. 2020) proposes squeeze and excitation modules to model the interdependencies between channels. CBAM(Woo et al. 2018) models the inter-channel and inter-spatial relationships of features. The Non-Local network(Wang et al. 2018) can directly capture the long-range dependencies of any two locations, computing the weighted sum of the features of all locations in the input feature map as the response of a location. Since many previous works(Lin et al. 2017; Liu et al. 2018) have shown the importance of multi-scale feature fusion, we consider it an essential technique for solving the problem of prohibited item detection. In X-ray images, many vital details of objects are lost, such as texture and appearance information. Moreover, the contours of objects overlap, which also brings great challenges to detection. Therefore, we propose a selective dense attention network.

Method

Overall Architecture

In this paper, to alleviate the occlusion problem due to the penetration effect in x-ray images, we improve on RetinaNet by proposing an edge enhancement module that uses an edge detection operator to extract edge information from the original image and uses this information to extend the original image from 3 channels to more channels. We also use a variety of solid data enhancement approaches and incorporate methods such as Swin Transformer(Bochkovskiy, Wang, and Liao 2020) into our network. In the process of data enhancement, in addition to conventional methods such as resizing, we also incorporate data enhancement methods such as mosaic(Bochkovskiy, Wang, and Liao 2020), mixup(Zhang et al. 2018), and copy-paste(Ghiasi et al. 2021); in the process of feature extraction, we use the current best Transformer-based backbone, called Swin Transformer, and use nasfpn(Ghiasi et al. 2019) as the neck for feature fusion and enhancement; in the final prediction head, we borrow the idea of yolox(Ge et al. 2021), use decoupled prediction head and design three branches for prediction. Fig.2 show the overall structure of the model network.

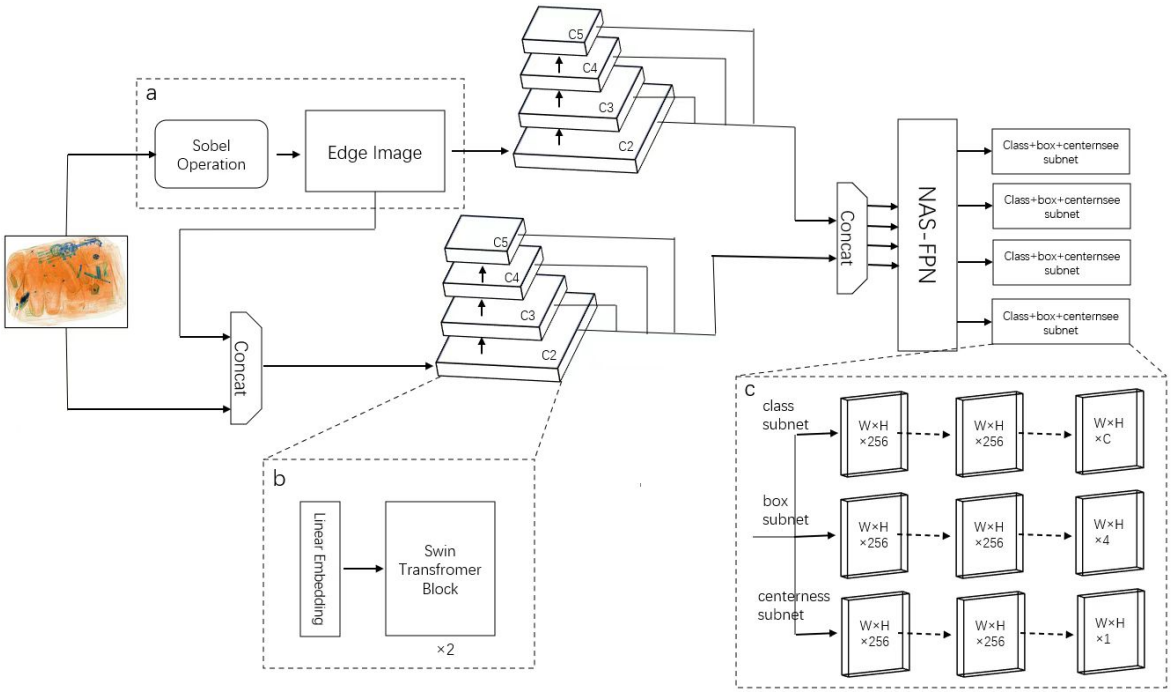

Figure 2: Overall architecture of the model network

Data Augmentation

Mosaic This data enhancement method takes four images and stitches them together by random scaling, cropping, and lining up. It enriches the background and small targets of the detected objects. When calculating Batch Normalization, it calculates the data of four images at a time so that the mini-batch size does not need to be significant, and a GPU can achieve better results.

Mixup MixUp data enhancement is a pixel-by-pixel overlay of two images in random proportions and then integrates the labels of the subgraphs together as the labels of the mixed images.

CopyPaste CopyPaste is similar to MixUp but copies only the pixels of the instance, not all the pixels in the instance's detection frame. First, two images are randomly selected, each with random scale dithering, and then some instances are randomly selected from one image and directly pasted to the other image while updating the detection frame, category labels and masks.

Edge Enhancement Module

Since the occlusion in x-ray images is very serious, we generally recognize them by edges. Therefore, this paper proposes an edge enhancement module for extracting the edges in x-ray images, as illustrated in Fig.3.

The edge enhancement module uses various operators, such as the Sobel operator, Roberts operator, Prewitt operator, etc., for edge extraction, and all the operators can be selected in terms of type and number. When the required operators are selected, the original image is extracted using these operators to obtain the extracted edge map. Then all the edge maps are concatenated with the original image to obtain the enhanced original image. The number of channels depends on the number of edge detection operators used.

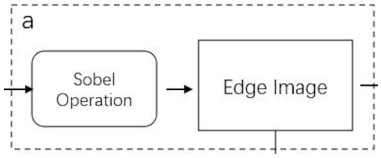

Figure 3: Edge enhancement module

In addition, this paper also designs a separate feature extraction network for the edge detection map, which uses Swin Transformer to obtain the edge feature map, and concatenate the edge feature map with the original feature map before the backbone is input to the neck and then the hybrid feature map is input to the neck to continue the feature fusion and then the hybrid feature map is input to the neck for feature fusion and enhancement.

Feature Extraction

Feature extraction consists of the backbone and neck. The backbone is mainly used for initial feature extraction, while the neck is primarily used for feature fusion and enhancement, which can fuse the spatial information in the bottom layer of the backbone with the rich semantic information in the top layer to obtain a more effective feature map.

Our backbone uses Swin Transformer, as illustrated in Fig.4, which is a new feature extraction network based on

Transformer. Swin Transformer uses the idea of sliding windows to compute self-attention in the local range, which can effectively reduce the number of parameters in the backbone. The whole Swin Transformer contains four stages. When an image of size 448*448 is input into Swin Transformer, it will pass through each stage in turn to produce feature map outputs C2, C3, C4 and C5 with a gradually increasing number of channels and decreasing resolution.

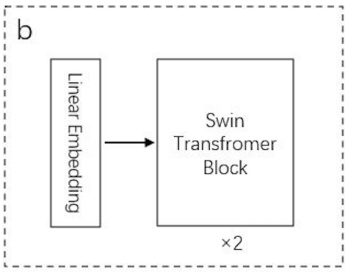

Figure 4: The backbone of the network is Swin Transformer

After obtaining the four feature maps from the backbone, we input them into NAS-FPN for feature fusion and enhancement, which uses network structure search to select the best model structure in the search space using augmentation learning. It receives C2, C3, C4, and C5 as input and obtains C6 by performing a maximum pooling operation with a stride of 2 on C5, and inputs the five feature maps into NAS-FPN to obtain the feature maps P2, P3, P4, P5, and P6 after feature fusion.

Decoupled Head

After the feature extraction, we input the feature maps into the prediction head. All prediction heads share the same parameters and use a decoupled head with three branches: classification branch, regression branch and Centerness branch, as illustrated in Fig.5. Each prediction head contains four stacked convolutional layers, and after the operation of the feature map, we output the feature maps with channel sizes C, 4, and 1, which correspond to the classification branch, regression branch, and Centerness branch, where C indicates the number of categories in the dataset, 4 indicates the four coordinates of the bounding box, and one is used to distinguish between foreground and background.

Loss

There are three major losses used in the network, which are classification loss, regression loss and Centerness loss.

Classification Loss There are two reasons for choosing focal loss: first, when assigning positive and negative samples, the number of negative samples is too large and will occupy most of the loss, and many of these negative samples are simple samples, which makes the model optimization direction not as we hope, and focal loss by reducing the weight of easy to classify; Second, since some classes occupy the majority of the dataset and some classes are very small in number, this also makes the model tend to give the classification to the class with the large number during training,

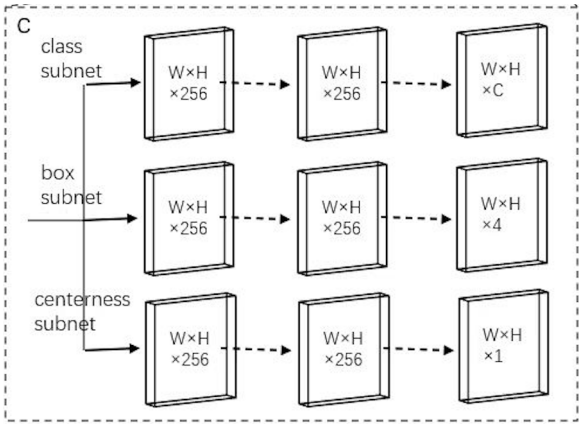

Figure 5: Decoupled head consists of three branches, classification branch, regression branch and Centerness branch

and the multi-classification version of focal loss can solve this problem well. Focal Loss formula is shown in Eqa.1. $(1-p_t)$ is the modulation factor, α and γ are the hyperparameter to control the modulation factor, it means that when the network has greater confidence in the prediction result, its proportion in the loss is smaller, it allows the network to focus on learning more difficult samples.

$$Class_{Loss} = FL(p_t) = -\alpha_t(1-p_t)^{\gamma}log(p_t) \quad (1)$$

Regression Loss The regression loss uses giou loss (Rezatofighi et al. 2019). The main reason for choosing giou loss is that when the traditional iou is used as the loss function, if the two boxes do not intersect, then iou = 0, and this time, because the loss is also 0, the gradient cannot be back propagated and cannot learn the training. In contrast, giou takes into account the proportion of the region that does not belong to the two boxes in the closed region to the whole closed region so that it can solve the above problem well. GIoU Loss formula is shown in Eqa.2 and Eqa.3. a and b are the ground truth box and prediction box, c is the smallest enclosing rectangle covering a and b.

$$GIoU = IoU - \frac{area_c - area_{a\cup b}}{area_c} \quad (2)$$

$$Reg_{Loss} = L_{GIoU} = 1 - GIoU \quad (3)$$

Centerness Loss The final Centerness loss uses the cross-entropy loss function, which is consistent with that in FCOS. Our Centerness branch is independent of the other two branches, and in our tests we found no difference between the effect of the Centerness branch and the other two branches together and not together, with the following formula Eqa.4 for Centerness.

$$Centerness = \sqrt{\frac{min(l\times r)}{max(l\times r)} \times \frac{min(t,b)}{max(t,b)}} \quad (4)$$

where l, r, t, b denote the distance from the center point of bounding box to each side of this bounding box. The final

Table 1: The category distribution of the OPIXray dataset. Due to that some images contain more than one prohibited item, the sum of all items in the different categories is greater than the total number of images.

OPIXray	Categories					Total
	Folding	Straight	Scissor	Utility	Multi-tool	
Training	1589	809	1494	1635	1612	7109
Testing	404	235	369	343	430	1776
Total	1993	1044	1863	1978	2042	8885

Centerness value is optimized by BCELoss, which has the following formula illustrated by Eqa.5

$$Center_{Loss} = -C \times log(C^*) - (1-C) \times log(1-C^*) \quad (5)$$

where C denotes the true value of Centerness and C^* denotes the Centerness calculated from the bounding box.

Total Loss Through the weighted combination of the above methods,we show the total loss in Eqa.6

$$L = Class_{Loss} + W_R \times Reg_{Loss} + W_C \times Center_{Loss} \quad (6)$$

where $Class_{Loss}, Reg_{Loss}, Center_{Loss}$ denote the loss of classification, regression and Centerness, and W_R, W_C denote the weights of regression loss and Centerness loss.

Experiments

In this section, we prepare some experiments to validate our model. The OPIXray dataset is used for the experiments, and the details of the dataset will be described below. We also designed experiments to verify the superiority of the proposed method over the baseline to demonstrate the effectiveness of the method in this paper. Finally, we also designed ablation experiments to demonstrate that all the modules designed in this paper are valid.

Dataset

All images of OPIXray dataset are scanned by security inspection machine and annotated manually by professional inspectors from an international airport, and the standard of annotating is based on the standard of training security inspectors. And OPIXray dataset contains a total of 8885 X-ray images(7019 for training, 1776 for testing), including 5 categories of cutters, namely, Folding Knife, Straight Knife, Scissor, Utility Knife, Multi-tool Knife. In order to study the impact brought by object occlusion levels, it divide the testing set into three subsets and name them Occlusion Level 1 (OL1), Occlusion Level 2 (OL2) and Occlusion Level 3 (OL3), where the number indicates occlusion level of prohibited items in images. Examples of graphs for datasets of different difficulty levels are shown in Fig.6. The number of categories and the total number for this dataset are shown in Tab.1. The information structure of annotation file is as follows: image name, category, top-left position of prohibited item (x_1, y_1), bottom-right position of prohibited item (x_2, y_2). You can check some examples in OPIXray in Fig.1.

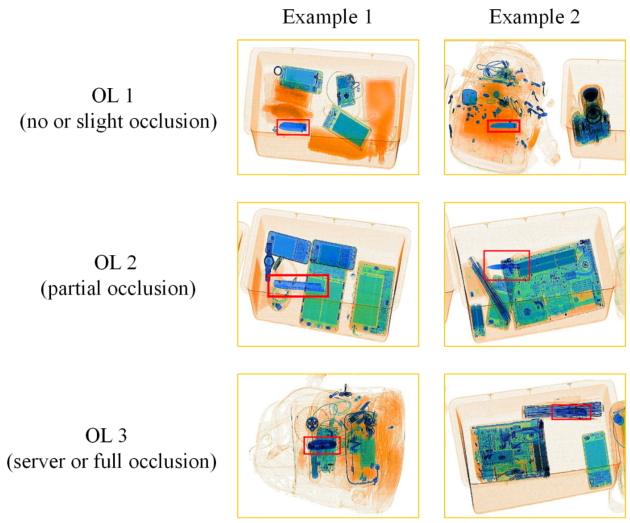

Figure 6: Samples of different occlusion levels

Experimental Details

In all our experiments, the model was optimized using the SDG optimizer, with the learning rate set to 0.0025, the batch size set to 4, the momentum set to 0.9, and the weight decay set to 0.0001. A total of 70 training rounds were eventually performed. The graphics card was an NVIDIA GTX 3090 with 24 GB of video memory. mAP, mean Average Precision, was used to evaluate the performance of the model and the IoU threshold was set to 0.5.

Comparing with Other Models

We compared some of the more commonly used classical target detectors with our proposed method, and the specific results can be viewed in Tab.2. From this experimental result, we can see that our model improves the performance over SSD, YOLOv3 and FCOS by 15.68%, 8.36% and 4.55%, respectively. It can also be seen that in the class-by-class results, our proposed method gets the best results in the Folding, Straight, Scissor, and Utility classes, and improves 2.35%, 0.76%, 0.88%, and 3.94% over the second place method, respectively, which makes the overall performance of our method compared to the second place method, which is FCOS was able to have an improvement of 4.55%. And we also notice that on the Multi-tool class, our method is 6.17% lower than the highest performing method with 94.37%. We guess that the shape of the Multi-tool class is more similar to many other objects, and after using the edge enhancement module, more shapes similar to this class may be generated, leading to many false positives from the detector, which in turn leads to degradation of its performance.

Visualization

In this section, we visualize some of the detection results of the detector in Fig.7. Each of these columns represents different kinds of results, and each row represents results of different difficulties, for a total of three difficulties.

Figure 7: Visualization

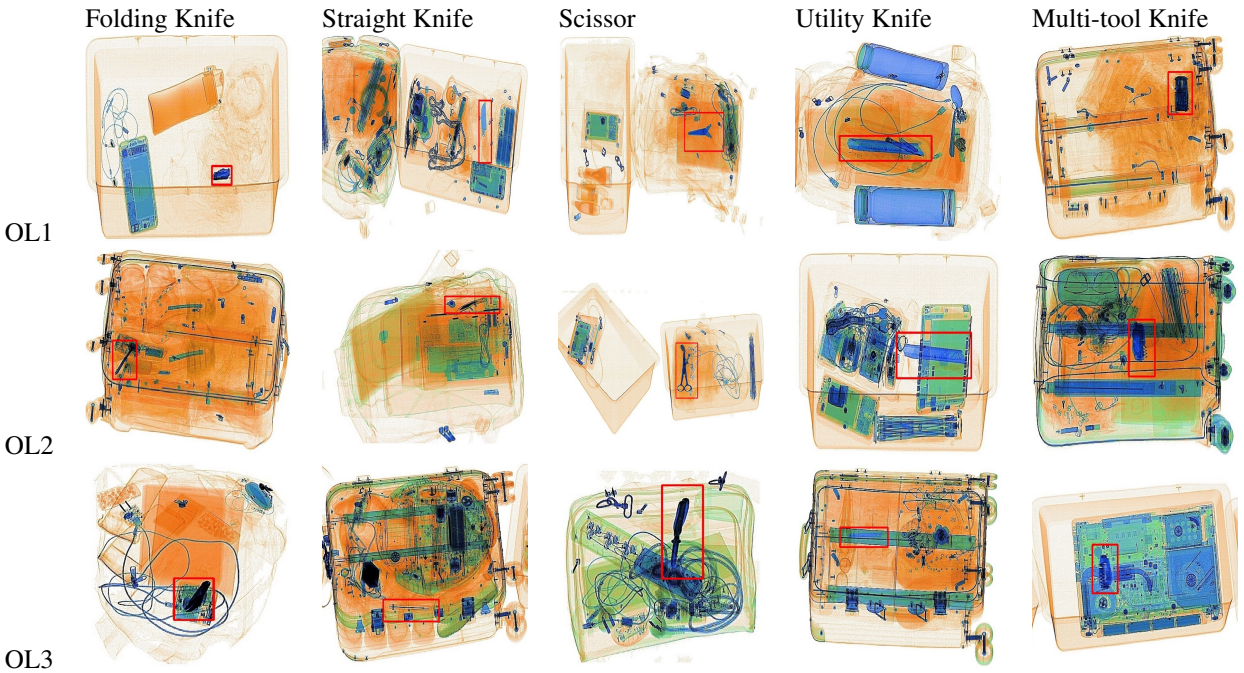

Table 2: Performance comparison between some famous detectors. FO, ST, SC, UT and MU represent Folding Knife, Straight Knife, Scissor, Utility Knife and Multi-tool Knife, respectively.

Method	mAP	Category				
		FO	ST	SC	UT	MU
SSD	70.89	76.91	35.02	93.41	65.87	83.27
YOLOv3	78.21	92.53	36.02	97.34	70.81	**94.37**
FCOS	82.02	86.41	68.47	90.22	78.39	86.60
Ours	**86.57**	**94.88**	**69.23**	**98.22**	**82.33**	88.20

Conclusion

In this paper, we investigate the task of using target detection in x-ray security screening to assist in hazardous materials detection. Our proposed method is able to adapt to hazardous materials detection in many different x-ray scenarios, and in particular, our proposed edge enhancement module is well suited to edge enhancement for problems such as missing textures in x-ray images, thus allowing the detector to perform better classification and localization based on edge features. Our experiments also finally and fully demonstrate the validation of the proposed method in this paper.

References

Akcay, S., and Breckon, T. P. 2020. Towards automatic threat detection: A survey of advances of deep learning within x-ray security imaging. *CoRR* abs/2001.01293.

AYDIN, I.; KARAKOSE, M.; and AKIN, E. 2018. A new approach for baggage inspection by using deep convolutional neural networks. 1–6.

Bochkovskiy, A.; Wang, C.-Y.; and Liao, H.-Y. M. 2020. Yolov4: Optimal speed and accuracy of object detection. *ArXiv* abs/2004.10934.

Cai, Y.; Du, D.; Zhang, L.; Wen, L.; Wang, W.; Wu, Y.; and Lyu, S. 2019. Guided attention network for object detection and counting on drones. *CoRR* abs/1909.11307.

Cui, Y., and Oztan, B. 2019. Automated firearms detection in cargo x-ray images using retinanet.

Dhiraj, and Jain, D. K. 2019. An evaluation of deep learning based object detection strategies for threat object detection in baggage security imagery. *Pattern Recognit. Lett.* 120:112–119.

Everingham, M.; Gool, L. V.; Williams, C. K. I.; Winn, J. M.; and Zisserman, A. 2009. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision* 88:303–338.

Felzenszwalb, P. F.; Girshick, R. B.; and McAllester, D. A. 2010. Cascade object detection with deformable part models. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 2241–2248.

Fu, C.-Y.; Liu, W.; Ranga, A.; Tyagi, A.; and Berg, A. C. 2017. Dssd : Deconvolutional single shot detector. *ArXiv* abs/1701.06659.

Ge, Z.; Liu, S.; Wang, F.; Li, Z.; and Sun, J. 2021. Yolox: Exceeding yolo series in 2021. *ArXiv* abs/2107.08430.

Ghiasi, G.; Lin, T.-Y.; Pang, R.; and Le, Q. V. 2019. Nas-fpn: Learning scalable feature pyramid architecture for object detection. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* 7029–7038.

Ghiasi, G.; Cui, Y.; Srinivas, A.; Qian, R.; Lin, T.-Y.; Cubuk, E. D.; Le, Q. V.; and Zoph, B. 2021. Simple copy-paste is a strong data augmentation method for instance segmentation. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* 2917–2927.

Girshick, R. B.; Donahue, J.; Darrell, T.; and Malik, J. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. *2014 IEEE Conference on Computer Vision and Pattern Recognition* 580–587.

Hu, J.; Shen, L.; Albanie, S.; Sun, G.; and Wu, E. 2020. Squeeze-and-excitation networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42:2011–2023.

Ji, R.; Du, D.; Zhang, L.; Wen, L.; Wu, Y.; Zhao, C.; Huang, F.; and Lyu, S. 2019a. Learning semantic neural tree for human parsing. *CoRR* abs/1912.09622.

Ji, R.; Wen, L.; Zhang, L.; Du, D.; Wu, Y.; Zhao, C.; Liu, X.; and Huang, F. 2019b. Attention convolutional binary neural tree for fine-grained visual categorization. *CoRR* abs/1909.11378.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. *Communications of the ACM* 60:84 – 90.

Li, C.; Du, D.; Zhang, L.; Wen, L.; Luo, T.; Wu, Y.; and Zhu, P. 2020. Spatial attention pyramid network for unsupervised domain adaptation. *CoRR* abs/2003.12979.

Liang, K. J.; Heilmann, G.; Gregory, C.; Diallo, S. O.; Carlson, D.; Spell, G.; Sigman, J. B.; Roe, K.; and Carin, L. 2018. Automatic threat recognition of prohibited items at aviation checkpoint with x-ray imaging: a deep learning approach.

Lim, J.-S.; Astrid, M.; Yoon, H.; and Lee, S.-I. 2021. Small object detection using context and attention. *2021 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)* 181–186.

Lin, T.-Y.; Dollár, P.; Girshick, R. B.; He, K.; Hariharan, B.; and Belongie, S. J. 2017. Feature pyramid networks for object detection. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 936–944.

Lin, T.-Y.; Goyal, P.; Girshick, R. B.; He, K.; and Dollár, P. 2020. Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42:318–327.

Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S. E.; Fu, C.-Y.; and Berg, A. C. 2016. Ssd: Single shot multibox detector. In *ECCV*.

Liu, S.; Qi, L.; Qin, H.; Shi, J.; and Jia, J. 2018. Path aggregation network for instance segmentation. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* 8759–8768.

Liu, Z.; Hu, H.; Lin, Y.; Yao, Z.; Xie, Z.; Wei, Y.; Ning, J.; Cao, Y.; Zhang, Z.; Dong, L.; Wei, F.; and Guo, B. 2022. Swin transformer v2: Scaling up capacity and resolution. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* 11999–12009.

Mery, D.; Riffo, V.; Zscherpel, U.; Mondragón, G.; Lillo, I.; Zuccar, I.; Lobel, H.; and Carrasco, M. 2015. Gdxray: The database of x-ray images for nondestructive testing. *Journal of Nondestructive Evaluation* 34:1–12.

Miao, C.; Xie, L.; Wan, F.; Su, C.; Liu, H.; Jiao, J.; and Ye, Q. 2019. Sixray: A large-scale security inspection x-ray benchmark for prohibited item discovery in overlapping images. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* 2114–2123.

Peng, Z.; Huang, W.; Gu, S.; Xie, L.; Wang, Y.; Jiao, J.; and Ye, Q. 2021. Conformer: Local features coupling global representations for visual recognition. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)* 357–366.

Redmon, J., and Farhadi, A. 2017. Yolo9000: Better, faster, stronger. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 6517–6525.

Redmon, J.; Divvala, S. K.; Girshick, R. B.; and Farhadi, A. 2016. You only look once: Unified, real-time object detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 779–788.

Ren, S.; He, K.; Girshick, R. B.; and Sun, J. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39:1137–1149.

Rezatofighi, S. H.; Tsoi, N.; Gwak, J.; Sadeghian, A.; Reid, I. D.; and Savarese, S. 2019. Generalized intersection over union: A metric and a loss for bounding box regression. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* 658–666.

Sermanet, P.; Eigen, D.; Zhang, X.; Mathieu, M.; Fergus, R.; and LeCun, Y. 2014. Overfeat: Integrated recognition, localization and detection using convolutional networks. *CoRR* abs/1312.6229.

Tian, Z.; Shen, C.; Chen, H.; and He, T. 2019. FCOS: fully convolutional one-stage object detection. *CoRR* abs/1904.01355.

Uijlings, J. R. R.; van de Sande, K. E. A.; Gevers, T.; and Smeulders, A. W. M. 2013. Selective search for object recognition. *International Journal of Computer Vision* 104:154–171.

Wang, X.; Girshick, R. B.; Gupta, A. K.; and He, K. 2018. Non-local neural networks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* 7794–7803.

Wei, Y.; Tao, R.; Wu, Z.; Ma, Y.; Zhang, L.; and Liu, X. 2020. Occluded prohibited items detection: An x-ray security inspection benchmark and de-occlusion attention module. *Proceedings of the 28th ACM International Conference on Multimedia*.

Woo, S.; Park, J.; Lee, J.-Y.; and Kweon, I.-S. 2018. Cbam: Convolutional block attention module. In *ECCV*.

Zhang, H.; Cissé, M.; Dauphin, Y.; and Lopez-Paz, D. 2018. mixup: Beyond empirical risk minimization. *ArXiv* abs/1710.09412.