International Journal of Computer Vision
https://doi.org/10.1007/s11263-023-01855-1

PIDray: A Large-Scale X-ray Benchmark for Real-World Prohibited Item Detection

Libo Zhang[1,2] · Lutao Jiang[1,2] · Ruyi Ji[3] · Heng Fan[4]

Received: 7 October 2022 / Accepted: 17 July 2023

Abstract

Automatic security inspection relying on computer vision technology is a challenging task in real-world scenarios due to many factors, such as intra-class variance, class imbalance, and occlusion. Most previous methods rarely touch the cases where the prohibited items are deliberately hidden in messy objects because of the scarcity of large-scale datasets, hindering their applications. To address this issue and facilitate related research, we present a large-scale dataset, named **PIDray**, which covers various cases in real-world scenarios for prohibited item detection, especially for deliberately hidden items. In specific, PIDray collects 124, 486 X-ray images for 12 categories of prohibited items, and each image is manually annotated with careful inspection, which characterizes it, to our best knowledge, with the largest volume and varieties of annotated images with prohibited items to date. Meanwhile, we propose a general divide-and-conquer pipeline to develop baseline algorithms on PIDray. Specifically, we adopt the tree-like structure to suppress the influence of the long-tailed issue in the PIDray dataset, where the first course-grained node is tasked with the binary classification to alleviate the influence of head category, while the subsequent fine-grained node is dedicated to the specific tasks of the tail categories. Based on this simple yet effective scheme, we offer strong task-specific baselines across object detection, instance segmentation, and multi-label classification tasks and verify the generalization ability on common datasets (*e.g.*, COCO and PASCAL VOC). Extensive experiments on PIDray demonstrate that the proposed method performs favorably against current state-of-the-art methods, especially for deliberately hidden items. Our benchmark and codes are available at https://github.com/lutao2021/PIDray.

Keywords Prohibited item dataset · Object detection · Instance segmentation · Multi-label classification

1 Introduction

Security inspection is tasked with checking packages against specific criteria and reveals any potential risks to ensure public safety, which is widely applied in real-world scenarios, such as public transportation and sensitive departments. In recent years, a set of surveys provide an in-depth review of developments in this field (Akcay & Breckon, 2022; Mery et al., 2020; Velayudhan et al., 2022). In practice, there is an ever-increasing demand for inspectors to monitor the scanned X-ray images generated by the security inspection machine to specify potentially prohibited items, such as guns, ammunition, explosives, corrosive substances, and toxic and radioactive substances. But unfortunately, it is highly challenging for inspectors to localize prohibited items hidden in messy objects accurately and efficiently, which poses a great threat to safety.

Deep learning technologies have sparked tremendous progress in the computer vision community (Ren et al., 2015;

Communicated by D. Scharstein.

✉ Ruyi Ji
jrylovezd@gmail.com

Libo Zhang
libo@iscas.ac.cn

Lutao Jiang
lutao2021@iscas.ac.cn

Heng Fan
heng.fan@unt.edu

[1] Institute of Software Chinese Academy of Sciences, Beijing, China

[2] University of Chinese Academy of Sciences, Beijing, China

[3] National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China

[4] Department of Computer Science and Engineering, University of North Texas, Denton, USA

Published online: 04 August 2023

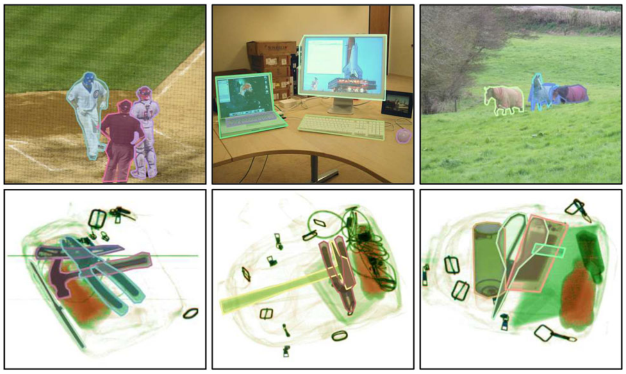

Fig. 1 Comparisons between the natural image (the 1st row) and X-ray image (the 2nd row)

Liu et al., 2016; Tian et al., 2019; Ji et al., 2020b, a; Li et al., 2020; Cai et al., 2020), which makes it possible to inspect prohibited items automatically. The security inspectors demand to quickly identify the locations and categories of prohibited items relying on computer vision technology. Most previous object detection algorithms are well-designed to detect objects in natural images, which are not optimal for detection in X-ray images due to the following factors. Firstly, X-rays have strong penetrating power, and different materials in the object absorb X-rays to different degrees, resulting in different colors. Secondly, the contours of the occluder and the occluded objects in the X-ray are mixed together. As shown in Fig. 1, compared with natural images, X-ray images present a quite different appearance and edges of objects and background, which brings new challenges in appearance modeling for X-ray detection. To advance the developments of prohibited item detection in X-ray images, some recent attempts devote to constructing security inspection benchmarks (Mery et al., 2015; Akcay & Breckon, 2017; Akcay et al., 2018; Miao et al., 2019; Wei et al., 2020). However, most of them fail to meet the requirements in real-world applications for three reasons. (1) Existing datasets (such as GDXray) are characterized by small volume and very few categories of prohibited items (*e.g.*, *knife*, *gun*, and *scissors*). For example, some common prohibited items such as *powerbank*, *lighter*, and *sprayer* are not involved in the previous datasets. (2) Some real-world scenarios require high-level security based on accurate predictions of masks and categories of prohibited items. Most existing benchmarks (for example SIXray and OPIXray) only offer image-level or bounding-box-level annotations, which are not optimal configurations in such scenarios. (3) Detecting prohibited items hidden in messy objects is one of the most significant challenges in security inspection. Unfortunately, few studies have been developed toward this goal due to the lack of comprehensive datasets covering such cases. These challenges urgently require a large-scale prohibited item benchmark and an efficient and effective method.

On the other hand, we observe that the models trained on the dataset, the majority of which are with prohibited items, are error-prone when processing the samples without any prohibited items. We argue that this issue arises from the fact that mainstream training schemes exclude all images without any bounding box by default during the pre-processing stage. Even though no significant influence on general datasets (*e.g.*, COCO (Lin et al., 2014) and PASCAL VOC (Everingham et al., 2010)) where annotated samples account for the majority, this arrangement of dataset incurs a dilemma in security inspection, as the images with prohibited items are just special cases in the real scenario. Such cases reflect the large gap between artificial datasets and real scenarios. Therefore, for a more stable and robust model in security inspection, the construction of the dataset should be in line with the real scenarios as far as possible.

To this end, we present a large-scale prohibited item detection dataset (PIDray) for real-world applications. The PIDray dataset covers 12 categories of prohibited items in X-ray images. From the exemplars with annotations in Fig. 2, we can observe that each image contains at least one prohibited item with both the bounding box and mask annotations. Notably, for the fine-grained investigation, the test set is well-divided into three parts, *i.e.*, *easy*, *hard*, and *hidden*. Particularly, the *hidden* subset focuses on the prohibited items deliberately hidden in messy objects (*e.g.*, changing the item shape by wrapping wires). To the best of our knowledge, it is a dataset characterized with the largest volume and varieties of annotated images with prohibited items to date for this domain.

Based on the observation that images without prohibited items account for the majority of the proposed dataset, which characterizes the dataset with long-tailed distribution, we propose a divide-and-conquer pipeline, which adopts the tree-like structure to suppress the influence of samples without prohibited items in a course-to-fine manner. Specifically, the sample first passes through the first coarse-grained node to determine whether it contains the prohibited item or not. If true, the sample is fed to the subsequent fine-grained node for the task-specific operations (*e.g.*, detection or segmentation). If not, that means this is a sample without any prohibited items. The key insight of our method is that the distribution of the proposed dataset enables us to cast such a task as a multi-task learning problem. We perform the binary classification in the first node to balance the head and tail categories in a course-grained manner, and then perform task-specific operations in the later node in a fine-grained manner.

For object detection and instance segmentation tasks, we utilize such a pipeline to construct a strong baseline on the top of two- or one-stage detectors like Cascade Mask R-CNN (Cai & Vasconcelos, 2019), where we also contribute to the FPN structure with the dense attention module. Concretely, we first use both the spatial- and channel-wise attention

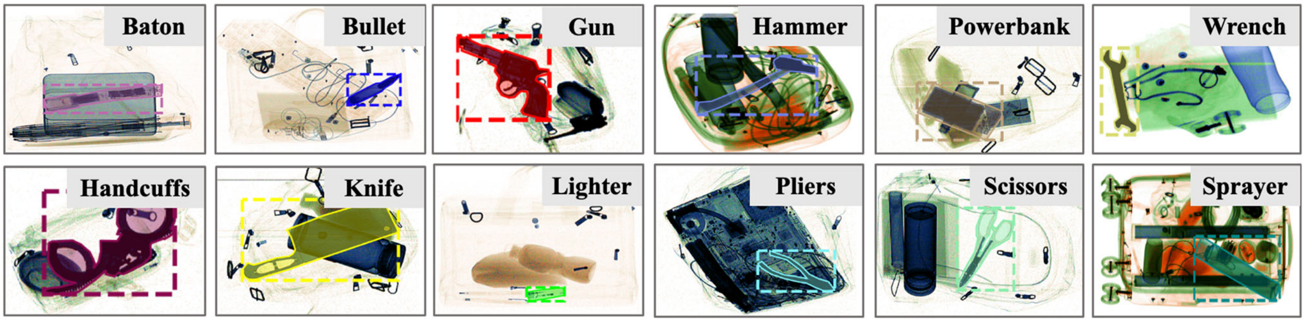

Fig. 2 Example images in the PIDray dataset with 12 categories of prohibited items. Each image is provided with image-level and instance-level annotation. For clarity, we show one category per image

mechanisms to exploit discriminative features, which is effective to locate the deliberately prohibited items hidden in messy objects. Then we design the dependency refinement module to explore the long-range dependencies within the feature map. Extensive experiments on the proposed dataset show that our method performs favorably against the state-of-the-art methods.

Meanwhile, to fully unleash the potential of the proposed dataset, we establish a multi-label classification task for this benchmark. And we extend the divide-and-conquer pipeline to this domain to alleviate the issue of long-tailed distribution. Specifically, the first coarse-grained node is tasked with binary classification, filtering out the head category without prohibited items. After that, the fine-grained node is dedicated to the multi-label classification of the tail categories with prohibited items. The experiment performance demonstrates that our design is a simple yet effective scheme for the multi-label classification task on the PIDray dataset.

To sum up, the main contributions of this work can be summarized into the following five folds.

- Towards the prohibited item detection in real-world scenarios, we present a large-scale benchmark, *i.e.*, PIDray, formed by 124, 486 images in total. To the best of our knowledge, it is an X-ray detection dataset with the largest volume and varieties of annotated images with prohibited items to date. Meanwhile, it is the first benchmark dedicated to cases where prohibited items are deliberately hidden in messy objects.
- We provide various tasks besides object detection on the proposed PIDray to fully unleash its potential in real-world applications, including segmentation and multi-label classification.
- We propose the divide-and-conquer pipeline to address the issue of long-tailed distribution in the PIDray dataset, which adopts the tree-like structure to suppress the influence of samples without prohibited items in a course-to-fine manner.
- With our novel divide-and-conquer pipeline, we deliver strong task-specific baselines across object detection, instance segmentation, and multi-label classification tasks on the PIDray dataset and verify its generalization ability on common datasets (*e.g.*, COCO and PASCAL VOC).
- Extensive experiments, carried out on the PIDray dataset and general dataset, verify the effectiveness of the proposed methods compared to the state-of-the-art methods.

This paper extends an early conference version in Wang et al. (2021). The main new contributions are as follows. (1) We enlarge PIDray by introducing 76,809 new X-ray images without prohibited items to bridge the gap between the artifact dataset and natural scenarios. (2) We enrich the applications of PIDray by introducing a new task of multi-label classification, which further unleashes the potential of PIDray. (3) We propose a novel divide-and-conquer pipeline for developing strong task-specific baselines on PIDray to facilitate future research. (4) More thorough experiments are conducted on PIDray with in-depth analysis to show the effectiveness of our approach. (5) Besides the experiments on PIDray, we further verify the generalization ability of our pipeline on common benchmarks (*e.g.*, COCO, PASCAL VOC).

The remainder of this paper is organized as follows. Section 2 briefly reviews research directions relevant to our method. In Sect. 3, we describe the construction of the PIDray dataset in detail. In Sects. 4 and 5, the task-specific strong baselines are realized under the guidance of the divide-and-conquer pipeline. In Sect. 6, the extensive experimental results are reported and analyzed, including comparing the proposed method and the state-of-the-art approaches, validation of generalization ability on the general datasets, and the comprehensive analysis of the ablation studies. Finally, Sect. 7 draws the conclusions of the proposed method.

2 Related Work

This section reviews six major research directions closely related to our work, *i.e.*, prohibited item benchmarks, object detection, the attention mechanism, multi-label classification, multi-scale feature fusion, and long-tailed distribution issues.

2.1 Prohibited Items Benchmarks

Due to discrepancies of penetrating capability, different materials tend to present various colors under X-ray. Such property incurs more challenges in cases where objects are overlapped. Moreover, like natural images, X-ray images are featured of notorious characteristics, *e.g.*, intra-class variances and distribution imbalance. Recently, a few datasets have been collected to advance prohibited item detection investigations. To be concrete, Mery et al. (2015) collects the GDXray dataset for nondestructive testing. GDXray is formed by three prohibited items: gun, shuriken and razor blade. Without complex background and overlap, it is easy to recognize or detect objects in this dataset. Differing from GDXray, Dbf$_6$ (Akcay & Breckon, 2017), Dbf$_3$ (Akcay et al., 2018) and OPIXray (Wei et al., 2020) cover complicated background and overlapping-data, but unfortunately, the volumes of images and prohibited items are still insufficient. Recently, Liu et al. (2019a) releases a dataset containing $32,253$ X-ray images, of which $12,683$ images include prohibited items. This dataset has 6 types of items, but none are strictly prohibited, such as mobile phones, umbrellas, computers, and keys. Miao et al. (2019) provides a large-scale security inspection dataset called SIXray, which covers $1,059,231$ X-ray images with image-level annotation. However, the proportion of images containing prohibited items is very small in the dataset (*i.e.*, only 0.84%). In addition, there are 6 categories of prohibited items, but only 5 categories are annotated. Unlike the aforementioned datasets, we propose a new large-scale security inspection benchmark that contains over 47K images with prohibited items and 12 categories of prohibited items with pixel-level annotation. Towards real-world application, we focus on detecting deliberately hidden prohibited items.

2.2 Object Detection

Object detection is a long-standing problem in the computer vision community. Generally speaking, modern object detectors fall into two groups: two-stage and one-stage detectors.

2.2.1 Two-stage Detectors

R-CNN (Girshick et al., 2014) exemplifies the first research line and proves that CNN can dramatically improve detection performance. However, it is time-consuming since each regional proposal is computed individually in this pipeline. To close this gap, Fast-RCNN (Girshick, 2015) utilizes the ROI pooling layer to extract fixed-size features for each proposal from the feature map of the full image. Therein, the selective search, which is utilized to generate proposal regions, becomes a major bottleneck. The follower Faster RCNN (Ren et al., 2015) introduces an efficient way to replace selective search, which designs the RPN network and derives numerous variants. For example, FPN (Lin et al., 2017a) assembles low-resolution features with high-resolution features through a top-down pathway and lateral connections, which greatly improves the detection ability at different sizes. Mask R-CNN (He et al., 2017) attaches a mask branch to the Faster-RCNN (Ren et al., 2015) structure to improve detection performance via a multi-task learning scheme. Cascade R-CNN (Cai & Vasconcelos, 2018) introduces the classic cascade structure into Faster R-CNN (Ren et al., 2015) framework, which consistently improves detection accuracy in a cascade manner. Libra R-CNN (Pang et al., 219) develops a simple yet effective strategy to alleviate the issue of data imbalance in the training process.

2.2.2 One-stage Detectors

OverFeat (Sermanet et al., 2013) is one of the seminal methods relying on deep learning among one-stage detectors. Later, tremendous efforts have been made to develop one-stage object detectors, like SSD (Liu et al., 2016), DSSD (Fu et al., 2017), and YOLO series (Redmon et al., 2016; Redmon & Farhadi, 2017, 2018). Notably, even though the YOLO series achieve high speed, the localization errors exist more than other two-stage detectors, and the performance on small object remains unsatisfactory. SSD combines the multi-scale feature maps to predict the fixed number of bounding boxes, then uses NMS method to reduce region redundancy, achieving a decent trade-off between both higher speed and accuracy. Yet, the multi-scale features involved in the SSD are independent of each other. Therefore, RetinaNet (Lin et al., 2017b) attempts to fuse the different scale feature maps and proposes the focal loss, making the one-stage detector rival the two-stage detector. Besides, these anchor-based methods usually encounter the following two problems: 1) The number of anchor boxes needs to be large. 2) The anchor-based algorithm introduces a number of hyper-parameters and design choices, which incurs complex network architecture. Recently, numerous anchor-free approaches have sparked considerable research attention by formulating the objects as key points, such as CornerNet (Law & Deng, 2018), CenterNet (Duan et al., 2019), and FCOS (Tian et al., 2019). These methods present a simplified detection scheme, getting rid of the limitation for anchors. Besides, DETR (Carion et al., 2020) pioneers the approach which feeds the serialized features into the Transformer architecture for prediction. DDOD (Chen et al., 2021) investigates some conjunctions of the training pipeline and proposes a simple yet effective disentanglement method to boost performance.

Notably, there are numerous works has well investigated a set of original object detectors in the security inspection field. For example, method (Saavedra et al., 2021) proposes to enrich the GDXray dataset via the GAN paradigm, which greatly compensates for the extremely scarce samples and evaluates the performance on a wide range of existing frameworks, including SSD300, YOLO v2, YOLO v3, and RetinaNet. As such, approach (Akcay & Breckon, 2017) validates the RCNN, R-FCN, YOLOv2, *etc*, on their proposed dataset. Unlike them, we not only evaluate a diverse set of detection frameworks but also propose a novel detection framework for security inspection.

Table 1 Comparison of the dataset statistics with existing X-ray benchmarks

Dataset	Year	Classes	Images		Annotations			Type	Channel	Resolution	Scene	Task	Avail
			Total	Prohibited	Image	Bbox	Mask						
GDXray [48]	2015	3	8, 150	8, 150	✓	✓	–	Real	Grayscale	From 612 × 452 to 2688 × 2208	–	C/O	✓
Dbf$_6$ [2]	2017	6	11, 627	11, 627	✓	✓	–	Real	RGB	–	–	C/O	✗
Dbf$_3$ [3]	2018	3	7, 603	7, 603	✓	✓	–	Real	RGB	–	–	C/O	✗
Liu *et al.* [42]	2019	6	32, 253	12, 683	✓	✓	–	Real	RGB	–	S	C/O	✗
SIXray [50]	2019	6	**1, 059, 231**	8, 929	✓		–	Real	RGB	657 × 578	S	M	✓
OPIXray [73]	2020	5	8, 885	8, 885	✓	✓	–	Synthetic	RGB	1225 × 954	A	C/O	✓
Our PIDray	2022	**12**	124, 486	**47, 677**	✓	✓	✓	Real	RGB	632 × 555	S/A/R	C/O/I/M	✓

The significance of the bold means the largest number among all the datasets involved in the table. “Total” and “Prohibited” indicate the number of total images and the images containing prohibited items in the dataset, respectively. **C**, **O**, **I**, and **M** represent Classification, Object Detection, Instance Segmentation, and Multi-label classification respectively. The column ’Channel’ is the number of channels of the image. The column ’Scene’ is the data sources. **S**, **A**, and **R** represent Subway, Airport, and Railway Station, respectively. The resolution is the average width and height of X-Ray images

2.3 Attention Mechanism

Recently, inspired by the human perceptual vision system, the attention mechanism has been successfully applied to a wide variety of visual understanding tasks, such as image recognition, image captioning, visual question answering, *etc*. The core of the attention mechanism is to emphasize the relevant parts while suppressing irrelevant ones. Regarding the stable and discriminative representation, considerable developments over it have been investigated in the literature. Specifically, as a pioneering work, RAM (Mnih et al., 2014) applies Recurrent Neural Network to locate informative regions recursively. It inspires a number of follow-ups in CNN networks. Later, SENet (Hu et al., 2018) proposes the Squeeze-and-Excitation module to re-calibrate the dependence from channel perspective. Its successors like GSoP-Net (Gao et al., 2019) and FcaNet (Qin et al., 2021) try to improve the squeeze module. And the follow-up method ECA-Net (Wang et al., 2020) pays more attention to the excitation module. However, these methods only focus on the channel aspect in the CNN network, neglecting great potential from the space perspective. The method CBAM (Woo et al., 2018) jointly explores the inter-channel and inter-spatial relationships between features. Nevertheless, it struggles from the long-range dependency on the image. Non-Local network (Wang et al., 2018) captures the long-range dependency and calculates the weighted representations for a certain position in the feature map with the consideration of other positions’ contributions. Yet it requires heavy computation cost. To solve this issue, Cao et al. (2019) design a lightweight module in conjunction with the simplified Non-Local pipeline. CCNet (Huang et al., 2019) presents the Recurrent Criss Cross Attention (RCCA) module, simplifying the global self-attention in a Non-Local network.

2.4 Multi-Label Classification

Multi-label classification has recently drawn increasing research attention. There are two primary paradigms for multi-label classification: methods based on attention mechanisms and those dedicated to exploring the specialized loss functions.

The attention mechanism underpins the first paradigm. CSRA (Zhu & Wu, 2021) proposes a class-specific residual attention module that adaptively allocates the spatial attention weight to each category. MCAR (Gao & Zhou, 2021) presents an attention mechanism to capture the most informative local region and then feeds the scaled input to the backbone again. Moreover, numerous methods introduce the Transformer (Vaswani et al., 2017) structure. Despite considerable efforts to apply the attention mechanism, their ability to capture long-range dependency is inferior to the transformer-like structure. Specifically, Query2Label (Liu et al., 2021) introduces the transformer structure to encode and decode category-related features in the feature map. But the computational burden is heavy. Therefore, ML-Decoder (Ridnik et al., 2021b) simplifies the structure. It designs a novel decoder structure and generalizes to the scenarios with a large number of categories. Another paradigm methods seek to improve performance via the specialized loss functions. For example, Asymmetric Loss (Ridnik et al., 2021a) proposes to dynamically reduce the effect of negative samples and even discard samples suspected to be mislabeled. The previous methods suffer heavily from partial annotation problems in existing large-scale datasets. Class-aware Selective Loss (Ben-Baruch et al., 2021) proposes a method to estimate the class distribution for the partial labeling problem and uses a dedicated asymmetric loss in the later training to put more emphasis on the contribution of labeled data rather than unlabeled data.

2.5 Multi-Scale Feature Fusion

Multi-scale feature fusion arises naturally in various tasks of the computer vision field due to the fact that objects usually show different sizes in the image. Numerous works attempt to utilize the intermediate feature maps of the backbone to improve detection accuracy. For instance, FPN (Lin et al., 2017a) pioneers the approach to fuse features from multiple layers in the CNN-based structure. It establishes a top-down path to provide the informative features to the lower-level feature maps and jointly performs probing at different scales. Later, much effort focuses on how to improve fusion capabilities between different layers in a more efficient manner, *e.g.*, Gong et al. (2021); Pang et al. (219); Liu et al. (2018, 2019b); Ghiasi et al. (2019); Chen et al. (2020). Concretely, to enhance the information propagation from upper to lower levels, Gong et al. (2021) propose a fusion factor that plays a key role in fusing feature maps at different scales. Liu et al. (2019b) present an adaptive spatial fusion method that greatly improves detection accuracy with little computation overhead. Recently, some more complex frameworks for feature fusion (Ghiasi et al., 2019; Chen et al., 2020) have been explored in the literature. NAS-FPN (Ghiasi et al., 2019) delivers Neural Architecture Search (NAS) to optimize the best path of FPN. Differing from NAS-FPN, FPG (Chen et al., 2020) proposes a deep multi-pathway feature pyramid structure to improve the generalization ability. Unlike the original FPN structure, CARAFE (Wang et al., 2019a) offers a lightweight learnable module to perform up-sampling. Li et al. (2022b) seek to reconstruct a feature pyramid for the backbones without a hierarchical structure.

As many previous works (Lin et al., 2017a; Liu et al., 2018) show the importance of multi-scale feature fusion, we argue it is the key to solving the problem of prohibited item detection. In X-ray images, many important details of objects are missing, such as texture and appearance information. Moreover, the contours of objects overlap, which also brings great challenges to detection. Multi-scale feature fusion jointly considers the low-level layers with rich detail information and the high-level layers with rich semantic information. Therefore, we propose a dense attention module to capture the relations between feature maps across different stages at inter-channel and inter-pixel positions.

2.6 Long-Tailed Distribution Issue

The long-tailed distribution issue refers to a small proportion of the categories that accounts for a large proportion of the overall dataset, which makes the network struggle to capture discriminative information from the categories in the tail. To this end, tremendous research efforts have been made in the literature, *i.e.*, class re-balancing (Kang et al., 2019; Wang et al., 2019b; Cui et al., 2019), information augmentation (Yin et al., 2019; Li et al., 2021), module improvement (Zhou et al., 2020; Xiang et al., 2020; Cai et al., 2021).

Methods in the first research line focus on the class rebalance strategy to alleviate long-tailed distribution issue. For instance, Kang et al. (2019) decouple the learning process into *representation learning* and *classification* and investigate the influence of different class balancing strategies. DCL (Wang et al., 2019b) proposes to recognize the tail categories by sampling them more frequently in the later stage of training. Class-Balanced Loss (Cui et al., 2019) develops a new re-weighting scheme according to the effective number of samples per class. These works try to solve this problem by making different samples have different influence weights, alleviating the problem that tail class samples fail to be focused on enough to some extent. The methods in the second stream rely on information augmentation to compensate for the lack of samples in the tail classes. For example, Feature Transfer Learning (Yin et al., 2019) proposes to leverage class information in the head to perform feature amplification on those in the tail. MetaSAug (Li et al., 2021) presents an approach based on implicit semantic data augmentation (ISDA) (Wang et al., 2019c) and adopts meta learning to learn to transform semantic orientations automatically. The methods in the last stream draw support from a module design perspective. Bbn (Zhou et al., 2020) proposes to divide the network into two separate branches, one branch applies naturally randomly and uniformly sample strategy for training, and the other one samples more tail classes. LFME (Xiang et al., 2020) proposes to divide the dataset into subsets with smaller "class longtailness" and optimizes multiple models individually. It applies knowledge distillation to learn a student model from multiple teacher models. However, these methods need to optimize the different parts of the network alternately. The proposed network structure is trained in an end-to-end fashion. It can be easily plugged into existing architectures, such as SSD, Faster RCNN, Cascade RCNN, *etc*, proving the flexibility of our pipeline in the security inspection scene.

3 Dataset PIDray

In this section, we turn to describe how to construct the PIDray dataset in detail, including the data collection, annotation, and statistical information.

3.1 Dataset Collection

The PIDray dataset is collected in various scenarios (*e.g.*, airports, subway stations, and railway stations), where we are allowed to set up a security inspection machine. For strong generalization, we deploy 3 security inspection machines provided by different manufacturers to collect X-ray data. Images generated by different machines usually present certain variances in the size and color of the objects and background. When the packages go through the security inspection machine, they are completely cut out under the guidance of blank parts of the image. In most cases, the X-ray image processes fixed height, while its width depends on the size of the package.

The details about the collection process are presented as follows: when the person is required for security inspection, we randomly put the pre-prepared prohibited items in his/her carry-on. Meanwhile, the rough location of the object is recorded, which guarantees that the subsequent annotation work can be carried out smoothly. PIDray dataset is involved with a total of 12 categories of prohibited items, namely, gun, knife, wrench, pliers, scissors, hammer, handcuffs, baton, sprayer, power bank, lighter and bullet. For the diversity, 2 $\sim$ 15 instances are ready for each kind of prohibited item. It takes more than six months to collect 124, 486 images for the PIDray dataset. Besides, Fig. 3 summarizes the distribution of tail categories in the dataset. It is notable that all images are saved in PNG format.

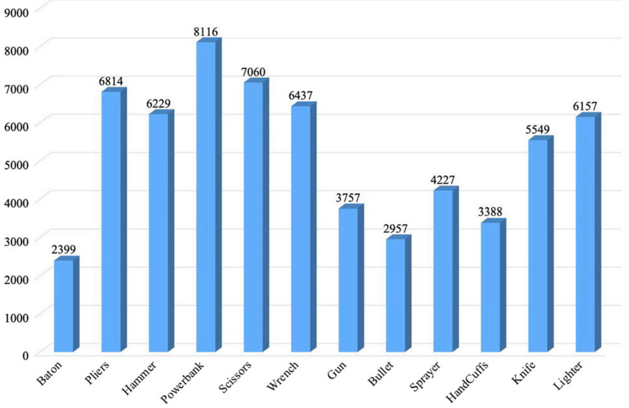

Fig. 3 Class distribution of the PIDray dataset. The blue bar represents the number of each class in the PIDray dataset

3.2 Data Annotation

For the annotation task, we recruited some interns, Masters and Ph.D. students, who are familiar with this research, as volunteers. Towards high-quality annotations for each image, we offer some training to recruited volunteers to identify prohibited items from X-ray images more quickly and accurately. After that, 5 volunteers are responsible for filtering out samples without any prohibited items from the dataset and annotating the image-level labels, which greatly facilitates the subsequent annotation work. For the fine-grained annotation, we organize over 10 volunteers to annotate our dataset with the tool named labelme.[1] In specific, we leverage a multi-step strategy to ensure high-quality annotation. Firstly, a group of volunteers who are familiar with the topic and an expert (*e.g.*, Ph.D. students working on related areas) participate in manually annotating each prohibited item in images. Then, a group of experts carefully inspect the initial annotations. If the annotation results are not unanimously agreed by experts, they will be returned to the labeling team for adjustment or refinement. Each image generally takes about 3 minutes to annotate, and each volunteer spends about 10 hours to annotate the image daily. During the annotation process, we label both the bounding box and the segmentation mask of each instance. Next, multiple rounds of double-check are performed to ensure minimum errors.

[1] http://labelme.csail.mit.edu/Release3.0/.

Table 2 Statistics of the PIDray dataset

Mode	Train	Test		
		Easy	Hard	Hidden
Count	76, 913	24, 758	9, 746	13, 069
Total	124, 486			

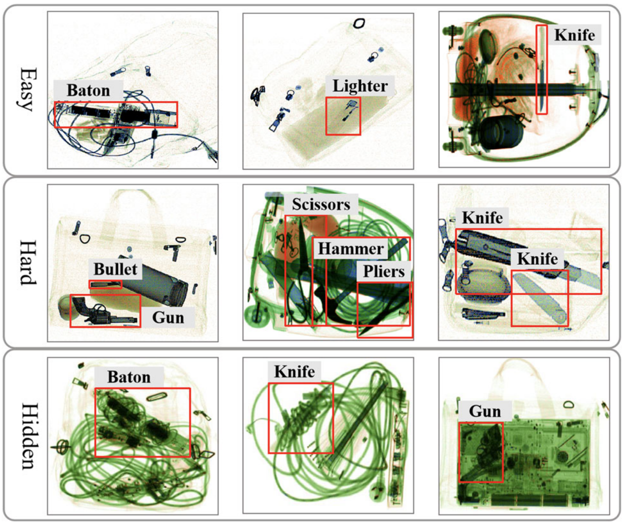

Fig. 4 Examples of test sets with different difficulty levels in the proposed PIDray dataset. From top to bottom, the degree of difficulty gradually increases

3.3 Data Statistics

To our best knowledge, the PIDray dataset is the largest X-ray prohibited item detection dataset until now. It covers 124, 486 images and 12 classes of prohibited items. As Table 2 shows, we divide those images into 76, 913 (roughly 60%) images for training and 47, 573 (remaining 40%) images for testing, respectively. Besides, according to the difficulty degree of prohibited item detection, we split the test set into three subsets, *i.e.*, easy, hard and hidden. In detail, the easy mode means that the image in the test set contains only one prohibited item. The hard mode indicates that the image in the test set contains more than one prohibited item. The hidden mode implies that the image in the test set contains deliberately hidden prohibited items. As shown in Fig. 4, we provide several examples in the test set with different difficulty levels.

4 Methodology of Object Detection and Instance Segmentation

For object detection and instance segmentation, we adopt the divide-and-conquer pipeline to alleviate the influence of the overwhelming samples without prohibited items on task-specific operations. As Fig. 5 illustrates, the input sample first passes through the node $\mathcal{R}_0$ to predict whether it contains prohibited items. The node $\mathcal{R}_0$ confines a confidence score ϕ in the interval [0, 1]. We define that when $\phi > 0.5$, it means the current sample contains prohibited items. Otherwise, it means just a normal sample. In our case, only samples with prohibited items need to be fed into the subsequent task-specific processing, which encourages the node $\mathcal{R}_1$ to focus more on samples with prohibited items. Meanwhile, we impose the constraint of BCE loss on the output of the node $\mathcal{R}_0$ to guide the gradient via back-propagation. Generally, the proportion of samples without prohibited items in the total batch size is uncertain. Hence, we multiply the original loss of the node $\mathcal{R}_1$ with the proportion of samples with prohibited items in the current batch size to avoid the collapse due to few samples going through the node $\mathcal{R}_1$, which assures the stability of training process.

4.1 Binary Classification on the Node of $\mathcal{R}_0$

Following the principle of divide-and-conquer, the course-grained node $\mathcal{R}_0$ (see the bottom left of Fig. 5) is tasked with extracting features from the multi-scale features $\mathcal{F}_1$-$\mathcal{F}_5$ to perform the binary classification, whose goal is to determine whether the input samples contain prohibited items. To fully unleash the potential of the multi-scale features, we first aggregate these features to obtain a fused feature $\mathcal{F}^{'}$. Generally, the previous methods usually compress the feature map into a one-dimensional feature via an average pooling operation. Unfortunately, this process tends to leak essential information for class prediction. Instead, we design a cross-attention module to facilitate the model to encode more informative features. Concretely, we take a learnable embedding as the query, each pixel on the feature map as the value, and each pixel with position embedding as the key. For the query, the global dependencies can be captured by Equ. (1)-(2).

$$\alpha_i = softmax\left(\frac{W_Q(Q)W_K(E_i)}{\sqrt{d}}\right) \tag{1}$$

$$h = \sum_{i=1}^{HW} \alpha_i W_V(F_i') \tag{2}$$

where $W_Q(\cdot)$, $W_K(\cdot)$ and $W_V(\cdot)$ are linear projections for query, key, and value, respectively, E refers to the spatial embedding obtained from the sum of $\mathcal{F}'$ and its position encoding (Carion et al., 2020), d is the dimension of the hidden layer, α_i is the score of each position, h denotes the specific head representation. Next, we feed the concatenation of the multi-head representation into a feed-forward neural network to obtain the final feature representation, which is expressed as Equ. (3).

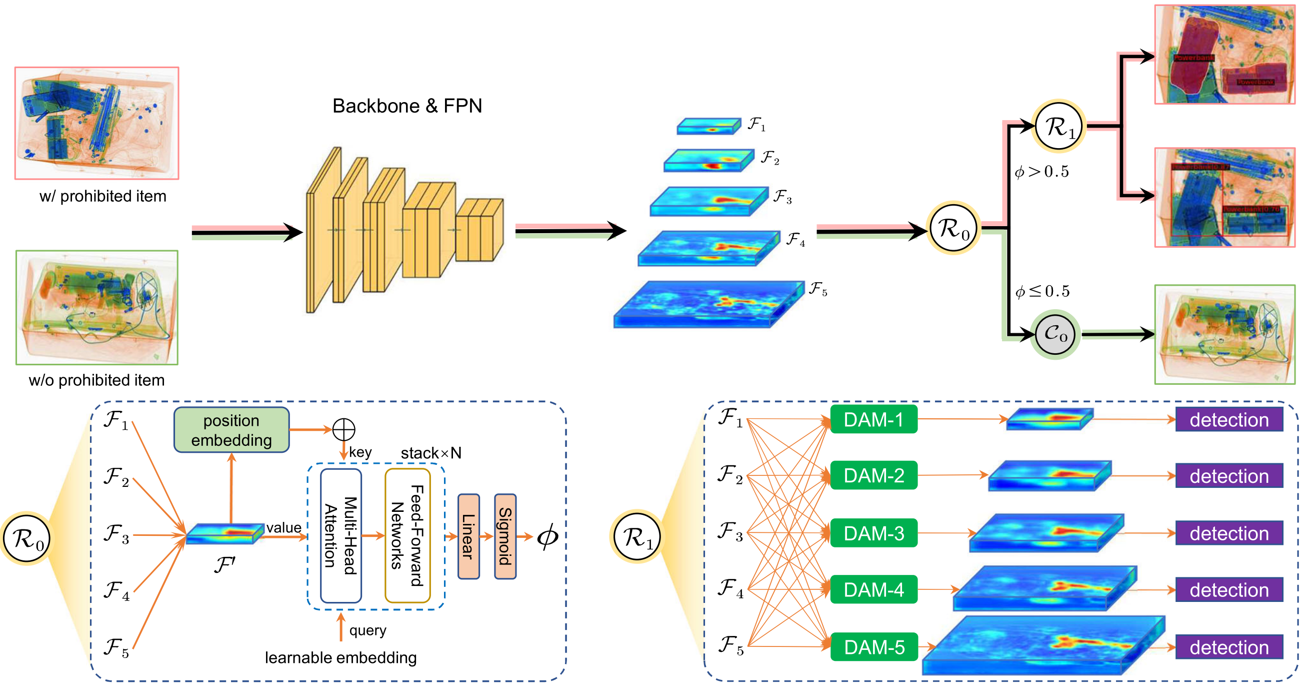

Fig. 5 The overall framework of the proposed method for the object detection/instance segmentation task. Specifically, coarse-grained node $\mathcal{R}_0$ is tasked with determining whether a sample contains prohibited items or not. The fine-grained node $\mathcal{R}_1$ focuses on task-specific improvements

$$V = FFN\left(W_H\left(\prod_{j=1}^{N} h_j\right)\right) \tag{3}$$

where N indicates the number of multiple heads, $\prod$ means the concatenation operation, $W_H(\cdot)$ is the linear projection for fusion of multi-head information, $FFN(\cdot)$ denotes a position-wise feed-forward network (Vaswani et al., 2017). Finally, we adopt a fully connected layer followed by a sigmoid layer as a classifier to project this feature vector V into a score ϕ.

4.2 Detection/Segmentation on the Node of $\mathcal{R}_1$

As discussed in the introduction section, the fine-grained node $\mathcal{R}_1$ focuses on the task-specific improvements, *i.e.*, contributions to the tasks of object detection and instance segmentation. In our design, we choose the two-stage object detection framework, *e.g.*, Cascade Mask R-CNN (Cai & Vasconcelos, 2019), to realize the node $\mathcal{R}_1$. Typically, the two-stage object detection frameworks often consist of components such as backbone, Feature Pyramid Network (FPN) (Lin et al., 2017a), and detection heads (Region Proposal Network (RPN) followed by the succinct heads). The detector head often works simultaneously on the five feature maps of different scales output by FPN to improve the performance of detection with different sizes. After the RPN provides proposals, the succinct heads (*e.g.*, a simple convolutional layer) are utilized on the pooled feature grid to predict bounding boxes and masks of instances. Nevertheless, the previous approaches usually adopt the feature pyramid structure to exploit multi-scale feature maps in the network, focusing on fusing features only in adjacent layers. The performance tends to be sub-optimal owing to the scale variation of objects in complex scenes.

To address the above issue, we design the dense attention module in our study. The proposed framework consists of five Dense Attention Modules (DAMs) (see the bottom right of Fig. 5), which are denoted as $\{DAM_1, DAM_2, \ldots, DAM_5\}$. Each DAM is fed with all multi-scale feature maps in the pyramid as input. Let the feature maps in the pyramid be $\{\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_5\}$. The workflow of the DAM_i is depicted in Fig. 6. Taking the $\mathcal{F}_i$ as an example, we first scale all the input feature maps to match the size of $\mathcal{F}_i$. We adopt the max pooling operation for down-sampling feature maps with sizes larger than $\mathcal{F}_i$, while the nearest interpolation for up-sampling those smaller than $\mathcal{F}_i$. Then the five feature maps are added and fed into the Channel-wise Attention Module and Spatial-wise Attention Module to obtain channel-wise and spatial-wise weights, respectively. In other words, we re-calibrate the importance of various feature maps from the channel and space perspectives. Subsequently, the multiplication operation emphasizes the attentive channel and spatial regions. After that, the sum of the channel and spatial features is fed into the dependency refinement module. Finally, we add this result directly to $\mathcal{F}_i$ via a lateral skip connec-

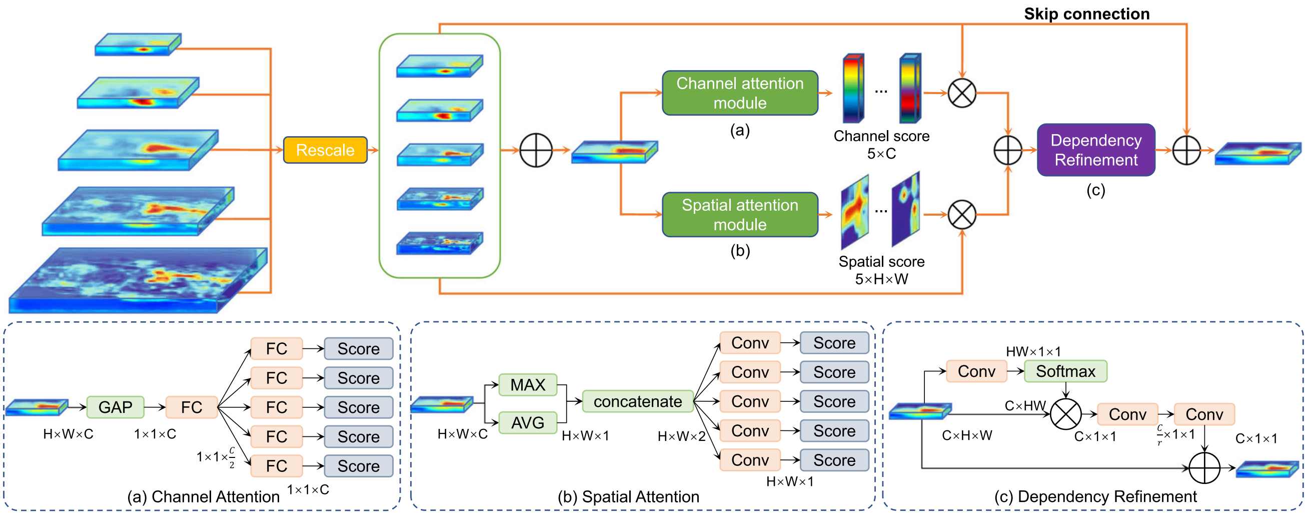

Fig. 6 The structure of Dense Attention Modules. DAM is mainly formed by the **a** channel attention, **b** spatial attention, and **c** dependency refinement module

tion. Note that each layer performs the above three steps on the feature map. After assembling the original and enhanced maps, the multi-scale representation is fed into the detection heads for final prediction.

Channel-wise Attention Module. As shown in Fig. 6 (a), we feed the channel-wise attention module with the aggregated representation to allocate the channel-wise weight to various feature maps. In detail, the aggregated representation is squeezed along the channel-wise dimension by a fully connected layer. The derived vector is then fed into the fully connected layers, followed by the softmax layers to obtain the corresponding channel scores.

Spatial-wise Attention Module. As Fig. 6 (b) illustrates, the aggregated representation is fed into the spatial-wise attention module to re-calibrate the spatial-wise weight of the feature maps. Unlike the channel-wise attention module, the pooling operation is performed along the channel dimension. We first impose max pooling and average pooling on the input individually. Then we concatenate these two feature maps and feed them into five convolutional layers followed by softmax layers to obtain the spatial-wise scores of $H \times W$.

Dependency Refinement. The spatial and channel attention adopted earlier primarily focuses on the fusion of the five feature maps according to their contribution. Generally, it plays an indispensable role in better performance in constructing the long-range dependency between distant pixels. Non-Local Block (Wang et al., 2018) is featured of the efficient acquisition of long-range dependency. Thus, we introduce its simplified version to compensate for this drawback in the proposed method. As Fig. 6 (c) depicts, towards the accurate results, we introduce such a module to perform dependency refinement for the aggregated feature maps.

5 Methodology of Multi-Label Classification

Since the proposed dataset can also serve as a benchmark dataset for the multi-label classification task, we extend the spirit of divide-and-conquer to this task. Specifically, we feed the input image $\mathcal{I}$ into the backbone to obtain the basic feature map $\mathcal{F}$. Previous methods feed this representation directly into the multi-label classifier to finalize the prediction. However, this scheme is heavily subject to the long-tailed issue, resulting in degraded performance. Observing the distribution in our proposed dataset, we offer a tree-like framework for this task. As in Fig. 7, in our design, the course-grained node predicts whether prohibited items exist, while the fine-grained node focuses on predicting the specifically prohibited classes. Interestingly, such a simple yet efficient scheme enables us to enjoy significant performance over state-of-the-art methods.

5.1 Binary Classification on the Node $\mathcal{R}_0$

As stated above, the course-grained node $\mathcal{R}_0$ is tasked with the binary classification based on the basic feature map $\mathcal{F}$. In our design, we realize the node $\mathcal{R}_0$ with a light-weight network which is mainly composed of a multi-head attention module (Vaswani et al., 2017), a position-wise feed-forward network (FFN) and a binary classifier, seeing the structure in the bottom left of Fig. 7. In detail, we treat a learnable embedding as the query, the feature map with position embedding as the key, and the original feature map as the value. Following the Equ. (1)-(3), we explore the global dependencies for the query in terms of specific head. We feed the concatenation of multi-head representation into a feed-forward neural network to generate the final feature representation. Experimentally, this scheme enables our model to pay more attention to the

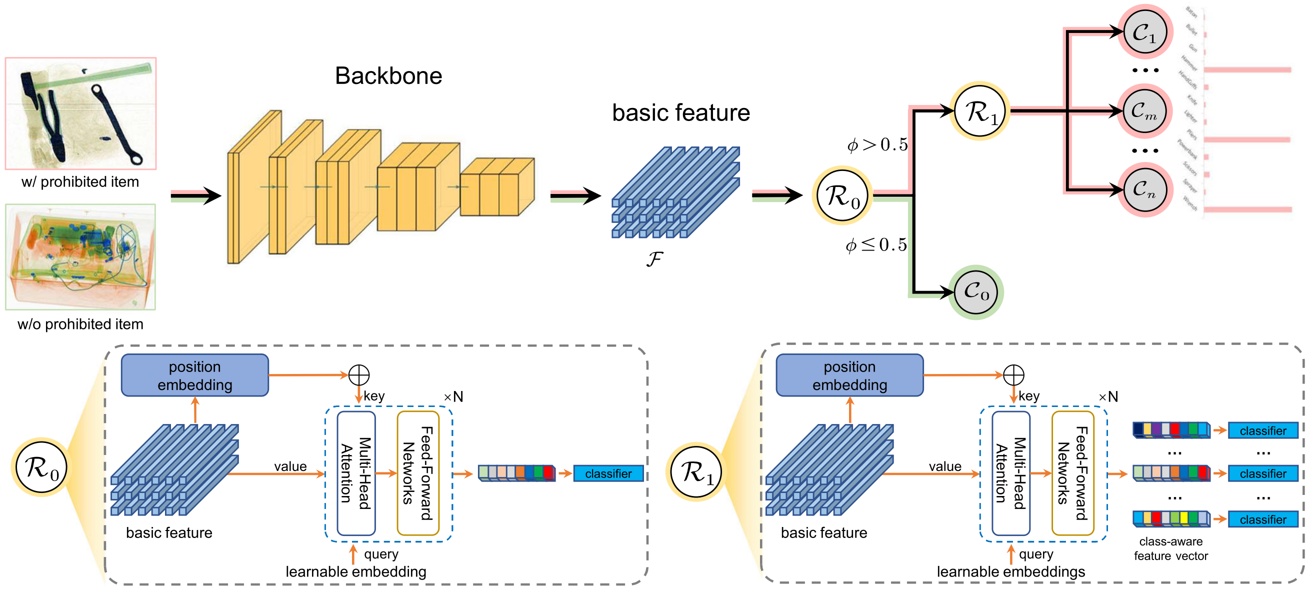

Fig. 7 The overall architecture of the proposed method for the multi-label classification task. The course-grained node $\mathcal{R}_0$ is tasked with binary classification, while the fine-grained node $\mathcal{R}_1$ is dedicated to the multi-label classification of the categories of prohibited items

discriminative regions. Finally, the feature representation is projected to a score ϕ with a binary classifier.

5.2 Multi-label Classification on the Node $\mathcal{R}_1$

With the help of node $\mathcal{R}_0$, the fine-grained node $\mathcal{R}_1$ is dedicated to the multi-label classification of the prohibited item categories. For convenience, we reuse the node $\mathcal{R}_0$ structure design. The node $\mathcal{R}_1$ is responsible for class-aware representation according to the class-specific query. This module allocates different attention weights in the space of the feature map according to specific queries and generates class-aware representations. Finally, the class-aware representations are mapped into a score via an independent binary classifier. The only difference is the node $\mathcal{R}_0$ needs a binary classifier while the node $\mathcal{R}_1$ is equipped with multiple binary classifiers. The structure $\mathcal{R}_1$ in the bottom right of Fig. 7 illustrates the details.

5.3 Loss Function

Our loss function mainly consists of two parts. One is the binary cross-entropy loss of the node $\mathcal{R}_0$, which is formulated as Equ. (4).

$$\mathcal{L}_{bc} = -[y \ln(p) + (1 - y) \ln(1 - p)] \tag{4}$$

where p is the prediction confidence, and y is the corresponding binary label.

The other one is the multi-label classification of the node $\mathcal{R}_1$, here we adopt the Asymmetric Loss (Ridnik et al., 2021a), which is presented as Equ. (5).

$$\mathcal{L}_{ml} = \frac{1}{C} \sum_{k=1}^{C} \begin{cases} (1 - p_k)^{\gamma+} \ln(p_k), & y_k = 1 \\ (p_k)^{\gamma-} \ln(1 - p_k), & y_k = 0 \end{cases} \tag{5}$$

where p_k is the prediction of the k-th class, and y_k is the corresponding label of the k-th class, γ^+ and γ^- are two hyper-parameters.

In a nutshell, the total loss function is expressed as below:

$$\mathcal{L} = \lambda * \mathcal{L}_{ml} + \mathcal{L}_{bc} \tag{6}$$

where λ is the hyper-parameter to balance two loss items, its value in our experiments is set to be the proportion of samples with prohibited items in batch size.

6 Experiments

In this section, we conduct extensive experiments on the PIDray dataset to validate the effectiveness of the proposed method systematically. To be specific, we first describe implementation details and evaluation metrics. Then the performance comparisons between the proposed method and the state-of-the-art methods on the PIDray dataset are reported. Next, the experiments on general image datasets (*i.e.*, COCO and PASCAL VOC) are performed to demonstrate the generalization ability of the proposed model. Finally, we present the insightful analysis to verify the importance of critical components in our method via ablation studies.

Table 3 Overall evaluation of object detection and instance segmentation

Method	Backbone	Detection AP				Segmentation AP			
		Easy	Hard	Hidden	Overall	Easy	Hard	Hidden	Overall
SSD300 (Liu et al., 2016)	VGG16	52.6	41.6	33.0	44.4	–	–	–	–
SSD300+DOAM (Liu et al., 2016; Wei et al., 2020)	VGG16	47.9	38.0	28.3	40.0	–	–	–	–
SSD300+LIM (Liu et al., 2016; Tao et al., 2021a)	VGG16	52.2	43.4	32.8	44.9	–	–	–	–
SSD512 (Liu et al., 2016)	VGG16	56.4	48.4	37.0	49.5	–	–	–	–
FCOS (Tian et al., 2019)	ResNet-101-FPN	62.2	52.5	39.2	54.1	–	–	–	–
RetinaNet (Lin et al., 2017b)	ResNet-101-FPN	65.8	55.1	43.2	57.6	–	–	–	–
YOLOv3 (Redmon & Farhadi, 2018)	DarkNet53	59.6	52.5	35.2	52.3	–	–	–	–
YOLOX-M (Ge et al., 2021)	CSPDarkNet	66.5	62.2	45.8	61.3	–	–	–	–
TOOD (Feng et al., 2021)	ResNet-101-FPN	68.2	63.2	45.3	62.2	–	–	–	–
DW (Li et al., 2022a)	ResNet-101-FPN	67.9	61.3	46.8	61.6	–	–	–	–
DDOD (Chen et al., 2021)	ResNet-101-FPN	69.5	63.6	48.9	63.6	–	–	–	–
DDOD+Ours	ResNet-101-FPN	**71.2**	**64.3**	**50.2**	**64.7**	–	–	–	–
Faster R-CNN (Ren et al., 2015)	ResNet-101-FPN	65.8	56.9	44.2	58.3	–	–	–	–
Libra R-CNN (Pang et al., 219)	ResNet-101-FPN	64.0	55.6	42.1	56.7	–	–	–	–
Mask R-CNN (He et al., 2017)	ResNet-101-FPN	66.2	58.6	43.8	59.1	59.2	50.1	35.5	51.2
Cascade R-CNN (Cai & Vasconcelos, 2019)	ResNet-101-FPN	70.5	61.2	49.1	63.1	–	–	–	–
Cascade Mask R-CNN (Cai & Vasconcelos, 2019)	ResNet-101-FPN	71.9	63.2	46.8	63.7	60.7	52.0	36.2	52.8
Cascade Mask R-CNN (Cai & Vasconcelos, 2019; Liu et al., 2018)	ResNet-101-PAFPN	72.2	63.7	48.3	64.4	61.1	52.0	37.0	53.1
SDANet (Wang et al., 2021) (our conference)	ResNet-101-FPN	72.5	63.7	48.0	64.4	61.1	51.7	37.0	52.9
Cascade Mask R-CNN+Ours	ResNet-101-FPN	**74.5**	**64.8**	**53.0**	**66.6**	**61.4**	**51.9**	**39.7**	**53.4**

The significance of the bold represents the best performance compared with the baselines

6.1 Implementation Details

We adopt the MMDetection[2] toolkit as our training platform, which is performed on a machine with four NVIDIA RTX 3090 GPUs. Our method is implemented in PyTorch deep learning framework. For a fair comparison, all the compared methods are trained on the training set and evaluated on the test set of the PIDray dataset. In terms of object detection and instance segmentation tasks, the proposed pipeline is realized on top of Cascade Mask-RCNN [5], where the ResNet-101 network is used as the backbone. According to our statistics, the average resolution of the images in our dataset is about 500 × 500. Hence, we resize the image to 500 × 500 for compared detectors. The entire network is optimized 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 0.02 and the batch size is set 16. We train 12 epochs and 24 epochs for two-stage and one-stage detectors, respectively. Unless otherwise specified, other hyper-parameters involved in the experiments follow the settings of MMDetection.

We perform all experiments on a machine with eight NVIDIA RTX 3090 GPUs for the multi-label classification task. The proposed model is trained for 80 epochs with an early stopping strategy. We use the Adam optimizer with True-weight-decay (Loshchilov & Hutter, 2017) of 1×10^{-2} and the one cycle policy (Smith & Topin, 2019) to optimize the proposed model. Notably, for ResNet-101-based models, we set the maximum learning rate of 8×10^{-5} and batch size of 288. For TResNetL-based models, we set the maximum learning rate of 1.2×10^{-4} and batch size of 288. For CvT-21-384-based models, we set the maximum learning rate of 1×10^{-4} and batch size of 240. For CvT-w24-384, we set the maximum learning rate of 5×10^{-5} and batch size of 40. Regarding hyper-parameters γ^+ and γ^- in the Equ. (5), we set 0 and 2 for ResNet-101 based models and 0 for other models.

6.2 Evaluation Metrics

Following common metrics of MS COCO (Lin et al., 2014), we evaluate the performance of the compared methods regarding the AP and AR metrics on our PIDray dataset. The scores are averaged over multiple Intersection over Union (IoU). We use 10 IoU thresholds between 0.50 and 0.95. Specifically, the AP score is averaged across all 10 IoU thresholds and all 12 categories. To better assess a model, we look at various data splits. AP_{50} and AP_{75} scores are calculated at IoU = 0.50 and IoU = 0.75 respectively. Note that many prohibited items are small (area $< 32^2$) in the PIDray dataset, which is evaluated by the AP_S metric. Besides, the AR score is the maximum recall given a fixed number of detections (*e.g.*, 1, 10, 100) per image, averaged over 12 front categories and 10 IoUs. For evaluate the performance on small object, we additionally adopt AP_S^{25}, AP_S^{50}, AR_S^{25} and AR_S^{50} to represent the AP and AR of IoU threshold 0.25 and 0.50 respectively. To further evaluate the detection performance, we also introduce the MR^{-2}, which is an important metric in the field of pedestrian detection, and report the missing rate at different FPPI (false positive per image) similar to P-R.

We use mean Average Precision (mAP) as an evaluation metric for the multi-label classification task. It first calculates the Average Precision(AP) for each category, *i.e.*, the area of the Precision-Recall curve, and then averages the AP over all categories.

6.3 Overall Evaluation

Firstly, as presented in Table 3, we report quantitative performance comparisons between our method and numerous one-stage or two-stage state-of-the-art object detectors. Notably, since some methods do not support the instance segmentation task in literature, we fill the placeholder '-' in Table 3 for illustration. Our method achieves superior performance in terms of all metrics on various subsets of the PIDray dataset. For example, compared with the second-best performed methods, our method gains absolute 3.9% and 2.7% AP gain for the two sub-tasks on the hidden test set, which strongly demonstrates the effectiveness of the proposed pipeline. Comparing the SOTA threat detection method DOAM (Wei et al., 2020), which employs two sub-modules to generate an attention distribution map, our method also shows a better result. Based on the assumption that the contours of prohibited items in X-ray images are intact and a large number of prohibited items are deliberately hidden in our dataset, DOAM doesn't perform well here. Besides, it changes the input dimension of the backbone network, impairing the feature extraction ability. LIM (Tao et al., 2021b) proposes an FPN-like structure and a bidirectional propagation module to suppress the irrelevant information near the prohibited items. However, it introduces a heavy calculation burden. It does not fully consider the positive–negative sample proportion in the security inspection scene, which greatly limits performance. As evidenced in Fig. 8 and Fig. 9, our method shows obvious advantages over other methods. For example, in the first column of Fig. 8, all methods except ours miss the powerbank deliberately hidden in the messy objects. In the second and fifth columns, one can see that these baselines over-identify or misidentify some items. In the sixth column, these baselines are prone to error when processing the image without any prohibited items. Figure 9 exhibits a similar visual trend. For instance, visual results show that these baselines generate incomplete masks in the

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

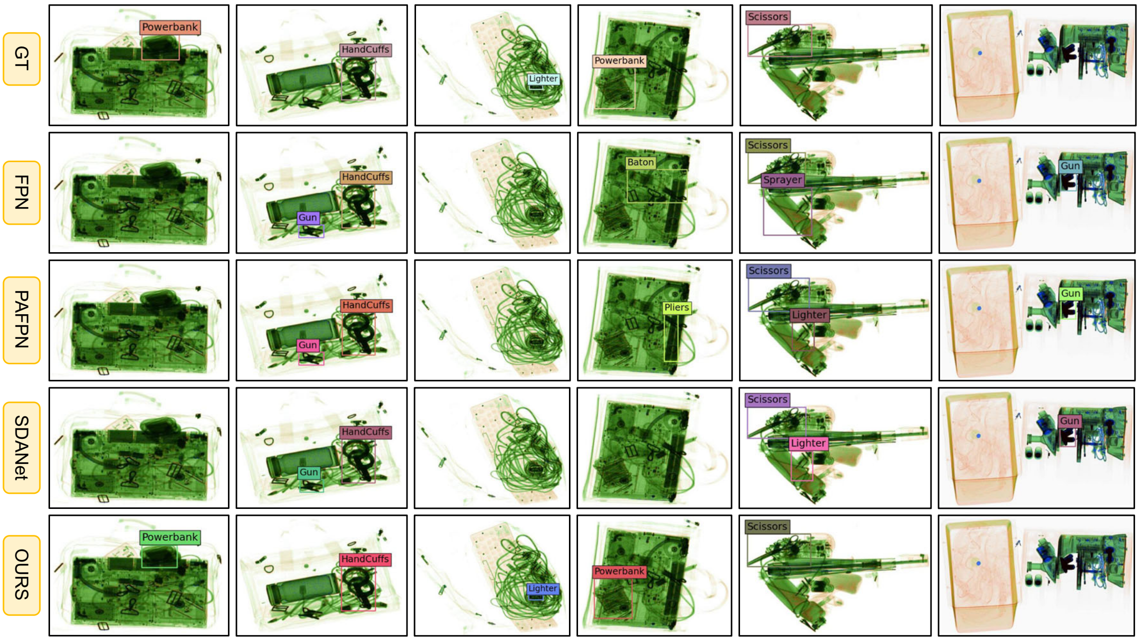

Fig. 8 Visual comparison on detection task. GT indicates Ground-truth, FPN denotes the results generated by Cascade Mask R-CNN with FPN, PAFPN denotes the results generated by Cascade Mask R-CNN with PAFPN, SDANet denotes the results generated by SDANet and Ours indicates the results generated by the proposed method

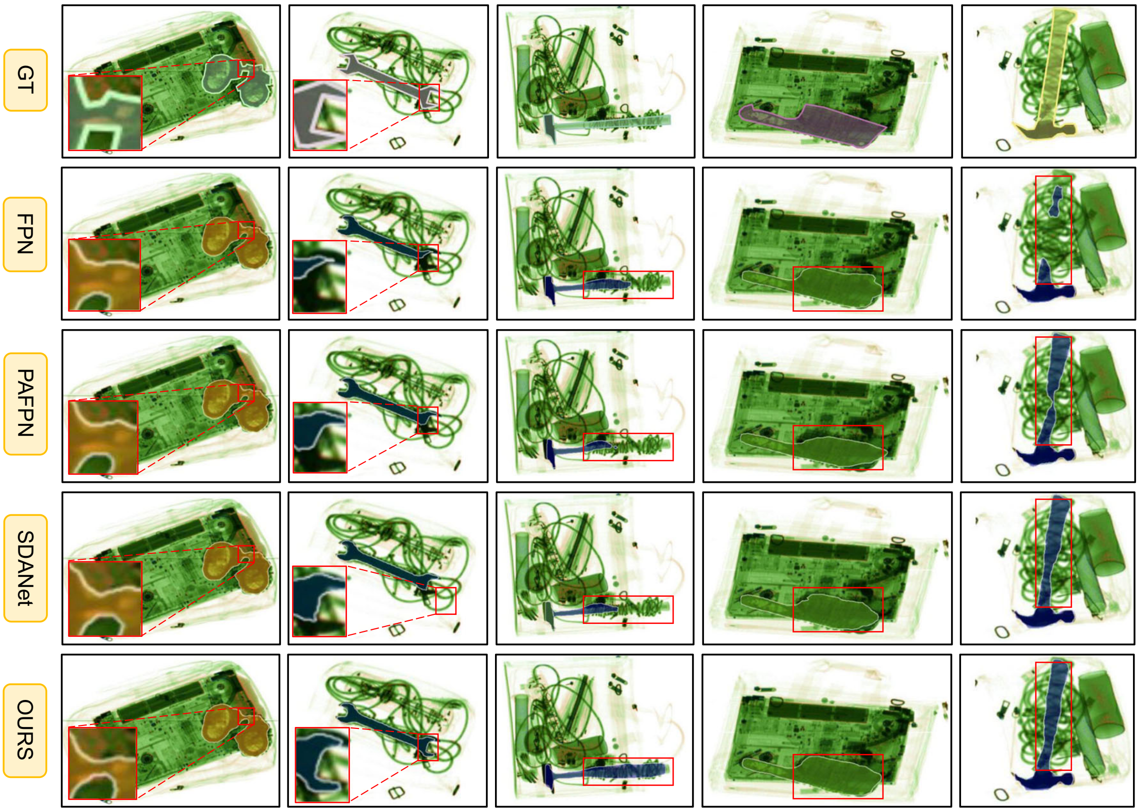

Fig. 9 Visual comparison on instance segmentation task. GT indicates Ground-truth, FPN denotes the results generated by Cascade Mask R-CNN with FPN, PAFPN denotes the results generated by Cascade Mask R-CNN with PAFPN, SDANet denotes the results generated by SDANet and Ours indicates the results generated by the proposed method

Table 4 The detection performance of small objects

Method	AP^S_{25}	AP^S_{50}	AR^S_{25}	AR^S_{50}
SSD300 (Liu et al., 2016)	42.9/–	39.0/–	78.8/–	64.2/–
SSD300+DOAM (Liu et al., 2016; Wei et al., 2020)	36.0/–	29.5/–	76.8/–	51.7/–
SSD300+LIM (Liu et al., 2016; Tao et al., 2021b)	53.3/–	40.2/–	73.7/–	53.9/–
FCOS (Tian et al., 2019)	64.9/–	50.8/–	82.2/–	67.7/–
RetinaNet (Lin et al., 2017b)	68.4/–	64.9/–	82.7/–	75.5/–
SSD512 (Liu et al., 2016)	38.6/–	35.2/–	61.2/–	52.3/–
YOLOv3 (Redmon et al., 2016)	51.5/–	43.3/–	56.8/–	46.8/–
YOLOX-M (Ge et al., 2021)	70.0/–	64.4/–	73.9/–	68.0/–
TOOD (Feng et al., 2021)	68.3/–	63.1/–	81.9/–	74.4/–
DW (Li et al., 2022a)	60.4/–	55.1/–	**85.3**/–	**75.4**/–
DDOD (Chen et al., 2021)	**72.4**/–	**65.3**/–	82.5/–	73.1/–
DDOD+Ours	71.8/–	65.0/–	83.0/–	74.9/–
Faster R-CNN (Ren et al., 2015)	58.8/–	49.5/–	67.3/–	55.3/–
Libra R-CNN (Pang et al., 219)	65.0/–	62.3/–	**74.8**/–	**70.3**/–
Mask R-CNN (He et al., 2017)	56.7/32.5	49.0/28.1	64.4/63.5	52.2/57.8
Cascade R-CNN (Cai & Vasconcelos, 2019)	63.3/–	59.8/–	68.1/–	64.4/–
Cascade Mask R-CNN (Cai & Vasconcelos, 2019)	59.4/34.4	55.8/30.5	62.8/61.4	58.3/58.0
Cascade Mask R-CNN PAFPN (Cai & Vasconcelos, 2019; Liu et al., 2018)	55.2/33.7	52.7/29.4	63.4/61.7	60.3/55.2
SDANet (Wang et al., 2021)	59.5/36.8	57.1/34.0	64.7/63.1	60.9/59.2
Cascade Mask R-CNN+Ours	**69.4**/**48.3**	**66.1**/**42.2**	73.3/**72.0**	69.7/**67.2**

The significance of the bold represents the best performance compared with the baselines

boxed area in the second, third, and fifth columns. In the fourth column, incomplete coverage or over-flowing masks characterize the results generated by baselines. These visual comparisons reveal that the previous baselines are challenged to capture the features of hidden items, while our approach detects prohibited items effectively, especially those deliberately hidden. We believe it is because the proposed pipeline endows the model with a more comprehensive understanding of the characteristics of the proposed dataset. Moreover, our contributions to the task-specific node can bring the following two benefits. First, attention-wise modules can propagate semantic information across multi-layers densely. Second, the dependency refinement module can explore long-range dependencies among feature maps. These complementary design choices make our method can detect deliberately hidden data effectively.

In Table 4, we also report the detection performance in terms of the small objects. As can be seen, our method enhances performance across evaluation metrics consistently. An interesting observation is that network paradigm, *i.e.*, *one-stage vs. two-stage*, is more crucial when dealing with tiny items. Compared with Cascade Mask R-CNN, our method in two-stage frameworks gains 9.9/11.5, 9.0/8.2, 8.6/8.9, and 8.8/8.0 on AP^S_{25}, AP^S_{50}, AR^S_{25} and AR^S_{50} respectively, which illustrate the effect of out method. Considering that recall is an important metric in the field of security, we summarize experimental results in terms of the AR and MR^{-2} metric in Table 5. Likewise, our method outperforms existing methods significantly. Moreover, we present the P-R curves for different IoU thresholds in Fig. 10, demonstrating that our method has obvious advantages over all existing methods.

Secondly, we summarize the experimental performance between our method and the state-of-the-art works on the PIDray dataset for multi-labeling classification task in Table 6. Compared with these approaches, the proposed method shows superior performance, regardless of the backbone or image resolution. In detail, our method achieves the best result compared with approaches based on ResNet-101 (He et al., 2016), outperforming the second-best method Q2L (Liu et al., 2021) by an absolute improvement of 1.53 point in terms of the mAP. Regarding the methods built on top of TResNetL (Ridnik et al., 2020), the proposed method also shows superiority over them. When coupled with the CvT (Wu et al., 2021) series backbone, one can see that our method performs best consistently, boosting accuracy by 1.45 and 0.75, respectively. It can be concluded that these state-of-the-art methods still face challenges due to the apparent discrepancies between X-ray and natural images. While our approach relies on the divide-and-conquer pipeline and the attention mechanism to alleviate this issue in an effective manner.

Table 5 AR and MR^{-2} of different methods

One-stage Methods	MR^{-2}	AR	Two-stage Methods	MR^{-2}	AR
SSD300 (Liu et al., 2016)	12.43	56.8	Faster R-CNN (Ren et al., 2015)	14.67	65.9
SSD300+DOAM (Liu et al., 2016; Wei et al., 2020)	15.85	53.7	Libra R-CNN (Pang et al., 219)	14.70	65.1
SSD300+LIM (Liu et al., 2016; Tao et al., 2021b)	12.43	57.7	Mask R-CNN (He et al., 2017)	12.21	66.7
FCOS (Tian et al., 2019)	15.32	66.6	Cascade R-CNN (Cai & Vasconcelos, 2019)	14.72	69.3
RetinaNet (Lin et al., 2017b)	13.62	69.1	Cascade Mask R-CNN (Cai & Vasconcelos, 2019)	14.16	70.4
DW (Li et al., 2022a)	13.87	71.3	Cascade Mask R-CNN PAFPN (Cai & Vasconcelos, 2019; Liu et al., 2018)	14.00	71.1
DDOD (Chen et al., 2021)	11.11	71.6	SDANet (Wang et al., 2021)	14.10	71.4
DDOD+Ours	10.69	72.5	Cascade Mask R-CNN+Ours	11.86	73.0

The significance of the bold represents the best performance compared with the baselines

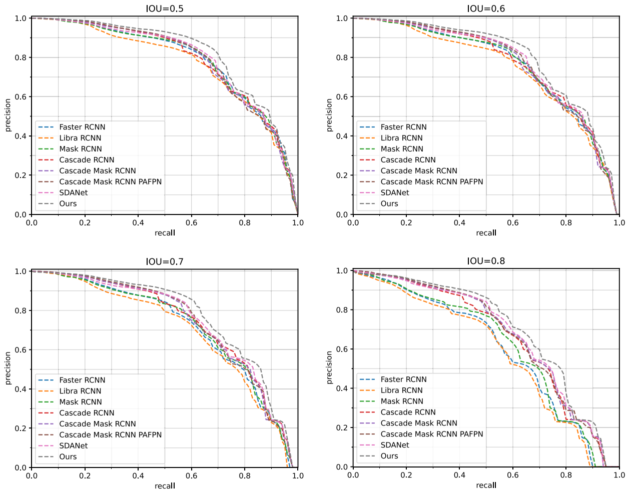

Fig. 10 The precision-recall curve at the IOU thresholds, 0.5, 0.6, 0.7, 0.8

6.4 Evaluation on Other Security Inspection Benchmarks

In this section, we extend our method to other security inspection datasets, *e.g.*, OPIXray V2 (Wei et al., 2020), SIXray (Miao et al., 2019), and present analysis on dilemma existing in COMPASS-XP (Lewis, 2019) and GDXray (Mery et al., 2015).

6.4.1 Evaluation on OPIXray V2

OPIXray V2 (Wei et al., 2020) contains 18,885 X-ray images with five classes, namely, *folding knife*, *straight knife*, *scissor*, *utility knife*, *multi-tool knife*. However, there are only 8,885 images containing the prohibited items. To evaluate the effectiveness of the proposed method, we choose two baselines, *i.e.*, SSD300 and Cascade RCNN. We follow the MMDetection default settings to train each method 120 epochs. Table 7 shows the experiment results. Therein, DOAM, proposed in method (Liu et al., 2016), leverages the different appearance information of the detection target. However, the input dimension change caused by DOMA module greatly limits the feature extraction ability of the backbone. By contrast, there is a great improvement when coupled with our divide-and-conquer pipeline (see the third row). We show a similar trend for the two-stage paradigm between Cascade RCNN and the variant with our divide-and-conquer pipeline. The above experiments demonstrate that our method shows strong performance, which proves the generality and flexibility of our pipeline to a certain degree.

6.4.2 Evaluation on SIXray

SIXray dataset is an image-level annotation dataset of the security inspection scene. This dataset contains 1,059,231 X-ray images with 8,929 images containing the prohibited items (see the 2nd row of Fig. 11). It has three splits, *i.e.*, SIXray10, SIXray100 and SIXray1000, which account for positive–negative proportions 1:10. 1:100 and 1:1,000 relatively. The comparisons are demonstrated in Table 8. As can be seen, our method achieves the best performance among existing methods. Therein, CHR (Miao et al., 2019) inserts some reversed connection in the backbone to deliver high-level visual cues to assist mid-level features. And they design a class-balanced loss function to alleviate the influence of negative samples. However, the changes of the backbone

Table 6 Overall evaluation of Multi-labeling classification

Method	Backbone	Resolution	mAP
CHR (Miao et al., 2019)	ResNet-101	448x448	75.15
CSRA (Zhu & Wu, 2021)	ResNet-101	448x448	86.94
Q2L (Liu et al., 2021)	ResNet-101	448x448	87.69
Ours	ResNet-101	448x448	**89.22 (+1.53)**
ASL (Ridnik et al., 2021a)	TResNetL	448x448	92.91
Q2L (Liu et al., 2021)	TResNetL	448x448	92.99
ML-Decoder (Ridnik et al., 2021b)	TResNetL	448x448	92.81
CSL (Ben-Baruch et al., 2021)	TResNetL	448x448	93.15
Ours	TResNetL	448x448	**93.44 (+0.29)**
Q2L (Liu et al., 2021)	CvT-21	384x384	91.71
Ours	CvT-21	384x384	**93.16 (+1.45)**
Q2L (Liu et al., 2021)	CvT-w24	384x384	93.49
Ours	CvT-w24	384x384	**94.24 (+0.75)**

The significance of the bold represents the best performance compared with the baselines

Table 7 AP results (VOC calculation method) on OPIXray V2 dataset

Method	AP
SSD300 (Liu et al., 2016)	70.7
SSD300 + DOAM (Liu et al., 2016; Wei et al., 2020)	70.5
SSD300 + Our Pipeline	77.7
Cascade R-CNN (Cai & Vasconcelos, 2019)	81.4
Cascade R-CNN + Ours	83.1

hardly guarantee its generality, drastically degrading performance. Q2L utilizes the Transformer structure to replace the conventional fully-connected layer in the classification network. Nevertheless, it is mainly designed for the general scene without fully considering negative samples in the security inspection scenarios, which limits its ability in this field. Different from them, we design a divide-and-conquer pipeline to alleviate the influence of negative samples, which is conducive to better performance. As the positive–negative proportion decreases (SIXray10 and SIXray100), the performance of CHR drops dramatically. In terms of SIXray1000, there are only about 1000 positive samples, which limits the learning representation capability of the network.

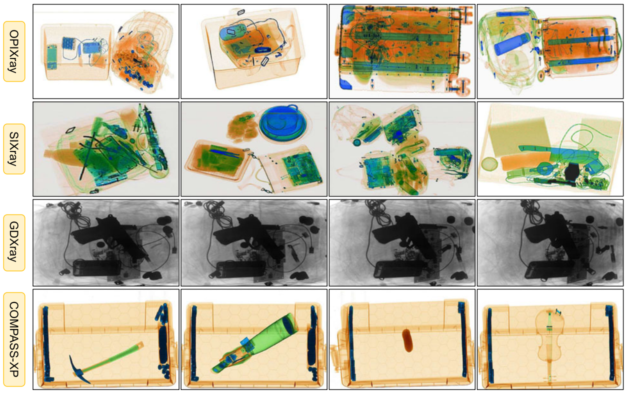

Fig. 11 The exmaples of other security datasets

Table 8 The mAP results on SIXray10, SIXray100, SIXray1000 dataset

Method	SIXray 10	SIXray100	SIXray1000
CHR (Miao et al., 2019)	79.37	60.63	38.14
Q2L (Liu et al., 2021)	89.39	82.47	57.50
Ours	89.70	83.14	59.62

Table 9 AP results (VOC calculation method) on GDXray dataset

Method	AP
Cascade R-CNN (Cai & Vasconcelos, 2019)	88.4
Cascade R-CNN + Ours	89.1

6.4.3 Problems of GDXray

GDXray (Mery et al., 2015) is a grayscale X-ray image dataset for the scenes of *castings*, *welds*, *baggage*, *natural objects* and *settings* (see the 3-rd row of the Fig. 11). The data relevant to the security screening scenario is the Baggage split. There are numerous image series, and each series only contains one scene. The images in the same series differ slightly in rotation. Therefore, the images show extremely high similarity, as the third row of Fig. 11 shows. After our check, we choose the B0009-B0038 series as the training set, which contains 181 images, and select the B0039-B0043 series as the test set, which contains 59 images. Based on this, we test the Cascade R-CNN and its variant with our method. And we gain a 0.7 improvement on this dataset, which prove the effectiveness of our method.

Table 10 Evaluation results on the MS COCO and PASCAL VOC detection datasets

Method	MS COCO2014	MS COCO2017	PASCAL VOC
Baseline	40.8	42.8	81.7
Ours	**41.6**	**43.2**	**82.1**

The significance of the bold represents the best performance compared with the baselines

6.4.4 Problems of COMPASS-XP

Notably, COMPASS-XP (Lewis, 2019) only has 1,928 X-ray images but has 370 general classes (not in the security inspection field). And each image only contains one object (see the last row in Fig. 11), which is far from the realistic scene and incurs each class only contains very limited samples. Such a small amount of data can not offer support for our model, which explains the absence of performance comparison on COMPASS-XP.

6.5 Evaluation on General Dataset

In this part, we shift attention toward general detection datasets to validate the generalization ability of the proposed method on the natural image. The experiments are performed on MS COCO (Lin et al., 2014) and PASCAL VOC (Everingham et al., 2010b), quite popular datasets for the natural image detection domain. For a fair comparison, we follow the default experiment settings in MMDetection. The experimental results are reported in Table 10. Compared with the baseline methods (Cascade Mask R-CNN for MS COCO2014 and MS COCO2017, Cascade R-CNN for PASCAL VOC), we have achieved +0.8 AP, +0.4 AP and +0.4 AP gain on MS COCO2014, MS COCO2017, and PASCAL VOC, respectively. Experimental results demonstrate that our method is not only effective for the detection of prohibited items but also suitable for general scenarios.

6.6 The Generality of Divide-and-Conquer Pipeline

Our proposed divide-and-conquer pipeline can be easily plugged into any threat detection framework. Therefore, we perform extensive experiments of introducing such a pipeline into different detection frameworks without dataset-specific adjustments to the model architecture and reported results in Table 11. The results in Table 11 demonstrate that our proposed pipeline enjoys high flexibility and adaptability to the detection framework. It can be seen that our pipeline brings significant improvements to both one-stage and two-stage methods. Thus, we can conclude that the divide-and-conquer pipeline facilitates existing detectors to handle security inspection datasets, which strongly validates the contributions of this paper.

6.7 Ablation Study

In this section, we perform ablation studies on the necessity of samples without prohibited items, the effect of the divide-and-conquer pipeline, and contributions of the fine-grained node to specific tasks.

6.7.1 Necessity of Samples Without Prohibited Items

In real-world scenarios, it is not always true that only images with prohibited items need to be detected. Hence, an open question is if a model pre-trained on the dataset of which samples with target object account for the majority can work well in security inspection. As Table 12 shows, we report the necessity of samples without prohibited items in the PIDray dataset for object detection task in terms of the mAP and Error Rate. Error Rate evaluates the ability of the network to predict whether or not an image contains prohibited items. We adopt the Cascade Mask R-CNN (Cai & Vasconcelos, 2019) as the test model. During the test, to compute the Error Rate, we define the image of which any bounding box with a con-

Table 11 Generality of the proposed divide-and-conquer pipeline on different detection frameworks

Method	AP	AP (+$\mathcal{R}_0$)
FCOS (Tian et al., 2019)	54.1	57.0 (+2.9)
RetinaNet (Lin et al., 2017b)	57.6	58.6 (+1.0)
YOLOX-M (Ge et al., 2021)	61.3	61.5 (+0.3)
TOOD (Feng et al., 2021)	62.2	64.0 (+1.8)
DW (Li et al., 2022a)	61.6	63.7 (+2.1)
DDOD (Chen et al., 2021)	63.6	64.5 (+0.9)
Faster R-CNN (Ren et al., 2015)	58.3	58.4 (+0.1)
Libra R-CNN (Pang et al., 219)	56.7	57.6 (+0.9)
Mask R-CNN (He et al., 2017)	59.1	59.7 (+0.6)
Cascade R-CNN (Cai & Vasconcelos, 2019)	63.1	63.4 (+0.3)
Cascade Mask R-CNN (Cai & Vasconcelos, 2019)	63.7	65.4 (+1.7)

Table 12 Influence of samples without prohibited items. Error Rate denotes determining whether or not they contain prohibited items

Training Dataset	Detection AP	Error Rate
w/o Non-Prohibited Samples	63.2	21.4
Full Dataset	63.7	4.8

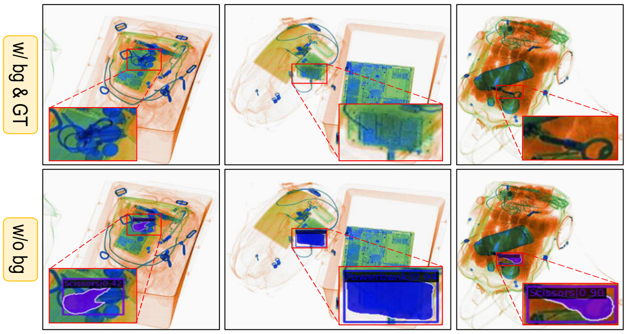

Fig. 12 The influence of images without prohibited items

Table 13 Influence of samples without prohibited items

Train Dataset	mAP	Error Rate	FP Rate
w/o Non-Prohibited Samples	90.04	61.70	≈100
Full Dataset	93.49	6.3×10^{-3}	≈0

fidence level greater than 0.5 to be a sample with prohibited items. By comparison, we can conclude that the introduction of samples without prohibited items to the training set not only improves detection accuracy (+0.5), but also significantly reduces the error rate (−16.6). And as evidenced in Fig. 12, a model trained on a training set excluding samples that do not contain prohibited items still has an item circled on the normal test image.

We also verify this argument on the multi-label classification task. The neural model should predict 0 for each category when no prohibited items are present. We use the Q2L with CvT-w24-384 backbone as the test model. Regarding Error Rate, we define the input as a sample without prohibited items when all twelve categories in the dataset present a confidence score below 0.5. As shown in Table 13, performance on the full dataset significantly improves mAP (+3.45) and reduces the Error Rate (−61.7). Further, we report the rate of false positive rate (FP). If there are no samples without prohibited items in the training dataset, the network tends to guess at least one category wildly, leading to an oddly high rate of misjudgments.

Based on the above observation, we conclude that there is a strong necessity to introduce samples without prohibited items into the PIDray dataset.

Table 14 Effectiveness of divide-and-conquer pipeline

Method	AP	AP_{50}	AP_{75}	AP_S	AR_1	AR_{10}	AR_{100}	AR_S	AP^S_{25}	AP^S_{50}	AR^S_{25}	AR^S_{50}
DDOD	63.6/–	79.0/–	70.1/–	44.7/–	66.4/–	71.5/–	71.6/–	52.8/–	72.4/–	65.3/–	82.5/–	73.1/–
w/ Pipline	64.5/–	79.9/–	71.2/–	41.6/–	67.0/–	72.1/–	72.2/–	49.6/–	67.8/–	61.9/–	82.3/–	75.9/–
Cascade Mask R-CNN	63.7/52.5	78.6/76.3	71.1/60.6	43.6/17.1	66.7/55.9	70.4/58.3	70.4/58.3	45.7/33.5	59.4/34.4	55.8/30.5	62.8/61.4	58.3/58.0
w/ Pipline	65.4/52.7	80.3/77.6	72.8/60.5	43.0/17.2	68.0/55.9	71.7/58.2	71.7/58.2	50.0/36.3	62.3/36.9	58.9/31.5	71.5/69.3	67.5/64.2

All models are trained on the PIDray *training* subset and tested on the PIDray *test* set. The "detection AP/segmentation AP" denotes accuracy

Table 15 Effectiveness of key modules

CA	SA	DR	AP	AP_{50}	AP_{75}	AP_S	AR_1	AR_{10}	AR_{100}	AR_S	AP^S_{25}	AP^S_{50}	AR^S_{25}	AR^S_{50}
			65.4/52.7	80.3/77.6	72.8/60.5	43.0/17.2	68.0/55.9	71.7/58.2	71.7/58.2	50.0/36.3	62.3/36.9	58.9/31.5	71.5/69.3	67.5/64.2
✓			65.9/53.2	81.0/78.2	73.2/61.0	45.1/18.8	68.6/56.5	72.3/58.8	72.3/58.9	49.3/37.2	66.0/38.4	61.7/33.2	73.3/71.0	68.2/65.1
	✓		65.7/52.7	80.9/77.9	73.1/60.4	49.1/18.4	68.2/55.9	71.9/58.1	71.9/58.1	50.9/35.2	69.9/39.3	64.9/32.1	73.6/71.8	67.6/63.2
✓	✓		66.1/53.1	80.9/78.3	73.8/61.0	40.2/18.0	68.8/56.4	72.4/58.6	72.4/58.6	43.1/31.3	57.4/36.4	52.1/31.7	61.6/58.5	56.1/53.9
✓	✓	✓	66.6/53.4	81.7/78.6	74.3/61.4	49.9/24.3	69.2/56.6	73.0/58.9	73.0/58.9	52.4/38.6	69.4/48.3	66.1/42.2	73.3/72.0	69.7/67.2

All models are trained on the PIDray *training* subset and tested on the PIDray *test* set. CA, SA, and DR represent channel attention, spatial attention, and dependency refinement, respectively. The accuracies are denoted by "detection AP/segmentation AP"

6.7.2 Effect of the Divide-and-Conquer Pipeline

To verify the effect of our proposed divide-and-conquer pipeline, we choose Cascade Mask R-CNN (Cai & Vasconcelos, 2019) and DDOD (Chen et al., 2021) as the baseline to add a coarse-grained node between the backbone and detection head. As Table 14 shows, we achieve significant improvements (+0.9 and +1.7 on detection AP) over the two baselines. We believe it is because the original fine-grained node can focus more on the samples with prohibited items with the help of the coarse-grained node. When detached from the coarse-grained node, the task-specific node is challenged by overwhelming samples without prohibited items, resulting in degraded performance.

6.7.3 Contributions of the Fine-Grained Node to Object Detection/ Instance Segmentation

We conduct a set of experiments to verify the design of our proposed Dense Attention Modules (DAMs) in the fine-grained node. We use the Cascade Mask R-CNN (Cai & Vasconcelos, 2019) with ResNet-101-FPN backbone and our divide-and-conquer pipeline as baselines. The experimental performance is presented in Table 15. We can see that modules in our method improve the baseline strikingly when adding them one by one. Desirably, the assembly of these modules contributes to better performance. We believe that it is because our proposed DAMs not only fuse different feature maps in the FPN at each scale depending on their different importance but also solve the problem of long-distance dependency between different pixels to a certain extent.

6.7.4 Contributions of the Fine-Grained Node to Multi-Label Classification Task

To demonstrate the task-specific contribution in the multi-label classification, we conduct ablation experiments with different network components on the top of the CvT-21-384 backbone. In the first two lines of Table 16, we use global max pooling (GMP) or global average pooling (GAP) to process the feature map $\mathcal{F}$ and then deliver it to the classifier directly. In the third line, we adopt different queries to obtain corresponding class-aware feature vectors, which are used to finalize multi-label classifications. As seen from Table 16, it strongly proves the effectiveness of our designed module. Next, we introduce the divide-and-conquer pipeline, which brings an improvement of 0.37 in terms of mAP, demonstrating that the divide-and-conquer pipeline is also conducive to the multi-label classification task.

Table 16 Comparison with the contribution of different parts of multi-label classification task

Method	mAP
Baseline (GMP)	91.91
Baseline (GAP)	92.43
Cross-Attention Module	92.79
+Divide-and-Conquer Pipeline	93.16

7 Conclusion

In this paper, we construct a challenging dataset (namely PIDray) for prohibited item detection, especially in cases where the prohibited items are hidden in other objects. Moreover, all images with prohibited items are annotated with bounding boxes and masks of instances. As far as we know, PIDray is a significant dataset with the largest volume and varieties of annotated images with prohibited items to date. Meanwhile, we design a divide-and-conquer pipeline to make the proposed model more suitable for real-world application. Specifically, we adopt the tree-like structure to suppress the influence of a long-tailed issue in the PIDray dataset. The first course-grained node is tasked with binary classification to alleviate the influence of the head category. In contrast, the subsequent fine-grained node is dedicated to the specific tasks of the tail categories. Based on this simple yet effective scheme, we offer strong task-specific baselines across the object detection and instance segmentation to multi-label classification tasks on the PIDray dataset and verify its generalization ability on common datasets (like COCO and PASCAL VOC). We hope the proposed dataset will help the community establish a unified platform for evaluating the prohibited item detection methods for real applications. For future work, we will consider the object orientation factor and plan to extend the current dataset to include more images and richer annotations for comprehensive evaluation.

Data Availability Our benchmark and codes are released at https://github.com/lutao2021/PIDray.

References

Akcay, S., & Breckon, T. (2022). Towards automatic threat detection: A survey of advances of deep learning within x-ray security imaging. *Pattern Recognition, 122*(108), 245.

Akcay, S., & Breckon, T.P. (2017). An evaluation of region based object detection strategies within x-ray baggage security imagery. In IEEE International Conference on Image Processing (ICIP), pp. 1337–1341.

Akcay, S., Kundegorski, M. E., Willcocks, C. G., et al. (2018). Using deep convolutional neural network architectures for object classification and detection within x-ray baggage security imagery. *IEEE Transactions on Information Forensics and Security, 13*(9), 2203–2215.

Ben-Baruch, E., Ridnik, T., Friedman, I., et al. (2021). Multi-label classification with partial annotations using class-aware selective loss. arXiv preprint arXiv:2110.10955.

Cai, J., Wang, Y., & Hwang, J.N. (2021). Ace: Ally complementary experts for solving long-tailed recognition in one-shot. In *IEEE International Conference on Computer Vision (ICCV)*, pp. 112–121.

Cai, Y., Du, D., Zhang, L., et al. (2020). Guided attention network for object detection and counting on drones. In *ACM International Conference on Multimedia (ACM MM)*, pp. 709–717.

Cai, Z.,& Vasconcelos, N. (2018). Cascade r-cnn: Delving into high quality object detection. In *IEEE International Conference on Computer Vision and Pattern Recognition Conference* (CVPR), pp. 6154–6162.

Cai, Z., & Vasconcelos, N. (2019). Cascade r-cnn: High quality object detection and instance segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 43*(5), 1483–1498.

Cao, Y., Xu, J., Lin, S., et al. (2019). Gcnet: Non-local networks meet squeeze-excitation networks and beyond. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*.

Carion, N., Massa, F., Synnaeve, G., et al. (2020). End-to-end object detection with transformers. In *European Conference on Computer Vision (ECCV)*, Springer, pp. 213–229.

Chen, K., Cao, Y., Loy, C.C., et al. (2020). Feature pyramid grids. arXiv preprint arXiv:2004.03580.

Chen, Z., Yang, C., Li, Q., et al (2021) Disentangle your dense object detector. In *ACM International Conference on Multimedia (ACM MM)*, pp. 4939–4948.

Cui, Y., Jia, M., Lin, T.Y., et al. (2019). Class-balanced loss based on effective number of samples. In *IEEE International Conference on Computer Vision and Pattern Recognition Conference (CVPR)*. pp. 9268–9277.

Duan, K., Bai, S., Xie, L., et al .(2019). Centernet: Keypoint triplets for object detection. In*IEEE International Conference on Computer Vision (ICCV)*. pp. 6569–6578.

Everingham, M., Gool, L., Williams, C. K., et al. (2010). The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision, 88*(2), 303–338.

Everingham, M., Van Gool, L., Williams, C. K., et al. (2010). The pascal visual object classes (voc) challenge. *International Journal of Computer Vision, 88*, 303–338.

Feng, C., Zhong, Y., Gao, Y., et al. (2021). Tood: Task-aligned one-stage object detection. In *IEEE International Conference on Computer Vision (ICCV)*, pp. 3490–3499.

Fu, C.Y., Liu, W., Ranga, A., et al (2017) Dssd: Deconvolutional single shot detector. arXiv preprint arXiv:1701.06659.

Gao, B. B., & Zhou, H. Y. (2021). Learning to discover multi-class attentional regions for multi-label image recognition. *IEEE Transactions on Image Processing, 30*, 5920–5932.

Gao, Z., Xie, J., Wang, Q., et al. (2019). Global second-order pooling convolutional networks. In *IEEE International Conference on Computer Vision and Pattern Recognition Conference (CVPR)*. pp. 3024–3033.

Ge, Z., Liu, S., Wang, F., et al. (2021). YOLOX: Exceeding yolo series in 2021. arXiv preprint arXiv:2107.08430.

Ghiasi, G., Lin, T.Y., & Le, Q.V. (2019). Nas-fpn: Learning scalable feature pyramid architecture for object detection. In *IEEE International Conference on Computer Vision and Pattern Recognition Conference (CVPR)*. pp. 7036–7045.

Girshick, R. (2015). Fast r-cnn. In *IEEE International Conference on Computer Vision (ICCV)*, pp. 1440–1448.

Girshick, R., Donahue, J., Darrell, T., et al. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *IEEE International Conference on Computer Vision and Pattern Recognition Conference (CVPR)*., pp. 580–587.

Gong, Y., Yu, X., Ding, Y., et al. (2021). Effective fusion factor in fpn for tiny object detection. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1160–1168.

He, K., Zhang, X., Ren, S., et al (2016) Deep residual learning for image recognition. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition Conference (CVPR)*. pp. 770–778.

He, K., Gkioxari, G., Dollár, P., et al. (2017). Mask r-cnn. In *IEEE International Conference on Computer Vision (ICCV)*. pp. 2961–2969.

Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation networks. In *IEEE International Conference on Computer Vision and Pattern Recognition Conference (CVPR)*. pp. 7132–7141.

Huang, Z., Wang, X., Huang, L., et al. (2019). Ccnet: Criss-cross attention for semantic segmentation. In *IEEE International Conference on Computer Vision (ICCV)*. pp. 603–612.

Ji, R., Du, D., Zhang, L., et al. (2020a). Learning semantic neural tree for human parsing. In *European Conference on Computer Vision (ECCV)*. pp. 205–221.

Ji, R., Wen, L., Zhang, L., et al. (2020b). Attention convolutional binary neural tree for fine-grained visual categorization. In *IEEE International Conference on Computer Vision and Pattern Recognition Conference (CVPR)*. pp. 10465–10474.

Kang, B., Xie, S., Rohrbach, M., et al. (2019). Decoupling representation and classifier for long-tailed recognition. arXiv preprint arXiv:1910.09217.

Law, H., & Deng, J. (2018). Cornernet: Detecting objects as paired keypoints. In *European Conference on Computer Vision (ECCV)*. pp. 734–750.

Lewis, D. Griffin JTAAMatthew Caldwell. (2019). Compass-xp.

Li, C., Du, D., Zhang, L., et al, (2020). Spatial attention pyramid network for unsupervised domain adaptation. In *European Conference on Computer Vision (ECCV)*. pp. 481–497.

Li, S., Gong, K., Liu, C.H., et al. (2021). Metasaug: Meta semantic augmentation for long-tailed visual recognition. In *IEEE International Conference on Computer Vision and Pattern Recognition Conference (CVPR)*. pp. 5212–5221.

Li, S., He, C., Li, R., et al (2022a) A dual weighting label assignment scheme for object detection. In *IEEE International Conference on Computer Vision and Pattern Recognition Conference (CVPR)*. pp. 9387–9396.

Li, Y., Mao, H., Girshick, R., et al. (2022b). Exploring plain vision transformer backbones for object detection. arXiv preprint arXiv:2203.16527.

Lin, T.Y., Maire. M., Belongie. S., et al. (2014). Microsoft coco: Common objects in context. In *European Conference on Computer Vision (ECCV)*. pp. 740–755.

Lin, T.Y., Dollár, P., Girshick, R., et al. (2017a). Feature pyramid networks for object detection. In *IEEE International Conference on Computer Vision and Pattern Recognition Conference (CVPR)*. pp. 2117–2125.

Lin, T.Y., Goyal, P., Girshick, R., et al. (2017b). Focal loss for dense object detection. In *IEEE International Conference on Computer Vision (ICCV)*. pp. 2980–2988.

Liu, J., Leng, J., Ying, L. (2019a). Deep convolutional neural network based object detector for x-ray baggage security imagery. In *IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*. pp. 1757–1761.

Liu, S., Qi, L., Qin, H., et al. (2018). Path aggregation network for instance segmentation. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition Conference (CVPR)*. pp. 8759–8768.

Liu, S., Huang, D., & Wang, Y. (2019b). Learning spatial fusion for single-shot object detection. arXiv preprint arXiv:1911.09516.

Liu, S., Zhang, L., Yang, X., et al. (2021). Query2label: A simple transformer way to multi-label classification. arXiv preprint arXiv:2107.10834.

Liu, W., Anguelov, D., Erhan, D., et al. (2016). Ssd: Single shot multibox detector. In *European Conference on Computer Vision (ECCV)*. pp. 21–37.

Loshchilov, I., & Hutter, F. (2017). Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101.

Mery, D., Riffo, V., Zscherpel, U., et al. (2015). Gdxray: The database of x-ray images for nondestructive testing. *Journal of Nondestructive Evaluation, 34*(4), 42.

Mery, D., Saavedra, D., & Prasad, M. (2020). X-ray baggage inspection with computer vision: A survey. *IEEE Access, 8*, 145620–145633.

Miao, C., Xie, L., Wan, F., et al. (2019). Sixray: A large-scale security inspection x-ray benchmark for prohibited item discovery in overlapping images. In *IEEE International Conference on Computer Vision and Pattern Recognition Conference (CVPR)*. pp. 2119–2128.

Mnih, V., Heess, N., Graves, A., et al. (2014). Recurrent models of visual attention. Advances in Neural Information Processing Systems In *Proceeings of Neural Information Processing Systems (NIPS)*. 27.

Pang, J., Chen, K., Shi, J., et al. (2019). Libra r-cnn: Towards balanced learning for object detection. In *IEEE International Conference on Computer Vision and Pattern Recognition Conference (CVPR)*. pp. 821–830.

Qin, Z., Zhang, P., Wu, F., et al. (2021), Fcanet: Frequency channel attention networks. In *IEEE International Conference on Computer Vision (ICCV)*. pp. 783–792.

Redmon, J., & Farhadi, A. (2017). Yolo9000: better, faster, stronger. In *IEEE International Conference on Computer Vision and Pattern Recognition Conference (CVPR)*. pp. 7263–7271.

Redmon, J., Farhadi, A. (2018). Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767.

Redmon, J., Divvala, S., Girshick, R., et al. (2016). You only look once: Unified, real-time object detection. In *IEEE International Conference on Computer Vision and Pattern Recognition Conference (CVPR)*. pp. 779–788.

Ren, S., He, K., Girshick, R., et al. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *Annual Conference on Neural Information Processing Systems (NIPS)*. pp. 91–99.

Ridnik, T., Lawen, H., Noy, A., et al. (2020). Tresnet: High performance gpu-dedicated architecture. 2003.13630.

Ridnik, T., Ben-Baruch, E., Zamir, N., et al. (2021a). Asymmetric loss for multi-label classification. In *IEEE International Conference on Computer Vision (ICCV)*. pp. 82–91.

Ridnik, T., Sharir, G., Ben-Cohen, A., et al. (2021b). Ml-decoder: Scalable and versatile classification head. arXiv preprint arXiv:2111.12933.

Saavedra, D., Banerjee, S., & Mery, D. (2021). Detection of threat objects in baggage inspection with x-ray images using deep learning. *Neural Computing and Applications, 33*, 7803–7819.

Sermanet, P., Eigen, D., Zhang, X., et al. (2013). Overfeat: Integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv:1312.6229.

Smith, L.N., & Topin, N. (2019). Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial Intelligence and Machine Learning for Multi-domain Operations Applications*. pp. 369–386.

Tao, R., Wei, Y., Jiang, X., et al. (2021a). Towards real-world x-ray security inspection: A high-quality benchmark and lateral inhibition module for prohibited items detection. In *IEEE ICCV*.

Tao, R., Wei, Y., Li, H., et al. (2021b). Over-sampling de-occlusion attention network for prohibited items detection in noisy x-ray images. arXiv preprint arXiv:2103.00809.

Tian, Z., Shen, C., Chen, H., et al. (2019). Fcos: Fully convolutional one-stage object detection. In *IEEE International Conference on Computer Vision (ICCV)* pp. 9627–9636.

Vaswani, A., Shazeer, N., Parmar, N., et al. (2017). Attention is all you need. NIPS 30 5998–6008.

Velayudhan, D., Hassan, T., Damiani, E., et al. (2022). Recent advances in baggage threat detection: A comprehensive and systematic survey. *ACM Computing Surveys, 55*(8), 1–38.

Wang, B., Zhang, L., Wen, L., et al. (2021). Towards real-world prohibited item detection: A large-scale x-ray benchmark. In *IEEE International Conference on Computer Vision (ICCV)*, pp. 5412–5421.

Wang, J., Chen, K., Xu, R., et al. (2019a). Carafe: Content-aware reassembly of features. In *IEEE International Conference on Computer Vision (ICCV)*. pp. 3007–3016.

Wang, Q., Wu, B., Zhu, P., et al. (2020). Eca-net: Efficient channel attention for deep convolutional neural networks. In *IEEE International Conference on Computer Vision and Pattern Recognition Conference (CVPR)*.

Wang, X., Girshick, R., Gupta. A., et al. (2018) Non-local neural networks. In: IEEE International Conference on Computer Vision and Pattern Recognition Conference (CVPR).

Wang, Y., Gan, W., Yang, J., et al. (2019b). Dynamic curriculum learning for imbalanced data classification. In: *IEEE International Conference on Computer Vision (ICCV)*. pp. 5017–5026.

Wang, Y., Pan, X., Song, S., et al. (2019c). Implicit semantic data augmentation for deep networks. Advances in Neural Information Processing Systems 32

Wei, Y., Tao, R., Wu, Z., et al. (2020). Occluded prohibited items detection: An x-ray security inspection benchmark and de-occlusion attention module. In *Proceedings of the 28th ACM International Conference on Multimedia*. pp. 138–146.

Woo, S., Park, J., Lee, J.Y., et al. (2018). Cbam: Convolutional block attention module. In *European Conference on Computer Vision (ECCV)*., pp. 3–19.

Wu, H., Xiao, B., Codella, N., et al. (2021). Cvt: Introducing convolutions to vision transformers. In *IEEE International Conference on Computer Vision (ICCV)*, pp. 22–31.

Xiang, L., Ding, G., & Han, J. (2020). Learning from multiple experts: Self-paced knowledge distillation for long-tailed classification. In *European Conference on Computer Vision (ECCV)*, Springer, pp. 247–263.

Yin, X., Yu, X., Sohn, K., et al. (2019). Feature transfer learning for face recognition with under-represented data. In *IEEE International Conference on Computer Vision and Pattern Recognition Conference (CVPR)*. pp. 5704–5713.

Zhou, B., Cui, Q., Wei, X.S., et al. (2020). Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In *IEEE International Conference on Computer Vision and Pattern Recognition Conference (CVPR)*, pp. 9719–9728.

Zhu, K., Wu, J. (2021). Residual attention: A simple but effective method for multi-label recognition. In *IEEE International Conference on Computer Vision (ICCV)*, pp. 184–193.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.