applied sciences

MDPI

Article

Deep Feature Fusion Based Dual Branch Network for X-ray Security Inspection Image Classification

Yingda Xu [1,2] and Jianming Wei [1,*]

[1] Shanghai Advanced Research Institute, Chinese Academy of Sciences, Shanghai 201210, China; xuyingda@sari.ac.cn
[2] University of Chinese Academy of Sciences, Beijing 100049, China
* Correspondence: wjm@sari.ac.cn

Abstract: Automatic computer security inspection of X-ray scanned images has an irresistible trend in modern life. Aiming to address the inconvenience of recognizing small-sized prohibited item objects, and the potential class imbalance within multi-label object classification of X-ray scanned images, this paper proposes a deep feature fusion model-based dual branch network architecture. Firstly, deep feature fusion is a method to fuse features extracted from several model layers. Specifically, it operates these features by upsampling and dimension reduction to match identical sizes, then fuses them by element-wise sum. In addition, this paper introduces focal loss to handle class imbalance. For balancing importance on samples of minority and majority class, it assigns weights to class predictions. Additionally, for distinguishing difficult samples from easy samples, it introduces modulating factor. Dual branch network adopts the two components above and integrates them in final loss calculation through the weighted sum. Experimental results illustrate that the proposed method outperforms baseline and state-of-art by a large margin on various positive/negative ratios of datasets. These demonstrate the competitivity of the proposed method in classification performance and its potential application under actual circumstances.

Keywords: multi-label object classification; convolutional neural network; deep feature fusion; dual branch network; X-ray security inspection image

Citation: Xu, Y.; Wei, J. Deep Feature Fusion Based Dual Branch Network for X-ray Security Inspection Image Classification. *Appl. Sci.* **2021**, *11*, 7485. https://doi.org/10.3390/app11167485

Academic Editor: Byung-Gyu Kim

Received: 7 June 2021
Accepted: 11 August 2021
Published: 14 August 2021

1. Introduction

Security issues in public places have always aroused general interest in the whole community. By taking advantage of X-ray penetration, X-ray security inspection devices can inspect the interior of baggage on the premise of ensuring the privacy of personnel. Additionally, they can mark various types of objects in different colors [1], leading them to be the broadest application until now. Traditional manual inspection is highly dependent on the judgment of security personnel, which mainly reflects the following drawbacks:

- The perplexing background of scanned images will affect the detection speed of security personnel;
- The rare occurrence of problematic baggage and the fatigue of security personnel are most likely to cause mistakes and misses;
- Long-term repetitive work is not conducive to the physical and mental health of security personnel.

In the face of increasing crowd density in public places, traditional manual detection is beginning to struggle to cope. Due to this phenomenon, researchers have diverted their attention to machine detection [2] while they facilitate related research. In recent years, the high-speed evolution of deep learning [3], especially convolutional neural networks, has made it a leader in image processing and visual understanding. Nowadays, deep learning is the dominant tool in many scenarios, including recognizing and detecting prohibited objects in X-ray scanned images.

A convolutional neural network (CNN) model mainly stacks multiple convolutional and pooling layers. It follows the fully connected layer, which gives prediction information about image class. The convolutional layer and the pooling layer provide feature extraction and selection operations in the network [4]. With the deepening of the network, the generated features have distinct characteristics. To improve the classification and detection performance of the network, one can take advantage of the information of features from various layers through fusing them. On this basis, to settle prohibited item recognition problems on small-sized objects in X-ray images, enlightened by [5], this paper proposes a CNN architecture based on deep feature fusion.

In addition, in practical applications, security inspection images with prohibited items (henceforth referred to as positive samples) are far less than those without prohibited items (henceforth referred to as negative samples). This means the classification task is a class imbalance problem. Under the consideration of loss function improvement, this paper introduces focal loss [6] to alleviate the class imbalance problem. This loss function can effectively deal with the class imbalance problem.

With the deep feature fusion model and focal loss function, adverse effects brought by difficult-to-classify objects and class imbalance can be dropped to a lower level, thereby improving the classification accuracy of the network. This paper proposes a dual branch network architecture to integrate them effectively. As focal loss is incompatible with the fused feature generated by the model, intuitively adopting focal loss function on deep feature fusion model will not be that effective. Hence, we separate these two modules into a dual branch network, and integrate them in the final loss calculation. For fused features and focal loss, it will avoid incompatibility between them. In addition, taking advantage of their merits as the update of final loss brings mutual supervision in the training stage.

The main work and innovation mechanism of this paper are as follows:

1. Proposes a CNN model based on deep feature fusion. Ablation experiments show its effectiveness on classification performance versus backbone;
2. Introduces focal loss, which can alleviate the class imbalance of the dataset, thereby improving the classification performance of the network;
3. Through integrating the deep feature fusion model and focal loss function, proposes a dual branch convolutional neural network architecture. It further improves the classification performance of the network by the supervision of the training process of one branch on the other.

2. Related Works

2.1. Automatic Prediction of Prohibited Items in X-ray Scanned Images

Contrary to other types of images, X-ray security inspection images have the following characteristics:

- Personal items are often placed casually, which leads to randomly stacked and overlapped objects. Consequently, prohibited items are often occluded by the background;
- In X-ray images, objects composed of the same material will be assigned similar colors, which leads to indistinguishability between prohibited items and background;
- X-ray security inspection image itself relates to the privacy of the inspected person.

These characteristics pose an enormous challenge to many aspects, including detection methods and dataset construction. Due to this, there are relatively few related contributions, especially to the latter.

Earlier work mainly relied on the bag of visual words (BoVW) model to extract handcrafted features [7–9], then adopted support vector machine [10] (SVM) as a classifier to recognize and detect prohibited item objects [11]. Additionally, there are some studies which used sparse representation methods [12].

As deep learning mounts, large scale datasets [13] become crucial. For dataset construction of X-ray security inspection images, the GRIMA (from Grupo de Inteligencia de Máquina, the name of Machine Intelligence Group at the Department of Computer Science of the Pontificia Universidad Católica de Chile) database of X-ray images (GDXray)

proposed in [14] was one of the very few public datasets for a long time. The initial purpose of this dataset is to help people study the performance of hand-crafted features. Hence, relatively few negative samples and a relatively simple background make it inapplicable to meet current demands. More recently, there were also some research works based on deep learning methods for the detection of prohibited items in X-ray security images [15–17]. With the help of transfer learning method based on ImageNet [18] pretrained models and some prevalent object detection techniques such as region-based convolutional neural networks (RCNN) [19], "You Only Look Once" (YOLO) [20], etc. They performed satisfactorily in private datasets or the GDXray dataset. The work completed in [21] proposed a large-scale security inspection X-ray benchmark (SIXray), consisting of more than one million images. There are 8929 images with six classes of prohibited items manually annotated, and each of these images contains at least one of them. The distribution of positive/negative samples in this dataset is extremely imbalanced, and the background information is highly complex. It makes SIXray much more challenging and closer to practical applications. This work provides an authoritative verification benchmark for the related research about X-ray security inspection images. In recent years, many studies have been conducted with the adoption of this dataset [22–25]. Specific to object occlusion in X-ray security inspection image, research [26] proposed a dataset named occluded prohibited items X-ray (OPIXray) image benchmark. This dataset contains 8885 high-quality X-ray images of five classes of prohibited items. It is worthy to note that, to better validate the performance of detectors, the test set is divided into three parts according to occlusion levels.

2.2. Deep Feature Fusion

Deep feature fusion can be divided into early fusion and late fusion according to the sequence of fusion and prediction. Early fusion is to fuse the features of various layers first, and then train the predictor based on the fused features. The main methods of this kind of fusion are concatenating and element-wise sum. In addition, the study conducted in [27] introduced a feature fusion method named discriminant correlation analysis (DCA). DCA maximizes the correlation between corresponding features in the two feature sets, while maximizing the difference between different classes. As a result, the authors of [27] effectively improved the classification performance for recognizing very-high-resolution (VHR) remote sensing images. Late fusion mainly improves performance by fusing the prediction results of different layers. Representative work includes single shot multibox detector (SSD) [28], multi-scale convolutional neural network (Multi Scale-CNN, MS-CNN) [29], etc.

There are also some works utilizing both early fusion and late fusion. One of the representative works is the feature fusion single shot multibox detector (FSSD) [30]. FSSD adds a feature fusion module so that features of the original SSD architecture can be fused. Then, it constructs pyramid feature generator and sends the output to the multibox detector to predict. In addition, it is worth mentioning that research in [31] indicated that, due to the gap between semantic information and high-resolution detail information, simply fusing high-level and low-level features is not effective. On this basis, two optimization schemes were proposed, including the introduction of more semantic information into low-level features and the embedding of more spatial information into high-level features. Experiments in [31] showed that using these schemes can significantly improve the segmentation performance by 4%.

2.3. Class Imbalance and Its Treatment

Class imbalance problem refers to the fact that the gap between training samples of various classes is comparatively large. On this basis, imbalanced training samples will make the model more likely to predict the sample as the majority class. This is as even if the model predicts all the samples as the majority class, it can still achieve a high accuracy rate. Nonetheless, this will bring about the incapability of the model to predict minority classes, and furthermore the loss of generalization. The class imbalance problem regularly

occurs, not only in classification, but also in other computer vision tasks such as object detection and semantic segmentation.

The class imbalance problem is mainly treated by two aspects, namely data and algorithm. Data level treatment balances samples of various classes by sampling the whole dataset, including under-sampling majority class samples and over-sampling minority class samples. In addition, some operations apply data augmentation of minority class samples [32], etc. Algorithm level treatment is mainly based on improving loss function. By increasing the cost that samples with minority class yield, or ignoring the contribution that some samples with majority class make, the attention to the minority class can increase. Research in [33] proposed an online hard example mining algorithm (OHEM), which solves the class imbalance problem by retraining some samples that yield much loss in model training process.

As for binary classification, p denotes probability predicted by the model, y denotes ground-truth label, and $y \in \{0, 1\}$. Define p_t as Formula (1).

$$p_t = \begin{cases} p & \text{if } y = 1 \\ 1 - p\,, & \text{otherwise} \end{cases} \tag{1}$$

Then, cross entropy (CE) is defined as $\mathrm{CE}(p_t) = -\log(p_t)$.

Proposed in [6], the design of focal loss is CE based. Focal loss introduces weight α for the minority class and $1 - \alpha$ for the majority class, to balance the importance of majority and minority class samples. It also adds a formulating factor $(1 - p_t)^\gamma$ with the focusing parameter $\gamma \geq 0$, to distinguish difficult samples ($p_t \leq 0.5$) from easy samples ($p_t > 0.5$). By defining α_t analogously to p_t, focal loss (FL) defines as Formula (2).

$$\mathrm{FL}(p_t) = \alpha_t (1 - p_t)^\gamma \mathrm{CE}(p_t) \tag{2}$$

Extend to multi-label classification, for $\mathbf{p} \in \mathbb{R}^C$ and $\mathbf{y} \in \mathbb{R}^C$ where C denotes the number of classes appearing in the dataset. Then, focal loss transforms into Formula (3).

$$\mathrm{FL}(\mathbf{p}_t) = 1^{\mathrm{T}}[\boldsymbol{\alpha}_t \odot (1 - \mathbf{p}_t)^\gamma \odot \mathbf{CE}(\mathbf{p}_t)] \tag{3}$$

where $\odot$ denotes element-wise multiplication among vectors.

3. Method

3.1. Dual Branch Network Architecture

The deep feature fusion model is in charge of improving the model's perception ability to small-sized objects, and works at the feature level. The focal loss function balances the importance between positive and negative samples; moreover, it distinguishes difficult samples from easy ones, which works at the loss level. To effectively combine their advantages, instead of intuitively adopting the focal loss function on loss calculation of the deep feature fusion model, this paper proposes a dual branch network architecture. Each branch of the network corresponds to one of the above two methods and implements the integration of two branches on the final loss calculation. The specific architecture is shown in Figure 1.

The network output on the left branch is based on a CNN model, and the loss calculation adopts the focal loss function. The network output on the right branch is based on the deep feature fusion model, and the loss calculation adopts a cross entropy loss function. The dual branch network enables the two branches to supervise the training process on each other due to the integration brought about by late fusion, effectively combining the advantages of different branches and further improving the performance of the network.

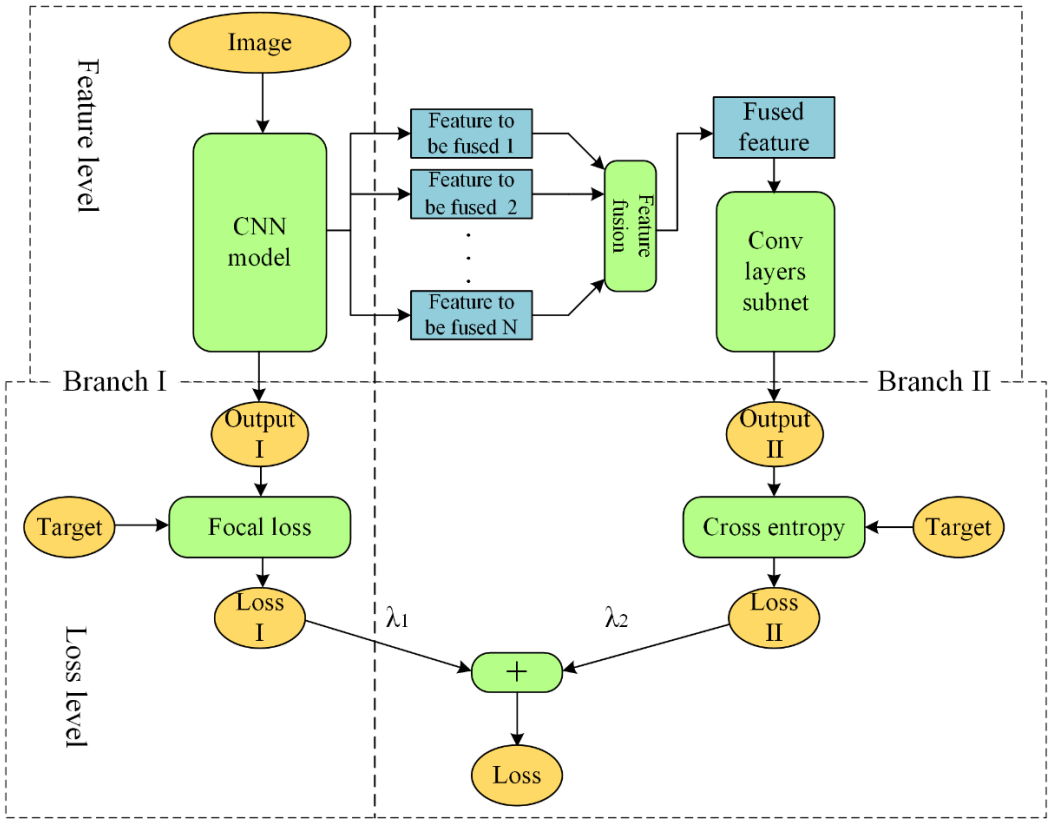

Figure 1. Dual branch network architecture. As for separating it from up to down, the entire architecture can consider as a fusion strategy that adopts both early fusion (up) and late fusion (down) at the level of feature and loss, respectively.

Based on the design of dual branch architecture, the entire network will yield two various outputs and losses. However, in principle, a deep neural network is an end-to-end system, so that the output and loss of the network should be unique. As for the output, we evaluate the performance of the outputs of the two branches on the test dataset and select the output with better performance. Assume that on each branch, as samples of the test dataset delivered into the network, o_{I} and o_{II} denote the yielding output, respectively. $M(o, t)$ denotes the performance measurement function. t denotes ground-truth labels of test dataset. Then, the output of network will be determined as follows:

$$o_{\mathrm{net}} = \mathrm{argmax}_{\{o_{\mathrm{I}}, o_{\mathrm{II}}\}} M(o, t) \tag{4}$$

while the loss calculation of the network adopts the form of weighted sum. Assume that L_{I} and L_{II} denote loss on each branch, respectively. Then, the loss of network is calculated as follows:

$$L_{\mathrm{net}} = \lambda_1 \cdot L_{\mathrm{I}} + \lambda_2 \cdot L_{\mathrm{II}} \tag{5}$$

where λ_1 and λ_2 are hyperparameters preset before training. For the loss of each branch, it controls the contribution to the network, respectively.

3.2. Feature Fusion Model

While generating features in a CNN model, with the network level deepening continuously, the feature maps generated at different levels show various characteristics. Specifically, the low-level features have higher resolution. This better preserves the spatial detail information the image itself has, but the semantic information utilized for recognition is insufficient. The high-level features are processed by multiple convolutional layers so that their semantic information is comparatively rich. However, due to the lack of resolution in the convolutional calculation process, the loss of spatial information is

quite enormous. Consequently, the receptive ability for detail becomes poor. The task of deep feature fusion is to endow the two types of features with efficient fusion and complementary advantages, so as to improve the classification and detection performance of the network.

In single-label image classification tasks, high-level features are adequate to make the model recognize and classify sample images. However, in multi-label object recognition tasks, sample images often contain multiple objects belonging to various classes. In this case, a CNN model is not capable of recognizing all objects when merely using semantic information from high-level features, especially objects of smaller sizes. Therefore, a network designed for single-label classification has poor performance for this task. Given this situation, this paper proposes a CNN model architecture that adds a deep feature fusion module, as shown in Figure 2.

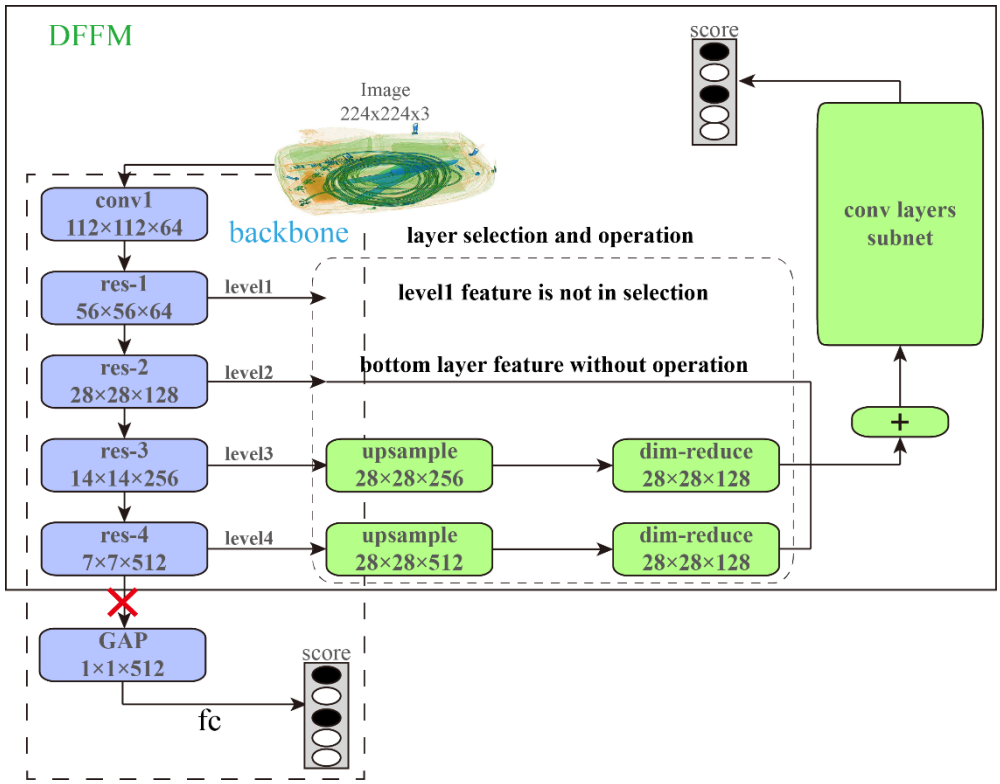

Figure 2. Proposed Deep Feature Fusion Model (DFFM) architecture. Components in leftmost dashed box belong to backbone model (e.g., ResNet34 in this figure). Components in the whole solid box comprise the model proposed in this paper. Numbers (H × W × C) in blocks indicate the output dimension.

As for CNN, a prevalent variant such as ResNet [34] consists of an initial 7 × 7 convolutional layer and four blocks. These blocks usually build by stacks of convolutional layers and pooling layers with similar structures. Sample image delivered to the CNN model will be operated by these components to produce features of various sizes. The feature produced by the first convolutional layer, whose semantic information is so lacking as too defy calculation, is out of consideration. Other features generated by blocks can all be utilized as features to be fused. Our deep feature fusion module precisely revolves around these four features. However, it is noteworthy that not all four features are necessary to attend feature fusion. The selection strategy involves a tradeoff with accuracy versus speed. Further analysis will be described herein in Section 4.4.2 through experiments.

To enable feature fusion between high-level and low-level features and preserve spatial detail information of the low-level feature, we make the plane size of high-level features and low-level features identical using upsampling operation. There are three prevalent

upsampling methods, i.e., deconvolution [35], reshape [36], and bilinear interpolation. Deconvolution brings learnable parameters, which improves the network performance but also increases the computational complexity. Reshape operation increases plane size at the expense of channels, which results in low computational complexity but poor performance. In contrast, bilinear interpolation increases plane size on the premise of keeping channels unchanged. It makes better tradeoff between performance and computational complexity, which is the most prevalent at present. The upsampling operation in this paper adopts this method and mainly acts it on high-level features.

Followed by the upsampling of high-level features, dimension reduction (dim-reduce) operation is performed. Specifically, we introduce a convolution operation with a kernel size of 1×1. This changes output channels, making them smaller than input channels. In addition, the 1×1 convolutional layer can also increase non-linear characteristics and further enhance the capability of the network.

The sizes of high-level and low-level features are already identical after upsampling and dimension reduction. Thus far, all features to be fused can conduct a fusion. This paper chooses element-wise summation to fuse the processed features. After that, the fused feature will deliver to the convolutional layer subnet (conv layer subnet), and then the score output vector for all classes of the sample is obtained. The hierarchical structure of the convolutional layer subnet is affected by the selection of features to be fused. Taking level2/level3/level4 as features to be fused as an example, Table 1 shows the hierarchical structure of the convolutional layer subnet.

Table 1. Convolutional layer subnet hierarchy (based on the selection of level2/level3/level4).

Layer Name	Output Size	Description
input	$28 \times 28 \times 128$	Fused feature
conv1	$14 \times 14 \times 256$	conv: 3×3, stride 2, 128→256; batchnorm, relu
conv2	$7 \times 7 \times 512$	conv: 3×3, stride 2, 256→512; batchnorm, relu
pool	$1 \times 1 \times 512$	Global average pooling (GAP)
fc	5×1	Fully connected (fc), 512→5

4. Experiments

4.1. Dataset

This paper evaluates the proposed method on the SIXray dataset proposed in [21]. The SIXray dataset (as shown in Figure 3) contains 1,059,231 X-ray images. There are 8929 images with six classes of prohibited items manually annotated, namely "gun", "knife", "wrench", "pliers", "scissors", and "hammer". Each positive sample image may contain more than one class of prohibited item. Among them, class "hammer" is deprecated, as too few samples belong to it.

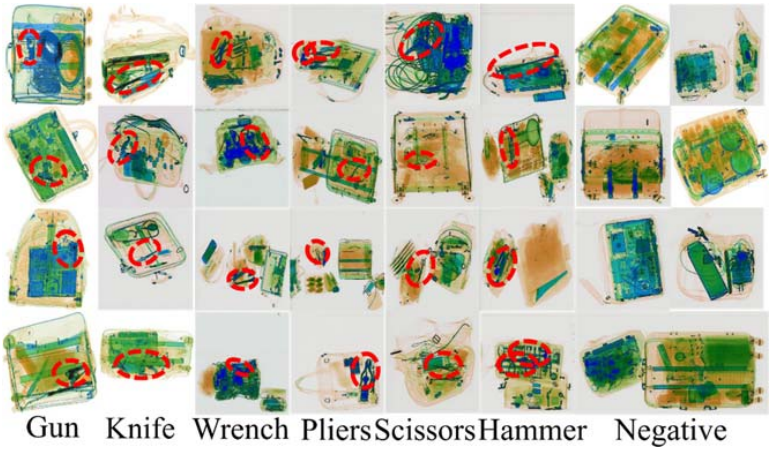

Figure 3. Sample images in SIXray dataset.

SIXray dataset contains three subsets, named SIXray10, SIXray100, and SIXray1000, respectively, according to the ratio of negative samples over positive samples, i.e., 1:10, 1:100, and 1:1000. As positive samples contained in SIXray1000 are different from SIXray10 and SIXray100, we only conducted experiments on SIXray10 and SIXray100 subsets.

For the partition strategy of the training set and test set, we complied with related settings in [21], splitting 7496 positive samples for training set and 1433 positive samples for test set. Then, negative samples were put in correspondence to positive/negative ratio, respectively. Class distribution of positive samples shows in Table 2.

Table 2. Class distribution of positive samples in SIXray dataset. As class "hammer" contains only 60 sample images, we deprecate it in later experiments.

Gun	Knife	Wrench	Pliers	Scissors	Hammer	Total
3131	1943	2199	3961	983	60	8929

4.2. Implementation Details

4.2.1. Baseline

We adopt ResNet34 as the backbone of all network architectures. For the training process, we adopt a transfer learning technique wherein parameters pretrained on ImageNet [18] are used for the pretrained model while reinitializing parameters of the fc layer. For the baseline setting, we adopt CE loss on the backbone of ResNet34 model.

4.2.2. Preprocessing of Images

Similar to the preprocessing procedure in [21] via open-source code provided by the authors, the training image is processed as follows:

- Resize to 256 $\times$ 256 pixels;
- Random cropping to 224 $\times$ 224 pixels;
- Random horizontal flipping with probability of 0.5.

The test image merely resizes to 224 $\times$ 224 pixels without any other processes.

4.2.3. Training Parameters

We trained our networks with Stochastic Gradient Descent (SGD) [37] optimizer. The initial learning rate was set to 1×10^{-2}, momentum parameter to 0.9, and weigh decay parameter to 1×10^{-4}. The total epochs of training are 40. The learning rate decays after the 15th epoch and 30th epoch by a factor of 0.1. The batch size was set to 256.

4.3. Evaluation Metric

We adopt the same evaluation metric as pattern analysis, statistical modeling, and computational learning visual object classes (PASCAL VOC) [38] image classification tasks. Specifically, for each class, all test images containing this class are sorted according to the degree of confidence (i.e., the output of the model). Additionally, the mean Average Precision (mAP) is calculated as the evaluation metric with respect to the classification performance.

4.4. Ablation Studies

4.4.1. Parameter Setting on Focal Loss

For focal loss, there are two vital parameters, namely α and γ. They are similar to hyperparameters such as learning rate, which need to be preset before training. As focal loss aims to alleviate the degree of class imbalance, we first study the influence of class imbalance on setting these parameters.

We refer to the setting of γ with corresponding optima α in [6], conducting experiments on two subsets, SIXray10 and SIXray100. The performance results are shown in Table 3.

Table 3. Effect of varying parameters setting to model performance on SIXray10 and SIXray100 subsets. First row with symbol "–" shows the performance result on baseline.

SIXray10			SIXray100		
γ	α	mAP/%	γ	α	mAP/%
–	–	74.83	–	–	52.74
0	0.5	**82.02** [1]	0	0.5	73.05
0	0.75	81.35	0	0.75	72.94
0.2	0.75	80.72	0.2	0.75	**76.09**
0.5	0.5	76.21	0.5	0.5	73.28
1	0.25	74.29	1	0.25	72.90
2	0.25	67.18	2	0.25	69.80

[1] As for the entire paragraph, number with bold font indicates optimal results.

The experimental results show that the adoption of the focus loss can improve the classification performance of the model for class imbalanced datasets. It has a significant effect on both SIXray10 and SIXray100 subsets. The optimal result has increased by 7.19 points compared with baseline on SIXray10, while 20.31 points on SIXray100. As the results show, the improvement on SIXray100 is far greater than SIXray10. This implies that, for a deeper class imbalance degree, focal loss with appropriate parameter setting can handle the scenario better, due to balancing the importance of negative and positive samples.

As for the value of focus loss parameter on optimal performance, the optimal result of the SIXray10 occurs at $\gamma = 0$, $\alpha = 0.5$. Meanwhile, with the continuous increase in γ, model performance continues to decline. For the focal loss function, when the value of γ is 0, it degrades into a weighted CE loss function. It also shows that the modulating factor of focal loss is not effective in this subset. It is possibly prompted by inadequate difficult samples in SIXray10 when the model is on predicting. As for the SIXray100 subset, the optimal result occurs at $\gamma = 0.2$, $\alpha = 0.75$. As the level of class imbalance further deepens, difficult samples gradually increase. Henceforth, the impact of modulating factor begins to gradually rise, further validating the effectiveness of the loss function setting in this paper.

4.4.2. Level Selection of Features to Be Fused

As for the deep feature fusion model proposed in this paper, we compare three level selections of features to be fused through adding low-level features to the fusion stepwise, i.e., level4/level3, level4/level3/level2, and level4/level3/level2/level1. We give the experimental results as shown in Table 4. The experiment here is based on the SIXray10 subset with the adoption of cross entropy loss function. To validate the effectiveness of the proposed model, the first row of Table 4 illustrates the experimental result of the baseline method.

Table 4. Effect of level selection of features to be fused on model performance.

Levels of Features to Be Fused	mAP/%
Baseline (without fusion)	74.83
level4/level3	79.22
level4/level3/level2	81.07
level4/level3/level2/level1	**81.27**

Through comparing the experimental results, we find that, as more levels attend to fusion, the classification performance of the model increases. On the other hand, this margin of growth is gradually reduced, from 4.39 points brought by merely adding level3 to the fusion, to 1.85 and 0.2 points brought by adding level2 and level1, successively. For the features of these four levels, higher-level ones need a larger parameter amount of occupancy. It implies that, for later added high-level features, the performance improvement gained by consuming so much computational cost is extremely limited. In other words, the improvement is inefficient, especially for the addition of level1. It also shows that the

output feature of level1 is inappropriate to add into the feature fusion module. Getting to the bottom of this, the feature of this layer experiences few convolution calculations, which brings about poor expressive ability. In consequence, no more information will be obtained by fusing the feature of this layer. Considering performance and computational cost comprehensively, we finally choose features of level4/level3/level2 for the fusion procedure.

4.4.3. Loss Weight Setting of Dual Branch Network

We design a dual branch network by integrating the deep feature fusion model and focal loss function, which reflects the importance of each branch through setting weights (λ_1 and λ_2) of loss on it. After the optimal setting of these two modules is determined, we pick multiple pairs of λ_1 and λ_2 for conducting classification performance experiments of dual branch network on SIXray10 subset, whose results show in Table 5.

Table 5. Effect of loss weight of each branch on dual branch network performance.

λ_1	λ_2	**mAP/%**	λ_2	λ_1	**mAP/%**
1	0.25	79.52	1	0.25	81.28
	0.5	80.09		0.5	80.83
	1	82.52		1	82.52
	2	81.92		2	81.73
	3	**83.10**		3	81.78
	5	81.92		5	82.05

Firstly, comparing experimental results row-wise, we find that network performance is better when λ_2 is larger than λ_1. Furthermore, investigating weight setting for optimal performance, we notice that network performance reaches the peak when $\lambda_1 = 1$, $\lambda_2 = 3$. This implies that, as for the dual branch network proposed in this paper, it is demanding to expand the importance of branch II, i.e., deep feature fusion module, for better network performance.

4.4.4. Results of Ablation Studies

To validate the effect of each component, we formed a table of ablation results by summarizing optimal results in the previous three sections, as shown in Table 6.

Table 6. Ablation results of each component based on SIXray10 subset.

	Backbone	**Feature Fusion**	**Focal Loss**	**mAP/%**
Baseline	ResNet34			74.83
Ours	ResNet34	√		81.07
			√	82.02
		√	√	**83.10**

From Table 6, we find that the two components, feature fusion and focal loss, are both capable of gaining considerable performance improvement. As improvement adopted by focal loss (7.19 points) is higher than that adopted by feature fusion module (6.24 points), it implies that:

- As for the class imbalance dataset, it is of the highest urgency to balance importance on positive and negative samples;
- As for the deep feature fusion module, the considerable optimization on model performance also validates its effectiveness.

Optimal performance is obtained through adopting the dual branch network, which integrates these two components. Hence, we can conclude that the dual branch network proposed in this paper has advantages of these two components complemented and makes effective integration. Furthermore, it will make better effects on the task of multi-label object classification in X-ray security inspection images.

4.5. Overall Performance Study

Table 7 compares performance results of baseline method with the Class-balanced Hierarchical Refinement (CHR) method, proposed in [21], and the dual branch network, proposed in this paper, on SIXray10 and SIXray100 subsets.

Table 7. Result of multi-label classification performance.

Dataset	AP/%	Baseline	CHR [21]	Ours
SIXray10	Gun	89.71	87.16	**90.66**
	Knife	85.46	87.17	**89.99**
	Wrench	62.48	64.31	**79.90**
	Pliers	83.50	85.79	**89.14**
	Scissors	52.99	61.58	**65.79**
	mean (mAP)	74.83	77.20	**83.10**
SIXray100	Gun	83.06	81.96	**84.84**
	Knife	78.75	77.70	**86.98**
	Wrench	30.49	36.85	**70.00**
	Pliers	55.24	64.56	**81.92**
	Scissors	16.14	14.49	**58.36**
	mean (mAP)	52.74	55.11	**76.42**

The analysis of Table 7 leads to the conclusion that compared to baseline and CHR, the method proposed in this paper achieves optimal performance of all classes. To show the difference of performance between the three methods more intuitively, according to Table 7, we compiled a bar chart, as shown in Figure 4.

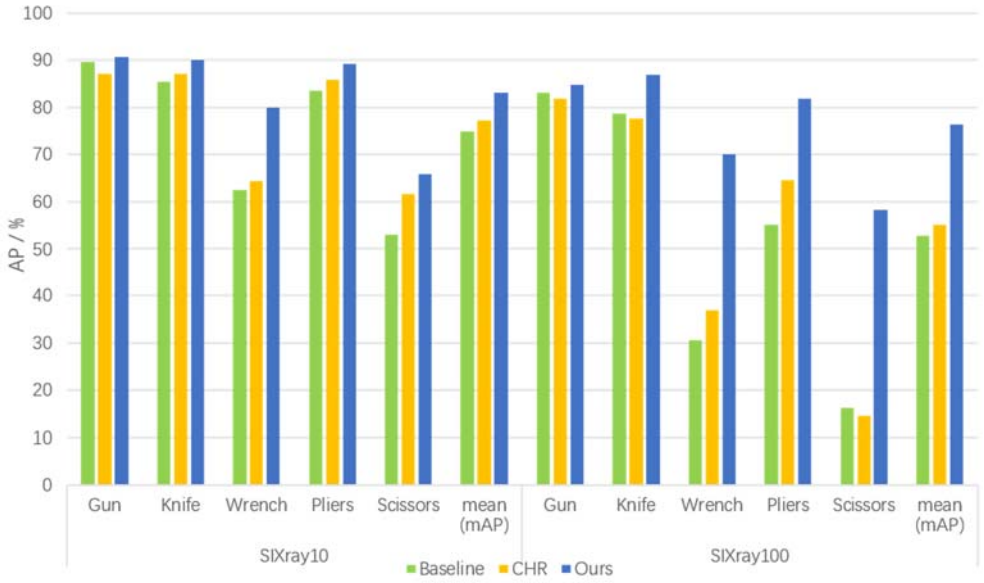

Figure 4. The performance gained by the proposed method shows enormous variation between classes.

Considering overall performance (mAP), compared to the CHR method, our method gains more improvement, with 5.90 points on SIXray10 and 21.31 points on SIXray100. Compared with the baseline method, the value achieves 8.27 points on SIXray10 and 23.68 points on SIXray100. It is worth mentioning that the improvement on SIXray100 is far more than that on SIXray10. This implies that, for the dataset with a higher level of class imbalance, our method gains more on performance. It would benefit from special treatment for class imbalance by the adopted focal loss function.

Considering average precision (AP) on a single class, improvements on various classes exist in enormous variation. As for classes wrench and scissors, on which baseline performs

worse, our method outperforms baseline by 17.42 and 12.80 points on the SIXray10 subset, and by 39.51 and 42.22 points on the SIXray100 subset, respectively. It plays a primary role in the improvement of overall performance. Based on this, it appears that our method makes a better alleviation to objects which suffer greater prediction bias in the training stage, as samples of class scissors are the least in the SIXray dataset. As for class wrench, without consideration of the distribution issue that the dataset itself has, it can be thought that the deep feature fusion module enhances receptive ability of the model to objects that belong to it.

5. Conclusions and Future Work

In this paper, we investigate the recognition problem of prohibited items in X-ray security inspection images. We propose a deep feature fusion model architecture that effectively takes advantage of spatial information of low-level features and semantic information of high-level features through fusing features of various levels in the backbone. For the presence of class imbalance within data samples in actual applications, we introduce focal loss to alleviate the prediction bias caused by it. To integrate these two components effectively, we propose a dual branch network. It reflects them on two branches and integrates training on two branches using weighted sum in final loss calculation. Experimental results on SIXray dataset demonstrate that the proposed method outperforms the baseline method and previous state-of-art by a large margin.

In future work, as the studied task is strongly real-time oriented, related research should be more considerate of judgment speed with acceptable accuracy guaranteed. One-stage detectors such as YOLOv4 [39] can afford some inspiration. Additionally, other applications of deep neural networks in the security domain, such as backdoor samples [13], are also worth continued study.

Author Contributions: Validation, J.W.; investigation, Y.X.; resources, J.W.; writing—review and editing, J.W.; visualization, Y.X.; and project administration, Y.X. Both authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Special Fund for Basic Research on Scientific Instruments of the National Natural Science Foundation of China, grant number 51827814, and Science and Technology Innovation Plan of Shanghai Science and Technology Commission, grant number 19DZ1202200.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The public dataset "SIXray" utilized in this paper can be found on the following link: https://github.com/MeioJane/SIXray (accessed on 11 August 2021).

Acknowledgments: We thank peers from Security and Emergency Laboratory for their inspiration of idea and guidance on conducting experiments, moreover review of this manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chen, Z.Q.; Zhang, L.; Jin, X. Recent progress on X-ray security inspection technologies. *Chin. Sci. Bull.* **2017**, *62*, 1350–1364. [CrossRef]
2. Mery, D. X-ray testing by computer vision. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops, Portland, OR, USA, 23–28 June 2013; pp. 360–367.
3. Lecun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef] [PubMed]
4. Song, W.; Li, S.; Fang, L.; Lu, T. Hyperspectral Image Classification with Deep Feature Fusion Network. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 3173–3184. [CrossRef]
5. Liu, Y.; Chen, X.; Wang, Z.; Ward, R.K.; Wang, X. Deep learning for pixel-level image fusion: Recent advances and future prospects. *Inf. Fusion* **2018**, *42*, 158–173. [CrossRef]
6. Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the 2017 IEEE International Conference on Computer Vision, Honolulu, HI, USA, 22–29 October 2017; pp. 2999–3007.

7. Baştan, M.; Yousefi, M.R.; Breuel, T.M. Visual words on baggage X-ray images. In *Computer Analysis of Images and Patterns, Proceedings of the 2011 International Conference on Computer Analysis of Images and Patterns, Seville, Spain, 29–31 August 2011*; Berciano, A., Díaz-Pernil, D., Kropatsch, W., Molina-Abril, H., Real, P., Eds.; Springer: Berlin, Germany, 2011; pp. 360–368.
8. Turcsany, D.; Mouton, A.; Breckon, T.P. Improving feature-based object recognition for X-ray baggage security screening using primed visual words. In Proceedings of the 2013 IEEE International Conference on Industrial Technology, Portland, OR, USA, 25–28 February 2013; pp. 1140–1145.
9. Kundegorski, M.E.; Akçay, S.; Devereux, M.; Mouton, A.; Breckon, T.P. On using feature descriptors as visual words for object detection within X-ray baggage security screening. In Proceedings of the 7th International Conference on Imaging for Crime Detection and Prevention, London, UK, 23–25 November 2016; pp. 1–6.
10. Hearst, M.A.; Dumais, S.T.; Osuna, E.; Platt, J.; Scholkopf, B. Support vector machines. *IEEE Intell. Syst. Appl.* **1998**, *13*, 18–28. [CrossRef]
11. Akcay, S.; Breckon, T. Towards automatic threat detection: A survey of advances of deep learning within X-ray security imaging. *arXiv* **2020**, arXiv:2001.01293.
12. Mery, D.; Svec, E.; Arias, M. Object recognition in baggage inspection using adaptive sparse representations of X-ray images. In *Image and Video Technology, Proceedings of the 2015 Pacific-Rim Symposium on Image and Video Technology, Auckland, New Zealand, 25–27 November 2015*; Bräunl, T., McCane, B., Rivera, M., Yu, X., Eds.; Springer: Berlin/Heidelberg, Germany, 2015; pp. 709–720.
13. Kwon, H.; Yoon, H.; Park, K.W. Multi-targeted backdoor: Indentifying backdoor attack for multiple deep neural networks. *IEICE Trans. Inf. Syst.* **2020**, *103*, 883–887. [CrossRef]
14. Mery, D.; Riffo, V.; Zscherpel, U.; Mondragón, G.; Lillo, I.; Zuccar, I.; Lobel, H.; Carrasco, M. GDXray: The database of X-ray images for nondestructive testing. *J. Nondestruct. Eval.* **2015**, *34*, 1–12. [CrossRef]
15. Akcay, S.; Kundegorski, M.E.; Willcocks, C.G.; Breckon, T.P. Using Deep Convolutional Neural Network Architectures for Object Classification and Detection Within X-Ray Baggage Security Imagery. *IEEE Trans. Inf. Forensics Secur.* **2018**, *13*, 2203–2215. [CrossRef]
16. Dhiraj; Jain, D.K. An evaluation of deep learning-based object detection strategies for threat object detection in baggage security imagery. *Pattern Recognit. Lett.* **2019**, *120*, 112–119. [CrossRef]
17. Ding, J.W.; Chen, S.Y.; Lu, G.R. X-ray security inspection method using active vision based on Q-learning algorithm. *J. Comput. Appl.* **2018**, *38*, 3414–3418.
18. Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; Li, F.-F. ImageNet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
19. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Region-Based Convolutional Networks for Accurate Object Detection and Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *38*, 142–158. [CrossRef] [PubMed]
20. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 779–788.
21. Miao, C.; Xie, L.; Wan, F.; Su, C.; Liu, H.; Jiao, J.; Ye, Q. SIXray: A large-scale security inspection X-ray benchmark for prohibited item discovery in overlapping images. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 2114–2123.
22. Rong, T.; Cai, H.; Xiong, Y. An Enhanced Prohibited Items Recognition Model. *arXiv* **2021**, arXiv:2102.12256.
23. Hu, B.; Zhang, C.; Wang, L.; Zhang, Q.; Liu, Y. Multi-label X-ray Imagery Classification via Bottom-up Attention and Meta Fusion. In Proceedings of the Asian Conference on Computer Vision, Tokyo, Japan, 30 November–4 December 2020.
24. Hassan, T.; Shafay, M.; Akçay, S.; Khan, S.; Bennamoun, M.; Damiani, E.; Werghi, N. Meta-Transfer Learning Driven Tensor-Shot Detector for the Autonomous Localization and Recognition of Concealed Baggage Threats. *Sensors* **2020**, *20*, 6450. [CrossRef] [PubMed]
25. Dumagpi, J.K.; Jeong, Y.J. Evaluating GAN-Based Image Augmentation for Threat Detection in Large-Scale Xray Security Images. *Appl. Sci.* **2021**, *11*, 36. [CrossRef]
26. Wei, Y.; Tao, R.; Wu, Z.; Ma, Y.; Zhang, L.; Liu, X. 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, Seattle, WA, USA, 12–16 October 2020; pp. 138–146.
27. Chaib, S.; Liu, H.; Gu, Y.; Yao, H. Deep Feature Fusion for VHR Remote Sensing Scene Classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 4775–4784. [CrossRef]
28. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single shot multibox detector. In Proceedings of the 2016 European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 21–37.
29. Eigen, D.; Fergus, R. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In Proceedings of the 2015 IEEE International Conference on Computer Vision, Boston, MA, USA, 7–13 December 2015; pp. 2650–2658.
30. Li, Z.; Zhou, F. FSSD: Feature fusion single shot multibox detector. *arXiv* **2017**, arXiv:1712.00960.
31. Zhang, Z.; Zhang, X.; Peng, C.; Xue, X.; Sun, J. ExFuse: Enhancing feature fusion for semantic segmentation. In Proceedings of the 2018 European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 273–288.
32. Wei, X.N.; Li, Y.H.; Wang, Z.Y.; Li, H.Z.; Wang, H.Z. Methods of training data augmentation for medical image artificial intelligence aided diagnosis. *J. Comput. Appl.* **2019**, *39*, 2558–2567.

33. Shrivastava, A.; Gupta, A.; Girshick, R. Training region-based object detectors with online hard example mining. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 761–769.
34. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
35. Zeiler, M.D.; Krishnan, D.; Taylor, G.W.; Fergus, R. Deconvolutional networks. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; pp. 2528–2535.
36. Shi, W.; Caballero, J.; Huszár, F.; Totz, J.; Aitken, A.P.; Bishop, R.; Rueckert, D.; Wang, Z. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1874–1883.
37. Bottou, L. Large-scale machine learning with stochastic gradient descent. In *COMPSTAT'2010, Proceedings of the 19th International Conference on Computational Statistics, Paris, France, 22–27 August 2010*; Physica-Verlag HD: Berlin/Heidelberg, Germany, 2010; pp. 177–186.
38. Everingham, M.; Gool, L.V.; Williams, C.K.I.; Winn, J.; Zisserman, A. The pascal Visual Object Classes (VOC) challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]
39. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.