Neural Computing and Applications (2020) 32:8711–8724
https://doi.org/10.1007/s00521-019-04360-0

ORIGINAL ARTICLE

Dangerous goods detection based on transfer learning in X-ray images

Yuanxi Wei[1] · Xiaoping Liu[1]

Received: 16 August 2018 / Accepted: 19 July 2019 / Published online: 30 July 2019
© Springer-Verlag London Ltd., part of Springer Nature 2019

Abstract
Computer vision technology is used to analyze X-ray images and detect dangerous goods in the process of logistics and express delivery. It is a security technology which can reduce labor strength and improve working efficiency. At present, there are many excellent detection models and methods in the field of object detection for visible light images, such as R-CNN, Fast R-CNN, Faster R-CNN, YOLO, SSD. These deep neural network-based detection methods achieved excellent performance on ImageNet. The training of object detection models on X-ray image datasets for dangerous goods detection is the focus of research in the field. Due to practical reasons, it is difficult to collect a comprehensive image dataset of dangerous goods (positive samples). In order to overcome this problem, this paper uses a multi-task transfer learning method on the basis of classification task and location search task on SSD network. The research in this paper focuses on adding additional convolutional layers in the SSD network to re-learn the knowledge of the model learned from the source domain. Experiments show that compared with the traditional method of fine-tuning, this method has better transfer learning ability on SSD network. This method was used to perform experiments in SSD300 on the image datasets screened from GDXray and achieved a mean average precision (mAP) of 0.915.

Keywords Transfer learning · Convolutional neural networks · SSD · X-ray security detection

1 Introduction

With the rapid development of the logistics and express delivery industry in recent years, it has become increasingly common for people to mail parcels. Express parcels are relatively closed, dangerous goods such as guns, daggers, bombs which can be easily hided in them, so it brings a safety risk that cannot be ignored. At present, it is a common practice to check various types of express parcels with manual inspection. This is achieved by arranging X-ray security inspection machines at important parts of express logistics. However, there are many limitations in the manual inspection. For example, inspectors will get bored after a long time of work, because most of the parcels are normal and only a small number of parcels contain dangerous goods [27]; for another example, if suspected dangerous goods are found in the inspection process, it is necessary to make a correct judgment in a short time [5] which also poses a challenge to the inspectors.

X-ray image-based detection of dangerous goods is the use of modern computer vision technology to detect and mark image objects in the X-ray image space. This detection has some similar problems as the visible light image object detection has, such as perspective imaging, geometric distortion, pose problems, self-occlusion and noise. Therefore, the object detection method of computer vision technology based on natural light images can be applied to the detection of dangerous goods in X-ray images [27].

In recent years, the application of computer vision technology based on X-ray images has made great progress. Adaptive sparse representations (XASR+) [26] were used to identify dangerous goods in X-ray images. In the 15 training samples, the fuzzy KNN classifier was trained by using the shape context descriptor [4] and Zernike moments [15] features to detect pistols in X-ray images

✉ Yuanxi Wei
weiyuanxi@bupt.edu.cn

Xiaoping Liu
liuxp@bupt.edu.cn

[1] Automation School, Beijing University of Posts and Telecommunications, No. 10, Xitucheng Road, Haidian District, Beijing 100876, People's Republic of China

[34]. Zhang and Zhu [46], Uroukov and Speller [41] used pseudo-color, texture, edge, shape features, Gabor texture features, etc., to realize the identification of dangerous goods in X-ray images. Franzel et al. [9] proposed a method to detect dangerous goods in X-ray images on the basis of visual vocabularies and support vector machines (SVM) classifiers. Turcsany et al. [40] proposed a X-ray image dangerous goods identification method to classify dangerous goods with SVM on the basis of SURF [3] and bag of visual words. Mery [22], Riffo and Mery [33] proposed a visual vocabulary, Implicit Shape Model (ISM) model to achieve X-ray image detection of dangerous goods. Mery and Riffo [24] used X-ray image LBP, SIFT and SURF features under multiple X-ray views to achieve dangerous goods detection. The above methods have successfully explored the application of computer vision technology in the identification and detection of dangerous goods in X-ray images. The use of various artificial features such as pseudo-color, texture, edge, shape features, LBP, SIFT and SURF introduced in methods, as well as the use of various classification models such as KNN, SVM, ISM, BOWs, has achieved considerable results.

Mery et al. [27] proposed a method based on deep learning to identify dangerous items in X-ray images. This method extracts depth features of X-ray images by training AlexNet [17] and GoogleNet [38]. Mery et al. [27] did not directly use the output of the fully connected layer as the classification result, but use the simple nearest neighbor classifier (KNN) to identify dangerous goods, thus avoiding over-fitting due to insufficient training sets. Vardhan and Priyadarsini [42] achieved the classification of dangerous goods in X-ray images by fine-tuning the weight parameters of AlexNet and GoogleNet in their own dataset (6997 X-ray images). This method realized the end-to-end identification of multiple types of dangerous goods based on deep learning.

In the above method, the depth features of the X-ray image are extracted through a convolutional neural network so as to realize the classification of dangerous goods. In particular, Vardhan and Priyadarsini [42] realized the end-to-end multi-class identification of dangerous goods, thereby avoiding the multistage redundant work such as manual feature extraction and classifier training of traditional machine learning methods. However, the above work is only applicable to the classification of dangerous goods by CNN.

Models such as Faster R-CNN [32], YOLO [30] and SSD [21] can realize the end to end of the natural light image and calculate the position and classification of the interested target at the same time. These methods are effectively applied in various fields. Zhang et al. [44, 45] use the mixture of multi-scale deformable part model [8], Faster R-CNN, SSD, etc. to achieve the detection of the clothes of the characters in the video. Akcay et al. [2] used the fine-tuning training method to transfer the deep neural network learned in the ImageNet [35] training set to their own "Dbp6" dataset to achieve the purpose of classifying and detecting X-ray image dangerous goods. They compared the training results of R-CNN [10], Faster R-CNN, YOLO, etc., respectively, and achieved mean average precision (mAP) of 88.5 on the dataset.

In this paper, the main work is to apply the deep learning model based on convolutional neural network to the detection and identification of dangerous goods in X-ray images. The contribution of this paper mainly involves the following two aspects:

1. When constructing the transfer model, we introduce the DenseNet [13] into the SSD300 network to further correct the knowledge of the model learned from the source domain (natural light image datasets), thus enhancing the ability of transfer learning in the target domain(X-ray image datasets).
2. In the process of model training, we use the SSD300 model weight parameters learned from the source domain (natural light image datasets) to initialize the weight parameters of the transfer network in the target domain (X-ray image datasets); then, only a part of the network weight parameters of the target domain are updated, thereby improving the convergence rate of the model.

Finally, this method is applied to a relatively small training dataset GDXray [25] for training and testing; this model can quickly converge and obtain mean average precision (mAP) of 0.915

2 Transfer learning

Although the X-ray image and the natural light image have a large commonality in the object detection, there are certain differences between the X-ray image and the natural light image in terms of image space:

1. X-ray images are X-rays that penetrate the object and radiate to the optics to image [23], while natural light images are imaged by reflection or refraction of light from the surface of the object to the light sensing device [14].
2. The X-ray image is an image formed by superimposing the shadow of the transparent layer over the shadow of the opaque layer inside the object to be observed; it obtains the internal structure of the observed object by nondestructively transmission imaging. The natural light image obtains the surface image of the observed object by natural light reflection. In other words,

internal objects with a higher density of texture form a shadow due to the absorption of more energy, while internal objects with a lower density of texture form a transparent layer due to the penetration of X-rays [11].

In addition, the specific object detection tasks in the two graphics spaces are also different, because the natural light images and X-ray images contain different types of objects, such as Pascal VOC [6, 37] and GDXray [25]. In practice, the trained model under natural light image cannot be directly applied to the X-ray image, so a new model which can be directly trained in the X-ray image space or a transfer learning method which can train the object detection model under the X-ray image is required.

In practical, it is difficult to find sufficient X-ray image dataset to learn, so that the model cannot be closed to the real expression of the X-ray image features of the dangerous goods (positive samples). However, the training datasets of the visible light image are relatively easy to download from the Internet, such as ImageNet, Caltech101 [19], Pascal VOC, COCO [20]. In order to achieve a better detection effect of dangerous goods, this paper uses a transfer learning method to train the model.

2.1 Transfer learning applied on neural networks

Pan and Yang [29] pointed out that transfer learning can be divided into three categories: inductive transfer learning, transductive transfer learning and unsupervised transfer learning. Inductive transfer learning mainly refers to the target task which is different from the source task, no matter whether the source and target domains are the same or not. For example, Lee et al. [18] proposed a convex optimization algorithm for concurrent learning meta-priors and feature weights from an ensemble of related prediction tasks. The meta-priors can be transferred among different tasks.

Yosinski et al. [43] pointed out that CNN features learned in the first layer are generic and similar across multiple tasks. These features become more and more task specific in the deeper layers. The authors also discuss the differential impact of source CNNs on the target task. This research present an empirical understanding of the impact of transfer features learned in different CNN layers. Hinton and Salakhutdinov [12] pointed out that fine-tuning remains the method of choice for transfer learning with neural networks: a model is pretrained on a source domain (where data is often abundant), the output layers of the model are adapted to the target domain, and the network is fine-tuned via back propagation. Oquab et al. [28] transferred the AlexNet model to different tasks of Pascal VOC 2007 and 2012 datasets by reusing the network layer trained on ImageNet. This transfer learning method provided significant improvement on the target task and has been utilized in different applications [1, 7, 39]. Rusu et al. [36] proposed progressive networks to implement continual learning, thus avoiding some of the drawbacks brought by fine-tuning: It is difficult to predict in advance how the target task is related to one task of source multi-task in multi-task learning, and the fine-tuning can cause damage to the knowledge acquired in the source domain, which can result in catastrophic forgetting.

2.2 Network structure of transfer layers

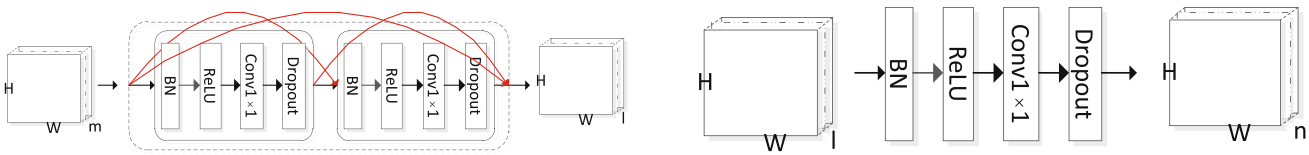
The research goal of this paper is to inductively transfer the object detection of visible light image to the dangerous goods detection of X-ray image, which is a typical multi-task learning (object location search task and object classification task). On the basis of the characteristics of the dangerous goods detection tasks involved in this paper, a knowledge transfer learning method of classification task and bounding box regression task is proposed based on SSD Network.

From the analysis in the previous discussion of the section, images in source and target domains involved in this paper are quite different, so the knowledge learned from the source domain cannot be directly applied to the model of target domain. The traditional solution is to initialize the target domain model with the model weight parameters trained in the source domain and then fine-tune the initialization weight parameters of the target domain model through training. Such a transfer learning method can achieve very good learning results. But experiments show that, based on the situation involved in this paper, it is necessary to train all the weight parameters of the model, and it needs more iterations and more computing resources.

This paper proposes to add additional convolution layers as transfer layers based on the source domain network to calibrate the difference between the source domain and the target domain, and the difference between the corresponding tasks of source domain and target domain. Therefore, the method can train only a small part of the weight parameters of the target domain model and all the weight parameters of the transfer layers, as a result, better transfer learning effects are achieved.

Huang et al. [13] proposed DenseNet; it is a densely connected convolutional neural network with excellent feature extraction and representation capabilities. This paper introduces DenseNet's network idea into the transfer layer network structure, as shown in Fig. 1.

We introduced two basic structures of DenseNet: DenseBlock and Transition Layers. The two structures are described as follows:



(a) DenseBlock structure

(b) Transition Layers structure

Fig. 1 DenseNet structure of the transfer layers

1. In order to avoid the destruction of the area information contained in the feature map learned by the network model in the source domain, we use a 1×1 convolution kernel for dense connection and adopt a step size of 1. Thus, the size of the input and output feature maps for each DenseBlock is $(H \times W)$. The forward propagation order of each convolution is BN $\rightarrow$ Relu $\rightarrow$ Conv1 $\times$ 1 $\rightarrow$ Dropout. Each DenseBlock contains two convolutional layers.
2. The forward propagation order of Transition Layers is the same as that of DenseBlock. Also based on the above-mentioned regional information learned in the source domain without destroying the model, we did not use the pooling technique to reduce the dimension at this layer, but changed the dimension of the feature map by changing the channel information of the convolution of 1×1.

We added DenseBlock and Transition Layers to the original deep neural network and combined these two types of structures to get DenseNet, and this structure is used as the network structure of the transfer layers. This structure does not change the size information of the input feature map, that is, it does not change the vision information of the object in the input feature map.

3 Detection method

At present, there are many object detection methods based on deep neural networks. We studied representative classical method such as Faster R-CNN, YOLO and SSD. Faster R-CNN introduced RPN (Region Proposal Networks), whose core idea is to directly generate a region proposal by using a convolutional neural network. The method used in essence is a sliding window. This method is widely used at present, but the speed cannot meet the real-time requirement because the existence of an extra RPN would lead to an increase in the amount of calculation. YOLO converted the object detection task into a regression problem, greatly speeding up the detection progress, enabling YOLO to process 45 images per second. YOLO does not have a region proposal mechanism, because it only uses 7×7 mesh regression; in particular, small targets cannot be positioned very accurately, which leads to the detection accuracy of this method not as high as Faster R-CNN.

SSD combines the regression idea in YOLO with the anchor mechanism in Faster R-CNN to regress multi-scale regional features at various locations of the full graph. The network structure is based on VGG16, and the last two fully connected layers are changed into convolutional layers, and additional four convolutional layers are added. The output maps of six different convolutional layers are convoluted with two different convolution kernels of 3×3 and 1×1, and the resulting output is used to calculate the confidence of the classification. Each category box generates 21 category confidences. Another type of output is used to calculate the localization of the object, and each default box generates 4 coordinate values (x, y, w, h). This method not only maintains the high speed of YOLO, but also guarantees the same accuracy as Faster R-CNN. Compared with Faster R-CNN, SSD does not generate a process for region proposal, which improves the detection speed to some extent.

The items in each parcel or baggage are varied, and each item has a different size. In respect of dangerous goods, for example, the size of the pistol is apparently different from the razor blade, as shown in Fig. 2. In order to accurately detect dangerous goods, it is necessary to simultaneously detect objects which have a large difference in pixel area in the X-ray image.

The feature is extracted by using a convolutional neural network. As the hierarchy deepens, the feature extracted is more abstract. Since the downsampling (pooling) technique is added in the convolution process, the smaller the distance between the convolutional layer and network end point, the larger the field of view of the convolution kernel. The SSD performs object detection on different levels of feature maps, thereby facilitating simultaneous detection of dangerous goods which have large differences in size in X-ray images.

3.1 Network structure

Take the SSD300 as an example; the transfer learning technology is used to transfer the SSD300 network from the Pascal VOC, COCO datasets to the X-ray image space, so that the new detection network can realize dangerous

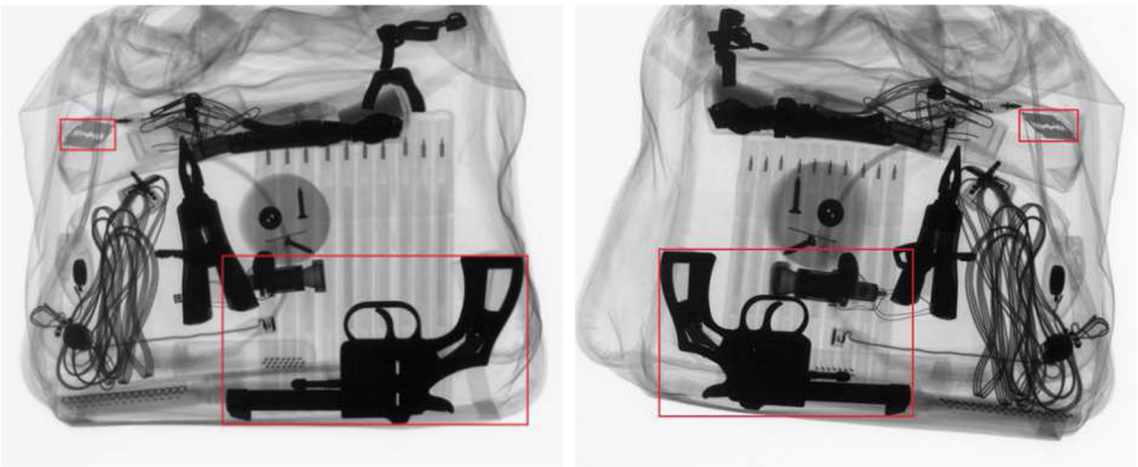


Fig. 2 X-ray images of dangerous goods such as pistols and razor blades in X-ray images

goods detection on X-ray images. As shown in Fig. 3, we add the alternate combination of DenseBlock and Transition Layers behind Block4, Block7, Block8, Block9, Block10 and Block11, respectively, and use them as the transfer layers.

As described in Sect. 2.2, the purpose of adding additional transfer layers to these Blocks is to calibrate the differences. Since SSD300 performs feature learning on different scales on convolutional neural networks, the degree of abstraction of features on the source domain is different at each scale. We chose a different number of alternate combinations of DenseBlock and Transition Layers as the transfer layers. Let the number of combinations be a, b, c, d, e, f, respectively, and we choose the number of combinations according to the different training parameters.

In order to retain the maximum knowledge that the model learned from the source domain, the added transfer layers do not break the dimensions of the input feature map. First, the convolution kernel of the transfer layer is 1×1, which does not change the visual field of the object of the input feature map. Second, the number of channels in the last transition of each DenseNet is equal to the number of channels in the input feature map.

3.2 Model training

The SSD network proposed by Liu et al. [21] in the natural light image space is trained, and the learned capability of object detection is transferred into the X-ray image space to realize X-ray image dangerous goods detection. The method of training the object detection model in the natural light image space is consistent with the training method of the SSD and therefore will not be repeated herein. On this basis, the weight parameters of transfer layer are trained in the X-ray image space. In this process, the related

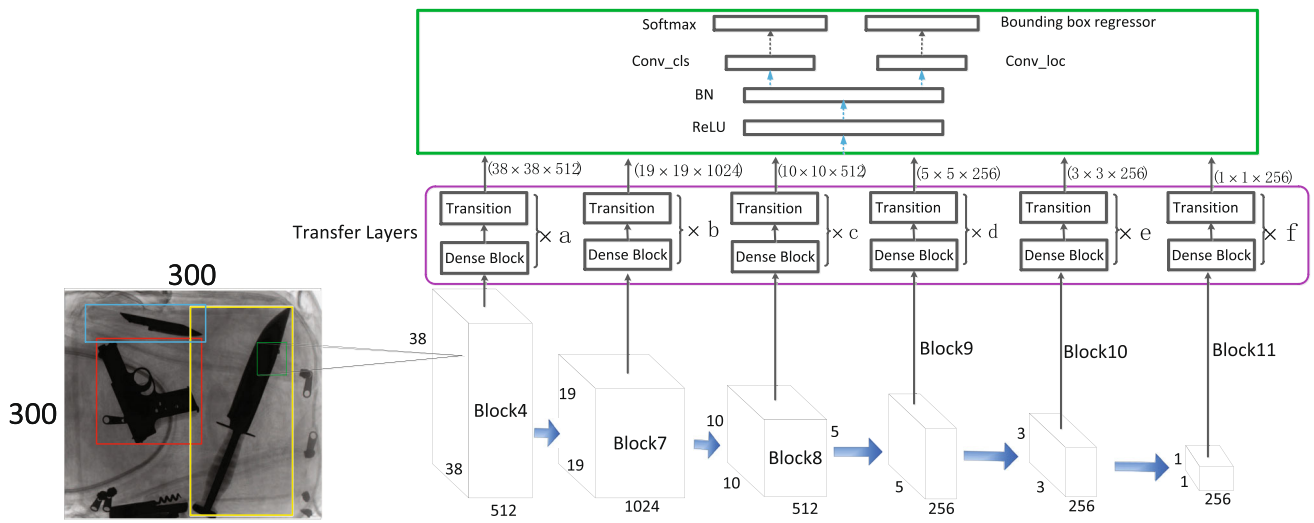


Fig. 3 Knowledge transfer network structure in SSD300

strategies proposed by Liu et al. [21], such as positive and negative sample matching strategy, default boxes selection strategy, positive and negative sample balance strategy are also used. In the transfer training process, the work of this paper focuses on the following three aspects:

3.2.1 Loss function calculation

The loss function in the transfer learning process is calculated in the same way as the loss function of the SSD network; only the transfer layer needs to be updated during the transfer learning process.

The model is a multitasking model for object classification prediction and object position prediction. From the network structure diagram of Fig. 3, the feature map output from the transfer layer enters two branches, respectively: One is to enter the softmax layer after the convolution layer of *Conv_cls* for class prediction of dangerous goods, and the other is to enter the bounding box regressor layer after the convolutional layer of *Conv_loc* for position prediction. Let the convolution kernel weight parameters of *Conv_cls* and *Conv_loc* be w_1 and w_2, respectively; let other weight parameters that need to be updated be w. We only update w_1, w_2 and w when making training in the target domain. The loss function is expressed as follows:

$$loss = \frac{cls(w_1, w, x) + a \cdot loc(w_2, w, l, g, x)}{N} \tag{1}$$

In Formula (1), *cls* represents the loss function of the class prediction, *loc* represents the loss function of the position prediction, and x represents the input feature map.

The other parameters are the same as those designed by Liu et al. [21]. l is predicted box, g is ground truth box, N is the number of matched default boxes, and when N is 0, *loss* is also equal to 0. a is the balance parameter and is set to 1.

It can be seen from Formula (1) that the knowledge acquired in the source domain is used to train the weight parameters of transferring layer in the target domain so as to achieve the purpose of transferring learning.

3.2.2 Data augmentation

In order to improve the robustness and generalization ability of the detection model, Liu et al. [21] proposed the perturbation strategy such as slicing strategy and horizontally flipped with probability of 0.5. On this basis, we completed the following work:

1. The items in a parcel in X-ray image have many forms, and the forms are much richer than the forms of common objects, such as bicycle, boat, bus, cow, dog in the image. The perturbation of the rotation transformation is added during the training, that is, the input image is randomly rotated within a range of 360°.
2. X-ray image is very different from visible light image; it does not have rich color information indicating the characteristics of the object like the visible light image. In order to improve the ability of the model to detect the gray color of the object, the image of the source dataset is copied and grayscale processed and then added it to the source training set for training to improve the robustness of the source model for detecting gray information.

3.2.3 Training process

We first add the transfer layer to the SSD300 model. When we start training, we use the SSD300 model weight parameters in the source domain to initialize the weight parameters of the target domain model except the transfer layer and the last layer. In the backpropagation, we update some weight parameters $(i > k)$ with the Adam [16] optimizer. The specific training process is shown in Algorithm 1.

During the training, we set the batch size (n) to 8 for each iteration. The parameters of the Adam [16] optimizer are set as follows: $\beta_1 = 0.9, \beta_2 = 0.999, \alpha = 0.001$. The learning rate (α) is exponentially decayed according to the decay rate of 0.94. In DenseBlock and Transition Layers, we used the dropout technique to set the output of the convolutional layers to 0 with a probability of 0.2.

Algorithm 1: Weight parameters training process in target domain

Initialization: s_i are the weight parameters of the i-th layer of SSD300 in source domain.

t_i are the initialization weight parameters of the i-th layer of the SSD300 in the target domain

z are initialization weight parameters of the transfer layers in the target domain

$t_i \leftarrow s_i$

set $w \in \{t_i, z | i > k\}$

for *number of training iterations* **do**

  Sample minibatch of n examples $\{x_1, x_2 ... x_n\}$ from the target domain.

  $loss = \frac{1}{n} \sum_{j=1}^{n} \frac{cls(w, x_j) + a \cdot loc(w, l, g, x_j)}{N_j}$

  $w \leftarrow Adam(loss)$

end

4 Experiments and analysis

The dataset used in the experiment is the GDXray.[1] The group of baggage of the dataset contains four dangerous goods: knife, handgun, shuriken and razor blade, a total of 8150 images. Part of these X-ray images were taken from different containers such as backpacks, pen cases, wallets. For example, series B0009–B0044, B0046–B0048 images are based on backpacks. There are also many multi-view images with no background, such as series B0007–B0008, B0049–B0054, B0075–B0077. In addition, B0055–B0059 and B0074 are cut off, and most of them contain no dangerous goods (positive sample) targets, so such kind of these images is discarded. Finally, the remaining images with and without the background were trained and tested. There are 1297 images with background and 2941 images with single object and no background. We randomly divided the dataset into five equal parts and used 80% of them for training and the remaining 20% for testing. In the experiments of this paper, we used cross-5-validation. The object distribution of the dataset is shown in Table 1.

Most of these objects are presented in multiple multi-view, which make the same kind of objects vary greatly in morphology, and different kinds of objects have smaller morphological differences.

As shown in Fig. 4a, b both are razor blade, but the difference in morphology is very large. (c) and (d) both are shuriken, and the difference between them is also very large. Although (a) and (c) belong to different categories, they are very similar in appearance. In the images with background, diversified shapes of object and occlusion problem make it difficult to distinguish one object from another visually even though the objects belong to the same type. The handgun in (f) has both occlusion and deformation problems, and it is also very different from the objects of the same category in (e), which greatly increases the difficulty of detection.

4.1 Convergence

First, the convergence of the SSD300 is examined. To illustrate the problem, the following two aspects are examined:

1. In order to enable the model to classify four types of dangerous goods, the convolutional layers of the object classification prediction after conv4 (Block4), conv7 (Block7), conv8 (Block8), conv9 (Block9), conv10 (Block10) and conv11 (Block11) of the SSD300 model are modified. The confidence is changed from 21 categories to 5 categories. The modified model is trained directly in the training dataset, and the relationship between the number of iterations and the loss function value is observed, as shown in Fig. 5a.
2. Using 120k iteration training on VOC2007, VOC2012 and COCO datasets to obtain the SSD300 model for transfer learning and using the transfer learning techniques proposed in this paper, as shown in Fig. 3, we add transfer layers to the SSD300 network and set the *abcdef* to 100000. When the weights are updated in training, we only update the convolutional layer after Block4 and fix the other convolution layer weights. We use the training datasets for transfer training and finally observe the relationship between the number of training iterations and the loss value, as shown in Fig. 5b.

Table 1 Dangerous goods distribution in Database

	Background		No background	
	Image	Object	Image	Object
Knife	128	131	2089	2089
Handgun	744	783	200	200
Shuriken	569	634	532	532
Razor blade	958	1025	120	120
Total	2399	2573	2941	2941

It is known from Formula (1) that the loss of the model includes the classification cross-entropy loss and the bounding box regression loss (represented by Locloss). Among them, the cross-entropy loss of the target classification includes positive sample loss (represented by Posloss) and negative sample loss (represented by Negloss), and the total loss is represented by Loss. The relationship between the number of iterations and the loss value is recorded. During the transfer process, positive sample loss (represented by Tposloss), negative sample loss (represented by Tnegloss), bounding box regression loss (represented by Tlocloss) and total loss (represented by Tloss) are also recorded.

For the first case, we did not use the transfer learning method to directly train the SSD300 model, and then, we observed 100000 iterations. From the perspective of Fig. 5a, its convergence is not very good. Although the Posloss value and the Locus value of the model have a downward trend, there is also a significant large-scale shock during the training iteration period. Although the Negloss value has fluctuated, it can converge to some extent after 65000 iterations. The above phenomenon occurs mainly due to the following reasons.

[1] GDXray is the GRIMA X-ray database, published by the Machine Intelligence Group at the Department of Computer Science of the Pontificia Universidad Catolica de Chile on http://dmery.ing.puc.cl/index.php/material/gdxray/.

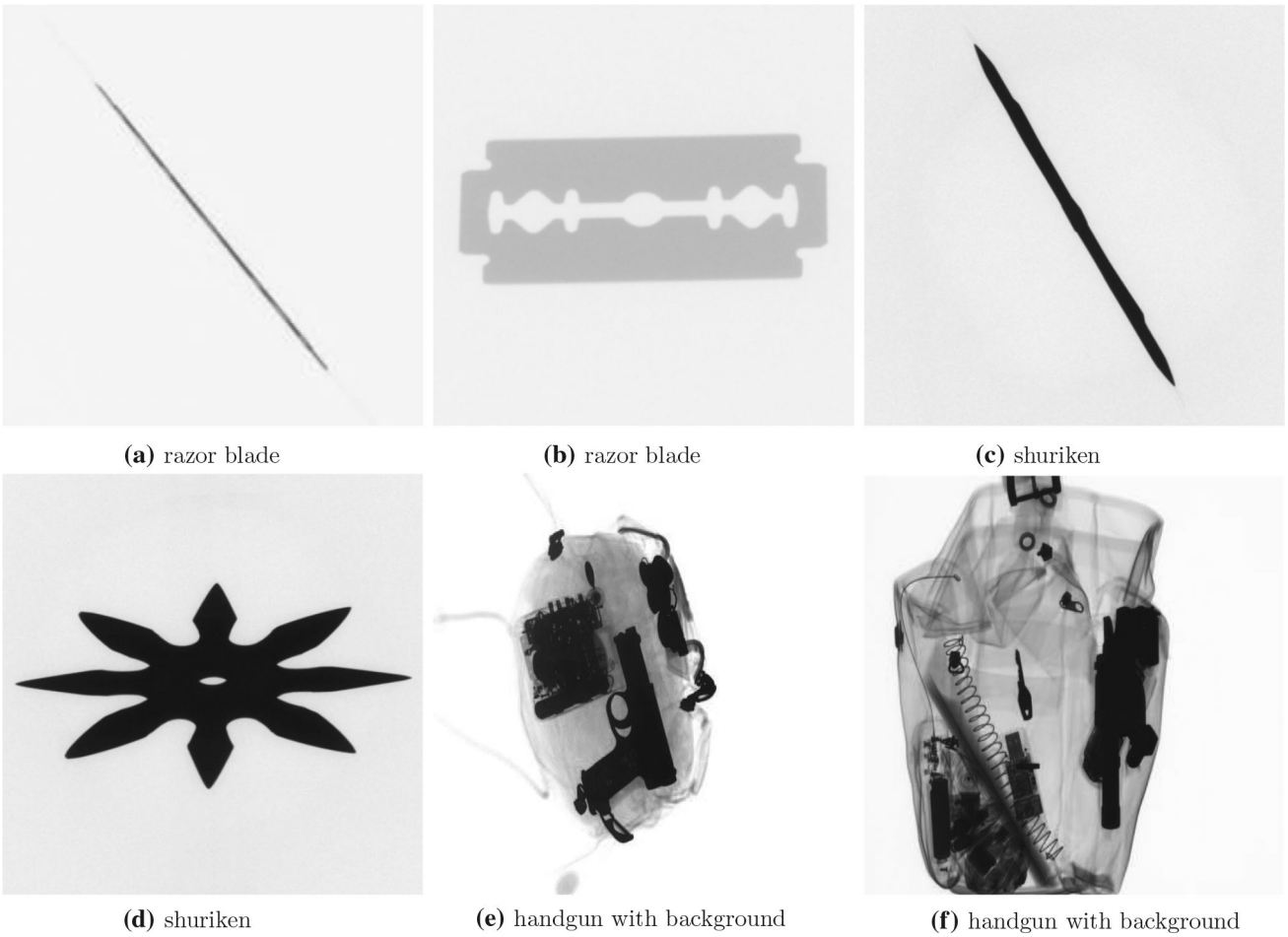


Fig. 4 Diversified shapes of object in X-ray images

On the one hand, the number of positive samples is too small relative to the negative samples, resulting in a better Negloss convergence effect than Posloss and Locus.

On the other hand, it can be seen from Fig. 4 that there is a case where the intra-class difference is larger than the inter-class difference in the positive sample, which also makes it difficult for the model to identify the category and position of the positive sample.

For the second case, we used the transfer learning method proposed in this paper to train the SSD300 model. We observed 30000 iterations. From Fig. 5b, we found that the model converges relatively well on the dataset. It can be seen from the total loss value (Tloss) that the model basically converges after 15000 iterations, wherein the loss curve is relatively smooth in contrast with Fig. 5a.

It can be seen that the transfer learning method proposed in this paper effectively overcomes the problems of insufficient training data of the target domain and difficulty in training of SSD300 model. This method has realized the training of end-to-end dangerous goods detection models in insufficient X-ray image dataset.

4.2 Detection performance of transfer learning

This section studies the impact of transfer learning on dangerous goods detection performance. Performance of the models is evaluated by mean average precision (mAP), used for PASCAL VOC object detection challenge [6]. For comparison, the average precision (AP) of objects such as knife, handgun, shuriken, razor blade is also calculated.

We take SSD300 as an example to experiment and analyze the model. Firstly, the weight parameters of the SSD300 model trained on the three datasets Pascal VOC2007, Pascal VOC2012 and COCO are fine-tuned so as to observe the influence of each layer weight parameters of the source domain on the target domain model. We start fine-tuning from the Block11 layer, that is, when we fine-tuning the Block11 and subsequent layer weight parameters in the training set, we fix the other layer weight parameters learned by the SSD300 network in the source domain. Therefore, as the level decreases, more and more weight parameters require fine-tuning. In the experiment, we also trained the SSD300 directly in our dataset without

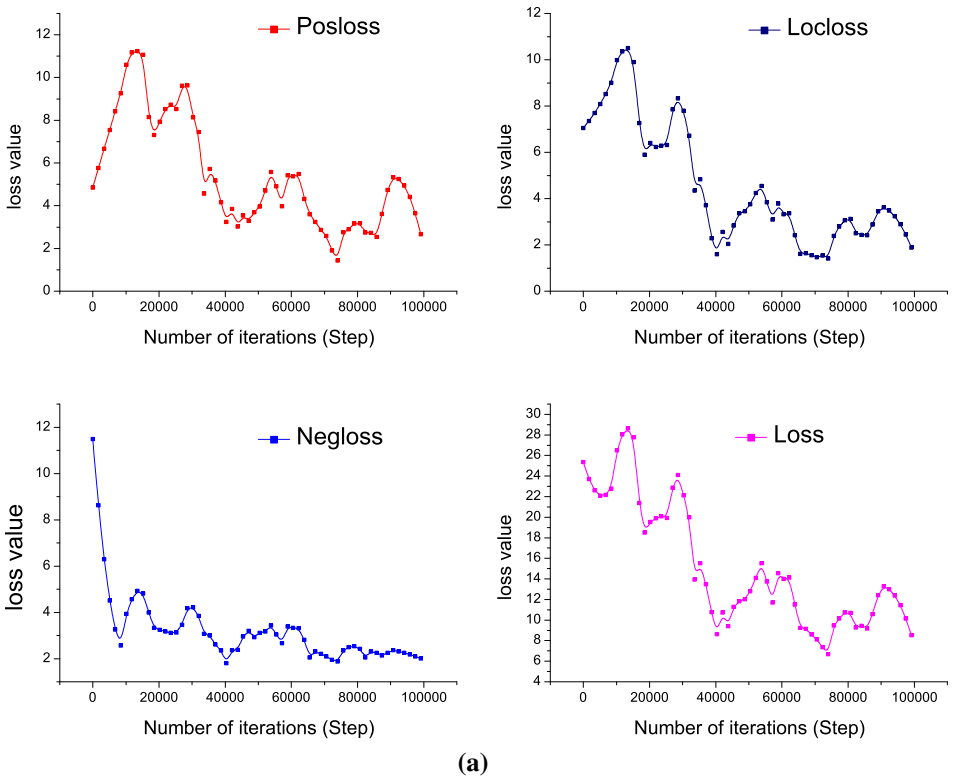


Fig. 5 Relationship between the number of model training iterations and the loss value

Table 2 SSD300 network model uses weight parameters with different levels to perform fine-tuning to obtain the model detection effect

Fine-tune layers	Iterations	Knife	Handgun	Shuriken	Razor blade	mAP
None[a]	100K	0.751	0.761	0.769	0.776	0.764
Block11	28K	0.668	0.693	0.671	0.701	0.683
Block10	30K	0.701	0.705	0.689	0.700	0.699
Block9	35K	0.720	0.705	0.686	0.740	0.713
Block8	45K	0.755	0.759	0.736	0.723	0.743
Block7	50K	0.801	0.782	0.799	0.787	0.792
Block4	50K	0.860	0.835	0.879	0.853	0.857
All[b]	90K	**0.891**	**0.905**	**0.887**	**0.904**	**0.897**

Bold values represent the best results

[a]"None" indicates that the SSD300 is directly trained in the target domain without using the transfer learning method

[b]"All" means to fine-tune all weight parameters of the SSD300 in the source domain

using the transfer learning method. The specific detection results are shown in Table 2.

Table 2 reveals the level of fine-tuning and corresponding detection performance such as the number of iterations required for network transfer training convergence, the average precision (AP) of each dangerous goods and mean average precision (mAP). Experiments show that when the SSD300 model is trained directly in the target domain without using the transfer learning method, we only get a mAP value of 0.764. In the case of transfer learning, the detection accuracy increases with the increase in weighting parameters to be updated. In this case, we need to increase the number of iterations required to train the SSD300 model in the target domain and consume more computing resources. Finally, we trained all the weight parameters and obtained a mAP value of 0.897.

From the discussion of the article [21], we know that SSD detects object in images from different scales. In the next, we discuss the impact of transfer layers network on the transfer learning capability at different scales of the SSD300 network. We set different values of *abcdef* in the experiment to observe the training convergence and detection accuracy.

We do not add any transfer layers, and the mAP value obtained by training the source domain SSD300 model weight in the target domain is 0.897. We set this value to be effective reference value (ERv) of the transfer learning method. When the mAP obtained by the SSD300 is larger than ERv, the setting of the transfer learning method is effective in this case.

In Table 2, we know that we can train the SSD300 network in GDXray at a small cost by fine-tuning the weight parameters of Block4 and subsequent layers while fixing the other layers weight parameters of the SSD300 network learning in the source domain, that is, only the weight parameter after Block 4 is updated in transfer training to obtain a mAP of 0.857. As mentioned above, we do not add any transfer layers, that is, set the value of *abcdef* to 000000. At this time, the mAP obtained is 0.857. We set this value as the basic reference value (BRv). When the mAP obtained by adding transfer layers is less than BRv, we regarded this situation as a failed case.

In the experiment, we take the SSD300 as an example and train the model according to the above settings. In the experiment, we recorded the number of iterations required for training convergence, as well as the AP of objects such as knife, handgun, shuriken, razor blade and mAP for the four categories of dangerous goods. The details are shown in Table 3.

In Table 3, we first set the values of *abcdef* to 333333, and we only got a mAP of 0.850. This value is lower than the BRv of 0.857. Starting from this setup, we gradually reduce the value of *fedcba*, and we find that the mAP value is increasing. Therefore, we summarize the following three points:

1. In the transfer training process, the more the transfer layers are added near Block11 of the SSD300 network, the greater the negative effect on the detection of dangerous goods.
2. The more the number of total transfer layers is increased, the worse the influence on the detection effect of the model.
3. In the transfer training process, adding transfer layers close to Block4 of the SSD300 network can effectively improve the detection performance, but when the transfer layers are added to a certain extent, it in turn affects the detection performance.

Although we added dropout to the transfer layers network, this reduces the complexity of the model to a certain extent and reduces the over-fitting phenomenon. But as we continue to add transfer layers, the complexity of overall model inevitably increases. Therefore, for smaller datasets,

Table 3 Detection performance of SSD300 model using a different number of transfer layers

abcde f	Iterations	Knife	Handgun	Shuriken	Razor blade	mAP
333333	20K	0.839	0.855	0.856	0.849	0.850
332211	20K	0.870	0.871	0.859	0.863	0.866
322110	15K	0.851	0.875	0.891	0.896	0.878
221100	15K	0.890	0.891	0.885	0.883	0.887
110000	15K	0.891	0.895	0.881	0.890	0.889
100000	15K	**0.910**	**0.916**	**0.918**	**0.915**	**0.915**
000000	50K	0.860	0.835	0.879	0.853	0.857

Bold values represent the best results

Table 4 Negative impact of transfer layers on detection performance of SSD300 model

abcde f	Iterations	Knife	Handgun	Shuriken	Razor blade	mAP
000001	15K	**0.869**	**0.881**	**0.883**	0.880	**0.878**
000012	15K	0.866	0.868	0.870	0.876	0.870
001122	15K	0.851	0.875	0.866	**0.885**	0.869
002233	15K	0.851	0.835	0.835	0.835	0.839
003333	20K	0.828	0.846	0.850	0.826	0.838
000000	50K	0.860	0.835	0.879	0.853	0.857

Bold values represent the best results

the over-fitting of the SSD300 model increases and detection performance of the SSD300 model reduces.

Through experiments, we found that on the SSD300, when we set the value of *abcdef* to 100000, during the training process, after 15000 iterations, the loss value stabilized. At this point, we obtained the highest mAP (0.915) through testing in the validation dataset.

At the same time, we fixed $a = b = 0$ to observe the impact of adding transfer layers on Block8, Block9, Block10 and Block11 on the transfer learning effect. In the experiment, we still use the SSD300 as an example to train the model according to the above settings. In the experiment, we recorded the number of iterations required for training convergence, as well as the AP of objects such as knife, handgun, shuriken, razor blade and mAP for the four categories of dangerous goods. The details are shown in Table 4.

In Table 4, we add the transfer layers network from Block11, that is, first set $f = 1$ and other values to 0. Then, we increase the value of *fedc* until $c = d = e = f = 3$. We found that although we only added a transfer layer in Block11, the mAP we obtained is still lower than the mAP value when the *abcdef* is 100000. As we add more transfer layers, the detection performance obtained is getting lower and lower. Finally, when we set the *cdef* to 2233 and 3333, the mAP values are 0.839 and 0.838, which are much lower than BRv.

As can be seen from Fig. 3, as the input data propagate forward in the network, the width (W) and height (H) of the feature map become smaller. For example, the width $\times$ height of the feature map output from Block4, Block7, Block8, Block9, Block10 and Block 11 of SSD300 are (38×38), (19×19), (10×10), (5×5), (3×3), (1×1). We add transfer layers network with dropout on the smaller feature map. This increases the convergence difficulty of the transfer network to a certain extent. Therefore, adding a transfer layers network on a smaller feature map can have a negative impact on overall detection performance.

4.3 Comparison of detection methods

For comparison, other detection models are also trained on this dataset, and the mean average precision (mAP) of these models is recorded as detection performance. In order to verify the effectiveness of the transfer learning method proposed in this paper, we compared with SSD300 to other deep neural network-based detection methods which have little difference in detection ability in natural light image dataset experiments. We chose the classic models such as Faster R-CNN and YOLOv2 [31] for comparison.

On the SSD300 model, we adopted the transfer learning method proposed in this paper. The *abcdef* values to the SSD300 of transfer layers are 100000. In the weight update process, we only updated the network weight parameters after Block4. At the same time, we also compare the case where we fine-tuned all the weight parameters of SSD300 model learned from the source domain without adding transfer layers.[2] The specific settings and results of the test are shown in Tables 5 and 6.

[2] For the convenience of comparison, we are labeled it as SSD300 * in Tables 5 and 6.

Table 5 Transfer method and parameter settings of compared models

Model	Net	Source domain[b]	Transfer layer	Weights updated
Faster R-CNN	VGGNet-16	07 + 12 + COCO	None	Whole weights
YOLOv2 416*416	Darknet-23[a]	07 + 12	None	Whole weights
SSD300 *	VGGNet-16	07 + 12 + COCO	None	Whole weights
SSD300	VGGNet-16	07 + 12 + COCO	Added	Partial weights

[a]https://pjreddie.com/darknet/yolov2

[b]The models train the datasets used in the source domain. 07+12 means union of VOC2007 and VOC2012 *trainval*. 07 + 12 + COCO means first training on COCO *trainval35k* and then fine-tuning on 07 + 12

Table 6 Comparison of detection performance of models

Model	Knife	Handgun	Shuriken	Razor blade	mAP
Faster R-CNN	**0.912**	0.915	0.913	0.911	0.913
YOLOv2 416*416	0.902	0.906	0.882	0.900	0.898
SSD300 *	0.891	0.905	0.887	0.904	0.897
SSD300	0.910	**0.916**	**0.918**	**0.915**	**0.915**

Bold values represent the best results

As shown in Table 6, the transfer learning method proposed in this paper can train the SSD300 model on a smaller dataset. Compared with the method of directly fine-tuning the model of the source domain, the transfer learning method proposed in this paper can improve the mAP on the GDXray.

5 Conclusion

In conclusion, the SSD model of object detection based on the computer vision is introduced into the X-ray image space to realize dangerous goods detection. During the experiment, the filtered GDXray dataset with a total of 4238 images is used. These images are randomly divided into 5 equal parts for experiment and cross-5-validation. We found that the SSD300 model did not converge well on the training dataset and the experimental accuracy was relatively low.

In order to overcome this problem, the transfer learning method is used to train the model, so that the SSD300 network model can be effectively converged. Finally, experiments show that the transfer learning method is applied to the SSD300 model to obtain mean average precision (mAP) of 0.915

For the purpose of improving the model's ability to detect dangerous goods, this paper adopts a method of adding transfer layers in different levels of SSD300 network to improve the ability of target domain model to deal with the difference between source domain datasets and target domain datasets. Our research focuses on the use of transfer learning methods on the SSD300, and the use of transfer learning method on SSD512 is not involved. From the article [21], we found that SSD300 and SSD512 are based on VGGNet-16, but the depth of the SSD512 network is deeper and the input image size $(W \times H)$ is larger. Therefore, SSD512 is more complicated than the SSD300 model. SSD512 can get a better mAP than SSD300 when the dataset is large enough. According to the research in this paper, adding transfer layers to the SSD model can improve the model's ability to detect dangerous goods. However, the impact of additional transfer layers on model complexity cannot be ignored. Because, on the one hand, it increases the over-fitting phenomenon, reduces the mAP of the model on the verification dataset and affects the generalization ability of the model, on the other hand, additional computing resources have also been added. Therefore, we propose that when learning the depth model of relatively complex such as SSD512 on small datasets, we should pay attention to the use of dropout and other regularization items which can prevent over-fitting so as to control the complexity of the model. On the one hand, it is necessary to find an appropriate transfer layer to improve the ability of the model to deal with the difference between the source and target domain datasets.

In the future, we will study the additional transfer layer in other depth detection models to transfer the models trained in the visible light image-based datasets to the dangerous goods detection tasks of x-ray image-based datasets. We will further optimize this method and increase its scope of application. For x-ray-based dangerous goods detection, the relationship between detection accuracy, detection efficiency and computational cost input should be balanced. At present, this paper only studies the detection accuracy level of the model and does not consider the detection efficiency. In the future, we will further study how to increase the detection efficiency while ensuring accuracy.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Agrawal P, Carreira J, Malik J (2015) Learning to see by moving. In: 2015 IEEE international conference on computer vision (ICCV), pp 37–45
2. Akcay S, Kundegorski ME, Willcocks CG, Breckon TP (2018) Using deep convolutional neural network architectures for object classification and detection within X-ray baggage security imagery. IEEE Trans Inf Forensics Secur 13(9):2203–2215
3. Bay H, Ess A, Tuytelaars T, Gool LJV (2008) Speeded-up robust features (surf). Comput Vis Image Underst 110(3):346–359
4. Belongie S, Malik J, Puzicha J (2001) Shape context: a new descriptor for shape matching and object recognition. In: Advances in neural information processing systems 13: proceedings of the 2000 conference, pp 831–837
5. Blalock G, Kadiyali V, Simon DH (2007) The impact of post-9/11 airport security measures on the demand for air travel. J Law Econ 50(4):731–755
6. Everingham M, Gool LJV, Williams CKI, Winn JM, Zisserman A (2010) The Pascal visual object classes (VOC) challenge. Int J Comput Vis 88(2):303–338
7. Farfade SS, Saberian MJ, Li LJ (2015) Multi-view face detection using deep convolutional neural networks. In: ACM on international conference on multimedia retrieval, pp 643–650
8. Felzenszwalb PF, Girshick RB, Mcallester DA, Ramanan D (2010) Object detection with discriminatively trained part-based models. IEEE Trans Pattern Anal Mach Intell 32(9):1627–1645
9. Franzel T, Schmidt U, Roth S (2012) Object detection in multi-view X-ray images. In: Pinz A, Pock T, Bischof H, Leberl F (eds) DAGM/OAGM 2012. LNCS, vol 7476. Springer, Heidelberg, pp 144–154
10. Girshick RB, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR '14 proceedings of the 2014 IEEE conference on computer vision and pattern recognition, pp 580–587
11. Hay GA (1978) X-ray imaging. J Phys E 11(5):377–385
12. Hinton GE, Salakhutdinov RR (2006) Reducing the dimensionality of data with neural networks. Science 313(5786):504–507
13. Huang G, Liu Z, Der Maaten LV, Weinberger KQ (2017) Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4700–4708
14. Kazlauciunas A (2001) Digital imaging: theory and application. Part I: theory. Surf Coat Int Part B Coat Trans 84(1):1–9
15. Khotanzad A, Hong YH (1990) Invariant image recognition by Zernike moments. IEEE Trans Pattern Anal Mach Intell 12(5):489–497
16. Kingma DP, Ba J (2015) Adam: a method for stochastic optimization. In: International conference on learning representations
17. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: International conference on neural information processing systems, pp 1097–1105
18. Lee SI, Chatalbashev V, Vickrey D, Koller D (2007) Learning a meta-level prior for feature relevance from multiple related tasks. In: Proceedings of the 24th international conference on Machine learning, pp 489–496
19. Li FF, Fergus R, Perona P (2007) Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories. Comput Vis Image Underst 106(1):59–70
20. Lin TY, Maire M, Belongie SJ, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL (2014) Microsoft coco: Common objects in context. In: European conference on computer vision, pp 740–755
21. Liu W, Anguelov D, Erhan D, Szegedy C, Reed SE, Fu CY, Berg AC (2016) Ssd: Single shot multibox detector. In: European conference on computer vision, pp 21–37
22. Mery D (2013) X-ray testing by computer vision. In: 2013 IEEE conference on computer vision and pattern recognition workshops, pp 360–367
23. Mery D (2014) Computer vision technology for X-ray testing. Insight 56(3):147–155
24. Mery D, Riffo V (2014) Automated object recognition using multiple X-ray views. Mater Eval 72(11):1362–1372
25. Mery D, Riffo V, Zscherpel U, Mondragón G, Lillo I, Zuccar I, Lobel H, Carrasco M (2015) Gdxray: the database of X-ray images for nondestructive testing. J Nondestr Eval 34(4):42
26. Mery D, Svec E, Arias M (2015) Object recognition in baggage inspection using adaptive sparse representations of X-ray images. In: Pacific-rim symposium on image and video technology, pp 709–720
27. Mery D, Svec E, Arias M, Riffo V, Saavedra JM, Banerjee S (2017) Modern computer vision techniques for X-ray testing in baggage inspection. IEEE Trans Syst Man Cybern Syst 47(4):682–692
28. Oquab M, Bottou L, Laptev I, Sivic J (2014) Learning and transferring mid-level image representations using convolutional neural networks. In: CVPR '14 proceedings of the 2014 IEEE conference on computer vision and pattern recognition, pp 1717–1724
29. Pan SJ, Yang Q (2010) A survey on transfer learning. IEEE Trans Knowl Data Eng 22(10):1345–1359
30. Redmon J, Divvala SK, Girshick RB, Farhadi A (2016) You only look once: Unified, real-time object detection. In: 2016 IEEE conference on computer vision and pattern recognition (CVPR), pp 779–788
31. Redmon J, Farhadi A (2017) Yolo9000: better, faster, stronger. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 7263–7271
32. Ren S, He K, Girshick RB, Sun J (2017) Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans Pattern Anal Mach Intell 39(6):1137–1149
33. Riffo V, Mery D (2017) Automated detection of threat objects using adapted implicit shape model. IEEE Trans Syst Man Cybern Syst 46(4):472–482
34. Roomi MM (2012) Detection of concealed weapons in X-ray images using fuzzy k-NN. Int J Comput Sci Eng Inf 2(2):187–196
35. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M, Berg AC, Fei-Fei L (2015) Imagenet large scale visual recognition challenge. Int J Comput Vis 115(3):211–252
36. Rusu AA, Rabinowitz NC, Desjardins G, Soyer H, Kirkpatrick J, Kavukcuoglu K, Pascanu R, Hadsell R (2016) Progressive neural networks. arXiv:1606.04671
37. Shetty S (2016) Application of convolutional neural network for image classification on pascal VOC challenge 2012 dataset. arXiv:1607.03785
38. Szegedy C, Liu W, Jia Y, Sermanet P, Reed SE, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: 2015 IEEE conference on computer vision and pattern recognition (CVPR), pp 1–9
39. Tulsiani S, Carreira J, Malik J (2015) Pose induction for novel object categories. In: 2015 IEEE international conference on computer vision (ICCV), pp 64–72

40. Turcsany D, Mouton A, Breckon TP (2013) Improving feature-based object recognition for X-ray baggage security screening using primed visualwords. In: IEEE international conference on industrial technology, pp 1140–1145
41. Uroukov I, Speller R (2015) A preliminary approach to intelligent X-ray imaging for baggage inspection at airports. Signal Process Res 4(5):1–11
42. Vardhan PH, Priyadarsini PSU (2016) Transfer learning using convolutional neural networks for object classification within X-ray baggage security imagery. Res J Pharm Biol Chem Sci 7:222–229
43. Yosinski J, Clune J, Bengio Y, Lipson H (2014) How transferable are features in deep neural networks? In: Advances in neural information processing systems, pp 3320–3328
44. Zhang H, Cao X, Ho JKL, Chow TWS (2017) Object-level video advertising: an optimization framework. IEEE Trans Ind Inform 13(2):520–531
45. Zhang H, Ji Y, Huang W, Liu L (2018) Sitcom-star-based clothing retrieval for video advertising: a deep learning framework. Neural Comput Appl. https://doi.org/10.1007/s00521-018-3579-x
46. Zhang N, Zhu J (2015) A study of X-ray machine image local semantic features extraction model based on bag-of-words for airport security. Int J Smart Sens Intell Syst 8(1):45–64

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