

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/346868409>

Meta-Transfer Learning Driven Tensor-Shot Detector for the Autonomous Localization and Recognition of Concealed Baggage Threats

Article in Sensors · November 2020

DOI: 10.3390/s20226450

CITATIONS

53

READS

1,805

7 authors, including:



Taimur Hassan

Khalifa University

113 PUBLICATIONS 1,777 CITATIONS

[SEE PROFILE](#)



Muhammad Shafay

Khalifa University

14 PUBLICATIONS 193 CITATIONS

[SEE PROFILE](#)



Samet Akcay

Durham University

45 PUBLICATIONS 2,892 CITATIONS

[SEE PROFILE](#)



Salman Hameed Khan

Mohamed bin Zayed University of Artificial Intelligence

174 PUBLICATIONS 7,404 CITATIONS

[SEE PROFILE](#)

Article

Meta-Transfer Learning Driven Tensor-Shot Detector for the Autonomous Localization and Recognition of Concealed Baggage Threats

Taimur Hassan ^{1,*}, Muhammad Shafay ¹, Samet Akçay ², Salman Khan ³,
Mohammed Bennamoun ⁴ and Ernesto Damiani ¹ and Naoufel Werghi ¹

¹ Center for Cyber-Physical Systems, Khalifa University of Science and Technology, Abu Dhabi 127788, UAE; 100057573@ku.ac.ae (M.S.); ernesto.damiani@ku.ac.ae (E.D.); naoufel.werghi@ku.ac.ae (N.W.)

² Department of Computer Science, Durham University, Durham DH1 3DE, UK; samet.akcay@durham.ac.uk

³ Inception Institute of Artificial Intelligence, Abu Dhabi 127788, UAE; salmaneme@gmail.com

⁴ Department of Computer Science and Software Engineering, The University of Western Australia, Perth 6907, Australia; mohammed.bennamoun@uwa.edu.au

* Correspondence: taimur.hassan@ku.ac.ae

Received: 30 September 2020; Accepted: 3 November 2020; Published: 12 November 2020

Abstract: Screening baggage against potential threats has become one of the prime aviation security concerns all over the world, where manual detection of prohibited items is a time-consuming and hectic process. Many researchers have developed autonomous systems to recognize baggage threats using security X-ray scans. However, all of these frameworks are vulnerable against screening cluttered and concealed contraband items. Furthermore, to the best of our knowledge, no framework possesses the capacity to recognize baggage threats across multiple scanner specifications without an explicit retraining process. To overcome this, we present a novel meta-transfer learning-driven tensor-shot detector that decomposes the candidate scan into dual-energy tensors and employs a meta-one-shot classification backbone to recognize and localize the cluttered baggage threats. In addition, the proposed detection framework can be well-generalized to multiple scanner specifications due to its capacity to generate object proposals from the unified tensor maps rather than diversified raw scans. We have rigorously evaluated the proposed tensor-shot detector on the publicly available SIXray and GDXray datasets (containing a cumulative of 1,067,381 grayscale and colored baggage X-ray scans). On the SIXray dataset, the proposed framework achieved a mean average precision (*mAP*) of 0.6457, and on the GDXray dataset, it achieved the precision and *F*₁ score of 0.9441 and 0.9598, respectively. Furthermore, it outperforms state-of-the-art frameworks by 8.03% in terms of *mAP*, 1.49% in terms of precision, and 0.573% in terms of *F*₁ on the SIXray and GDXray dataset, respectively.

Keywords: aviation security; meta-transfer learning; one-shot learning; convolutional neural networks; structure tensors; X-ray imagery

1. Introduction

Baggage threat recognition has gained the utmost attention due to increased terrorist activities, especially in the last two decades. According to a recent survey, approximately 1.5 million passengers are screened every day against weaponry in the United States [1]. To identify baggage threats at the airport, malls, and cargoes, radiography is mainly used due to its reliability and cost-effectiveness [2]. In addition, many researchers have quantitatively measured the detection capacity of the security officers towards recognizing baggage threats through X-ray imagery via receiver operator characteristics (ROC) curve [3]. However, manual screening of baggage content

(within the X-ray scans) to identify potential threats is a time-consuming task [4]. Furthermore, it is vulnerable to human errors caused due to fatigued work schedules [5]. Although, researchers have reported the high capacity (and less false alarm rate) of sniffer dogs to detect suspicious items as compared to humans. However, sniffer dogs can only work for an hour or so before they need rest [6]. Here, due to the capacity of autonomous frameworks to mass screen contraband items, many people have encouraged their utilization [4]. In addition, they recommended manual supervision (towards screening baggage threats) as a second-level inspection scheme to filter their erroneous detections [7].

For detecting objects from the RGB scans, many people have proposed one-staged and two-staged object detectors that produce promising results. However, due to the inherent differences between the X-ray and the RGB scans, these object detectors do not work well for identifying the baggage threats (via X-ray imagery) [8–10], especially in extreme concealment and cluttered scenarios [5,11]. To overcome this, researchers have developed exclusive frameworks for detecting and classifying baggage threats from the X-ray scans [12–14]. These frameworks can well recognize the visible and partially occluded baggage threats from the X-ray scans [5,11,13]. However, they are still vulnerable towards recognizing the extremely cluttered, concealed, and occluded objects [5,9] like, for example, the *guns* in Figure 1A–E, the *knives* in Figure 1F–I, and the *wrenches* in Figure 1F.

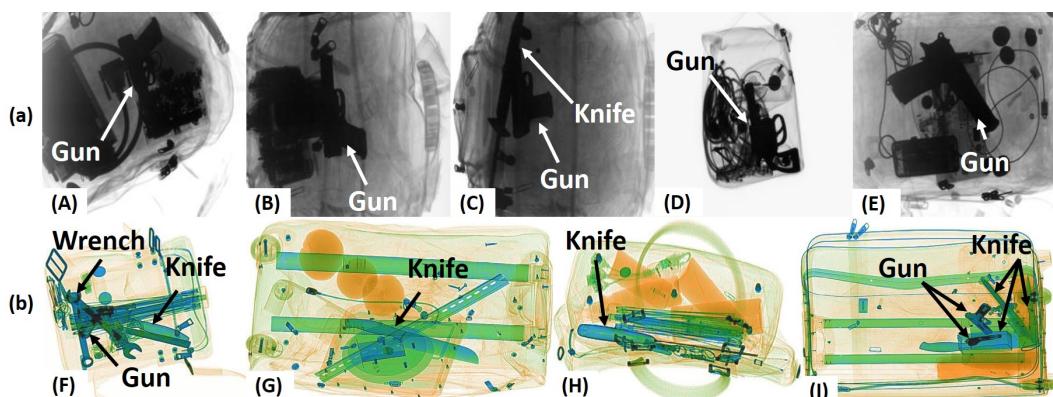


Figure 1. ((A)–(I)) Baggage X-ray scans containing extremely cluttered and overlapping contraband items such as *guns*, *knives* and *wrenches*. Row (a) shows the scans from the GDXray [15] dataset and row (b) shows the scans from the SIXray [14] dataset.

2. Related Work

Baggage threat detection has been a widely researched area where researchers initially employed conventional machine learning methods to recognize contraband items from the X-ray scan. Since the classical methods are based on hand-engineered features, they are confined to limited datasets and restricted experimental settings. More recently, deep learning has been employed for detecting prohibited items, outperforming traditional schemes in terms of accuracy, speed, and robustness. However, deep learning frameworks are still vulnerable to extreme occlusion, clutter, and diverse scanner specifications. Although, recent developments in recognizing baggage threats managed to address occlusion to some extent [5,13,14]. However, these frameworks are either tested on a single dataset [13,14] or they require extensive (parameter) tuning for different scanner specifications [5]. Furthermore, to the best of our knowledge, there is no mechanism (based on meta-learning [16] or meta-transfer learning [17]) to extend the capacity of these frameworks to generalize well across diverse ranging scanners without an explicit retraining process. In this section, we first shed light on some of the recent meta-learning (and meta-transfer learning [17]) frameworks, and then we discuss some of the popular frameworks for recognizing baggage threats. For an exhaustive survey on baggage threat recognition, we refer the readers to the work of [18–20].

2.1. Meta-Learning Frameworks

Meta-learning, also known as “learning to learn”, is a concept of extending the capacity of the deep neural networks to adapt (or generalize) to new tasks (or new domains) which have not been encountered during the training time. Essentially, the underlying network is given an exposure to learn from the large pool of experiences (during training), which they leverage on the set of unseen examples during the test time (via few-shot or zero-shot training). The major benefit of meta-learning over conventional transfer learning (or fine-tuning) approaches is that it allows the network to utilize its pretrained weights to effectively predict the unseen examples of the new underlying task without having to retrain on the large (and diverse) set of training examples for this current task to avoid overfitting [17]. Meta-learning has not only been employed for the supervised classification [16] and detection [21] tasks. It has also been used to acquire unlabeled data representation in an unsupervised manner [22]. More recently, Sun et al. [17] proposed a meta-transfer learning approach in which they transferred the pretrained weights of the deep neural networks for new tasks via few-shot learning where they achieved state-of-the-art performance on the benchmarked few-shot datasets such as miniImageNet [23] and Fewshot-CIFAR100 [24].

2.2. Traditional Machine Learning Methods

Initial solutions developed for baggage threat recognition involved classification [25], segmentation [26], and detection [27] strategies. While many of these schemes utilized SURF [28], and FAST-SURF [29] (coupled with Bag of Words), some of them also fused SIFT and SPIN features in conjunction with the Support Vector Machines (SVM) for classifying baggage threats from the multiview baggage imagery [8]. Moreover, Mery et al. proposed adaptive sparse representation [30] and adapted implicit shape model (AISM) [31] schemes for detecting prohibited baggage content. In another approach, they computed 3D feature points from the structure from motion to accurately recognize baggage threat from the X-ray imagery. In addition to this, Heitz et al. [26] proposed a region-growing technique coupled with SURF features to extract suspicious items from baggage X-ray scans.

2.3. Deep Learning Methods

Recently, researchers have developed deep learning methods for detecting prohibited items from the security X-ray scans. These methods have outperformed traditional approaches both in terms of robustness and efficiency. To increase readability, we have broadly categorized the deep learning methods (for screening baggage threats) as supervised and unsupervised approaches.

2.3.1. Supervised Approaches

The supervised deep learning methods to recognize baggage threats are based on object detection [32–34], classification [35,36], and segmentation [11,37] schemes. The majority of these methods also utilize one-staged [38] and two-staged [9] detectors such as YOLO [39], YOLOv2 [40], RetinaNet [41], and Faster R-CNN [42]. Moreover, Akçay et al. used GoogleNet [43] for classifying contraband items such as *cameras*, *laptops*, *guns*, *gun components*, and *knives* (particularly *ceramic knives*) from their local X-ray dataset scans. Xiao et al. [44] developed a computationally efficient variant of Faster R-CNN [42] (namely R-PCNN) for detecting prohibited items from TeraHertz (THz) imagery. R-PCNN takes around 150 min on average for training and around 16 µs on average for detecting objects. Gaus et al. [10] evaluated RetinaNet [41], Faster R-CNN [42] and Mask R-CNN [45] backboned through ResNet-18 [46], ResNet-50 [46], ResNet-101 [46], SqueezeNet [47], and VGG-16 [48] for screening baggage X-ray scans as benign or malignant [10]. Griffin et al. [36] classified unexpected items within the bagging areas based upon their shape, texture, and density, and semantic appearances. Moreover, Dhiraj et al. [33] evaluated the Faster R-CNN [42], YOLOv2 [40] and Tiny YOLO [40] to detect contraband items such as *shuriken*, *guns*, *knives*, and *razors* from publicly available GRIMA

X-ray Database (GDXray) [15]. Apart from this, Akçay et al. used AlexNet [49] as a features extractor coupled with SVM for baggage threat identification. Furthermore, they compared Faster R-CNN [42], sliding-window based CNN (SW-CNN), region-based fully convolutional networks (R-FCN) [50], and YOLOv2 [40] for recognizing occluded contraband items from the X-ray imagery. More recently, Wei et al. [13] proposed De-occlusion Attention Module (DOAM) module that can be integrated with the deep object detectors to recognize occluded threatening items. DOAM was thoroughly validated on a large-scale Occluded Prohibited Items X-ray (OPIXray) dataset, released publicly in [13]. Apart from this, Miao et al. [14] introduced one of the largest datasets for baggage threat detection, namely, Security Inspection X-ray (SIXray) dataset, containing extremely occluded and overlapping contraband items within highly imbalanced baggage X-ray scans. Furthermore, they proposed a framework, dubbed class-balanced hierarchical framework (CHR), to recognize contraband items such as *guns, knives, wrenches, pliers, and scissors* from different SIXray subsets indicating different levels of class imbalance [14]. SIXray dataset has been used by [51] in conjunction with their nonpublicly available dataset to analyze the transferability between different scanner specifications. Apart from this, we have also proposed a novel detection strategy, dubbed Cascaded Structure Tensor (CST), to recognize cluttered, occluded, and overlapping items from the SIXray [14] and GDXray [15] datasets.

2.3.2. Unsupervised Approaches

The majority of baggage screening systems employ supervised strategies to recognize threatening items. However, researchers have also used unsupervised approaches (particularly adversarial learning) to recognize baggage threats as anomalies. Akçay et al. pioneered this by first proposing encoder-decoder-encoder topology coupled with adversarial learning, termed GANomaly [52]. Afterward, they employed skip-connections, yielding Skip-GANomaly [53], to derive better latent representations to aid discriminator in accurately picking the threatening anomalies [53].

As evident from Table 1, baggage threat detection is an extensively researched area where researchers have proposed different classification, detection, and segmentation approaches to recognize prohibited items from the security X-ray scans. These frameworks, though, can autonomously recognize the concealed contraband items under low or partial occlusion, but they are limited towards recognizing highly cluttered (and occluded objects). Recently, some researchers have proposed frameworks that address the problem of occlusion to some extent [5,13,14]. However, either these methods are tested on a single dataset [13,14] or they require a lot of parameter tuning due to nonadaptability [5]. Furthermore, to the best of our knowledge, all of the existing works require an extensive amount of training data (for each scanner specifications) to perform acceptable results. Procuring such large-scale data for training is not feasible, limiting the deployment of such frameworks in the real world.

Table 1. Summary of the state-of-the-art baggage threat detection frameworks *.

Literature	Methodology	Performance	Limitations
Miao et al. [14]	Developed CHR [14], an imbalanced resistant framework that leverages reversed connections class-balanced loss function to effectively learn the imbalanced suspicious item categories in a highly imbalanced SIXray [14] dataset.	Achieved an overall mean average precision score of 0.793, 0.606, and 0.381 on SIXray10, SIXray100, and SIXray1000 [14], respectively when coupled with ResNet-101 [46] for recognizing five suspicious item categories.	Although the framework is resistant to an imbalanced dataset, it is still tested only on a single dataset.
Hassan et al. [11]	Proposed a contour instance segmentation framework for recognizing baggage threats regardless of the scanner specifications.	Achieved a mean average precision score of 0.4657 on a total of 223,686 multivendor baggage X-ray scans.	Built upon a conventional fine-tuning approach that requires a large-scale training dataset.
Gaus et al. [51]	Evaluated the transferability of different one-staged and two-staged object detection and instance segmentation models on SIXray10 [14] subset of the SIXray [14] dataset and also on their locally prepared dataset.	Achieved a mean average precision of 0.8500 for extracting <i>guns</i> and <i>knives</i> on SIXray10 [14] dataset.	Tested on only one public dataset i.e., the SIXray10 [14] for only extracting <i>guns</i> and <i>knives</i> .
Wei et al. [13]	Proposed a plug-and-play module dubbed DOAM [13] that can be integrated with the deep object detectors to recognize and localized the occluded threatening items.	Achieved the mean average precision score of 0.740 coupled with SSD [54].	DOAM [13] is not tested on publicly available GDXray [15] and SIXray [14] datasets.
Hassan et al. [5]	Developed a CST framework that leverages contours of the baggage content to generate object proposals that are screened via a single classification backbone.	Achieved a mean average precision score of 0.9343 and 0.9595 on GDXray [15] and SIXray [14] datasets.	CST, although, is tested on two public datasets, but it requires extensive parameter tuning to work well on both of them.

* For a detailed overview on the existing approaches, we refer the reader to the Supplementary Material of this article.

3. Contributions

This paper presents a novel meta-transfer learning-driven tensor-shot detector that recognizes the baggage threats in extremely cluttered, concealed, and occluded environment. Furthermore, due to its capacity to operate on the unified tensor maps rather than diverse raw scans, it can be well-generalized across multiple scanner specifications via pretrained weights and single-shot learning. Moreover, we rigorously evaluated the proposed framework on two (highly challenging) public datasets where it achieves state-of-the-art performance. To summarize, the major contributions of this paper are thus four-fold:

- A novel meta-transfer learning based single-shot detector capable of recognizing baggage threats under extreme occlusion.
- A highly generalizable detection framework that leverages the proposed dual-tensor scheme to localize and recognize the threatening items from the diverse ranging scans without retraining the backbone on the large set of examples.

- To the best of our knowledge, there is no generalized framework that leverages meta-transfer learning to autonomously recognize concealed baggage threats from the joint (combined) GDXray [15] and SIXray [14] datasets.
- The proposed tensor-shot detector has outperformed state-of-the-art frameworks by achieving 1.49% and 0.573% improvements over [33] in terms of precision and F1 scores on GDXray [15] dataset and 8.03% improvements (in terms of mean average precision) over [14] on SIXray [14] dataset.

The rest of the paper is organized as follows: Section 4 presents the proposed tensor-shot framework. Section 5 contains a detailed description of the datasets, training, and evaluation protocols. Section 6 contains the experimental results and comparison of the proposed framework against state-of-the-art solutions and Section 7 presents a detailed discussion on the proposed framework and also concludes the paper.

4. Proposed Framework

The block diagram of the proposed detection framework is shown in Figure 2. It is driven through a novel dual tensor mechanism that exploits the transitional variations of baggage items (with diversified spatial properties) by simultaneously generating the low and high energy tensor representation of the candidate scan. These tensors are then accumulated together and are passed through the edge suppression backbone which filters the irrelevant edge information and only retains the boundaries of the potential threatening items. These filtered edges are then postprocessed, upon which the bounding boxes (screened through nonmaximum suppression [55]) are fitted. Afterward, these bounding boxes are then used in crafting the object proposals which are further screened through the meta-one-shot classifier (driven through the edge suppression backbone).

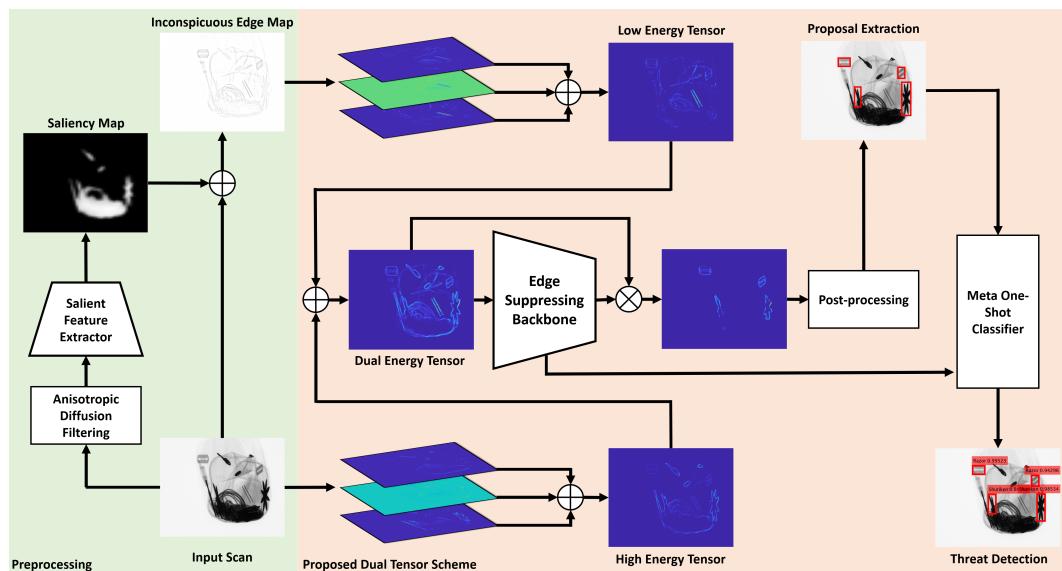


Figure 2. Block diagram of the proposed tensor-shot detector. First of all, we decompose the input scan into high and low energy tensors, where the high tensors are generated directly from the input scan (through the structure tensors [56]). Moreover, the low energy tensors are produced by first computing the salient image features (through the proposed feature extractor) and then accumulating them with the input scan. Both high and low energy tensors are then added together to produce a dual-energy tensor representation of the input scan that is then passed to the edge suppressing backbone to suppress the irrelevant baggage contours while simultaneously highlighting the threatening content. Then, the resultant baggage content is postprocessed, and for each extracted object, a bounding box is fitted to craft out its proposal that is passed to the meta-one-shot classifier for recognizing the object class.

4.1. Preprocessing

The input scan ξ_X is filtered through the anisotropic diffusion filter. Afterward, we generate the inconspicuous version of ξ_X to enhance the edges of the dulled baggage items.

Inconspicuous Edge Map Generation

To generate the inconspicuous edge map, we first compute the saliency map (representing set of salient features) through the proposed salient feature extractor, and then eliminate these representations from the original input scan to highlight the edges of the low contrast and low spectral baggage items. The saliency map here showcases the items having the higher spectral components within ξ_X , derived through the trainable edge-preserving kernels of the proposed salient feature extractor. Moreover, the architecture of the feature extractor is intentionally kept shallow by deploying only one input layer, three convolution layers, two batch normalization layers, three ReLUs, two max pooling layer, one lambda layer (for resizing) and one addition layer (as shown in Figure 3) having a total of 1601 learnable and 128 nonlearnable parameters. The reason for making the salient network shallower is to preserve the shape of the prominent objects (having higher spectral components) that are eliminated from ξ_X to retain the contours of the dulled items and also to avoid the generation of false edges.

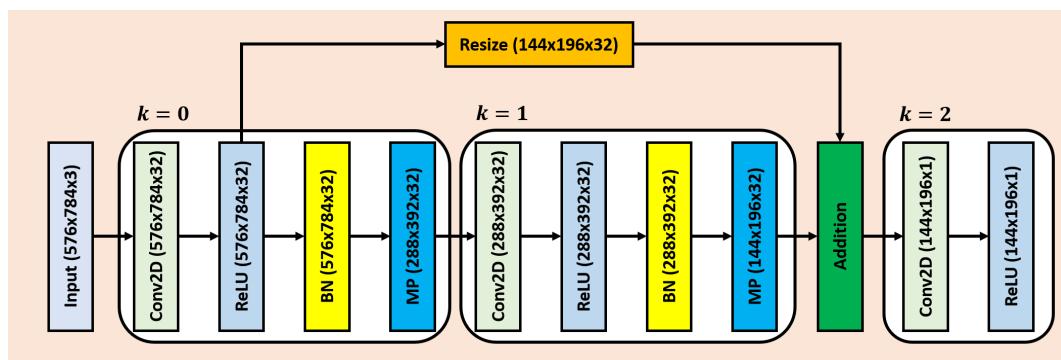


Figure 3. Salient network architecture. The abbreviations are Conv2D: 2D Convolution, BN: Batch Normalization, and MP: Max Pooling.

From Figure 3, we can also observe that the proposed salient feature extractor contains three salient blocks denoted by $k = 0, 1, 2$, wherein each block, the convolution and ReLU layers yields $f^k(x, y, z)$ and $f_r^k(x, y, z)$ of size $M_1 \times M_2 \times M_3$, respectively such that:

$$f^k(x, y, z) = \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \sum_{h=1}^{N_3} w_k(i, j) f_m^k \left(\left\lfloor \frac{x - i + 2p_1}{s_1} \right\rfloor + 1, \left\lfloor \frac{y - j + 2p_2}{s_2} \right\rfloor + 1, \left\lfloor \frac{z - h + 2p_3}{s_3} \right\rfloor + 1 \right) \quad (1)$$

and

$$f_r^k(x, y, z) = \begin{cases} f^k(x, y, z) & f^k(x, y, z) \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

w_k denotes the window of $N_1 \times N_2 \times N_3$ dimension (containing the trainable weights), $p_{1,2,3}$ denotes the input padding, $s_{1,2,3}$ denotes the stride rate, and f_m^k denotes the input feature maps. It should be noted here that for $k = 0$, $f_m^0(x, y, z) = \xi_X$, i.e., the input to the first convolution layer is the input scan ξ_X . Moreover, after extracting $f^k(x, y, z)$, it is normalized through the batch normalization layer as expressed below:

$$g^k(x, y, z) = \sum_{i=1}^{M_1} \sum_{j=1}^{M_2} \sum_{h=1}^{M_3} \frac{f_r^k(i, j, h) - \mu(f_r^k)}{\sqrt{\sigma(f_r^k)}} \quad (3)$$

where $\mu(f_r^k)$ and $\sigma(f_r^k)$ represent a mean and variance of the feature maps in k th block i.e., f_r^k , respectively. Then, $g^k(x, y, z)$ is passed through the max pooling layer, producing $g_m^k(x, y, z)$ such that:

$$g_m^k(x, y, z) = \max_{\Delta_x \in \frac{-K_1}{2}, \dots, \frac{K_1}{2}, \Delta_y \in \frac{-K_2}{2}, \dots, \frac{K_2}{2}, \Delta_z \in \frac{-K_3}{2}, \dots, \frac{K_3}{2}} g^k(x + \Delta_x, y + \Delta_y, z + \Delta_z) \quad (4)$$

K_1, K_2 and K_3 here denotes the pooling dimensions and the operations described in Equations (12)–(16) are performed in a cascaded fashion for $k = 1, 2$ as well. However, for $k = 2$, the input to the convolution layer is a fusion between resized high resolution features ($f_r^0(x, y, z)'$), and the output of the previous salient block i.e., $f^2(x, y, z) = g_m^1(x, y, z) + f_r^0(x, y, z)'$. In addition, the batch normalization and pooling operations are not performed at $k = 2$, rather, the network outputs $f_r^2(x, y, z)$ as the salient features. Afterward, the inconspicuous edge map (generated by accumulating the saliency features $f_r^2(x, y, z)$ with the input scan) is decomposed into a low energy tensor, which is further added with its high energy counterpart to generate a dual-tensor map. Here, contrary to the recent data fusion approaches (which use additional thermal [57] or depth [58] encoders), our approach utilizes a single lightweight feature extractor (containing only 1729 parameters) to produce good salient feature representations as evident from Figure 6 in Section 6.

4.2. Proposed Dual Tensor Scheme

After obtaining the inconspicuous edge map, we decompose it (along with the original input scan) into the low and high energy tensors to reveal the transitional variations of all the baggage content irrespective of their spatial characteristics. The motivation for proposing the dual-energy tensors stems from the fact that objects within the baggage X-ray scans exhibit different spatial characteristics. Some are composed of higher spectral bands whereas others blend more with the background (see, for example, the *shuriken* and *razors* in Figure 2). Therefore, such objects cannot be picked in one-go (especially through the trivial edge detection and representation methods). The proposed dual-tensor scheme amplifies the transitional variations of the cluttered items as compared to the state-of-the-art methods [5], leading towards more robust identification of the cluttered baggage threats. This dual-tensor decomposition within the proposed framework is performed through structure tensor [56], which in its simplest form a 2×2 symmetric matrix computed (pixel-wise) by taking an outer product of image gradients (defined by the neighborhood of each pixel within the candidate scan) [56], as expressed in Equation (5).

$$\mathfrak{S}_S = \begin{bmatrix} \varphi * (\nabla X \cdot \nabla X) & \varphi * (\nabla X \cdot \nabla Y) \\ \varphi * (\nabla Y \cdot \nabla X) & \varphi * (\nabla Y \cdot \nabla Y) \end{bmatrix} \quad (5)$$

where each outer product $\varphi * (\nabla i \cdot \nabla j)$, dubbed tensor, denotes the outer product of image gradients ∇i and ∇j oriented at direction i and j , respectively. φ denotes the Gaussian diffusion filter responsible for removing noisy outliers while retaining the transitional information of all the objects. It is computed through Equation (6):

$$\varphi = \sum_{s \in \omega_x} \sum_{t \in \omega_y} \frac{1}{2\pi\sigma^2} e^{-\frac{s^2+t^2}{2\sigma^2}} I(i, j; k) \quad (6)$$

$$I(i, j; k) = \sum_{n_k \in \langle \omega_x, \omega_y \rangle} \frac{1}{n_k! \Gamma(n_k + k + 1)} \left(\frac{i+j}{2} \right)^{2n_k + k} \quad (7)$$

where $I(i, j; k)$ denotes the modified Bessel function of k th order and $\Gamma(\cdot)$ represents the gamma function, i.e., $\Gamma(k) = (k - 1)!, \forall k \notin \mathbb{C}$. The block matrix in Equation (5) yields four outer products (tensors) from the candidate scan, where only three of them are unique (since this matrix is symmetric). Afterward, we add these tensors together to generate a single high energy tensor map (containing objects with the higher frequency components) and a single low energy tensor (depicting dulled

baggage content). These low and high energy tensors are further added together to generate a dual-energy tensor representation of the candidate scan as shown in Figure 2.

4.3. Edge Suppressing Backbone

The dual-energy tensor map emphasizes the edge representation of the dulled contraband items while retaining the prominent features of the baggage scans. However, before fitting the bounding boxes to localize the threatening items, we pass the dual-tensor map through the edge suppression backbone, trained via meta-transfer learning [17], to strain the irrelevant boundaries of the normal baggage content while only preserving the edges of the threatening items. The choice of the backbone network is extensively discussed in the ablation study (Section 6.1). In addition, the training of the backbone network via meta-transfer learning is presented in Section 4.5. Moreover, the processed tensor (obtained by multiplying the dual-energy tensor with the output of the backbone network) is then binarized from which the bounding boxes are fitted to localize the contraband items. The duplicate and redundant bounding boxes are removed through the nonmaximum suppression [55]. Afterward, these bounding boxes are utilized in cropping the object proposals from ζ_X which are then recognized via the meta-one-shot classifier.

4.4. Meta-One-Shot Classifier

Due to the capacity of the edge suppressing backbone network to differentiate between the contours of the threatening items and the normal baggage content, we also deploy it in conjunction with the fully connected layers to recognize the localized threatening items. Here, we fine-tune the backbone network (coupled with the fully connected layers) to recognize contraband items within the cropped object proposals via a single training example of each suspicious item category (i.e., we perform one-shot learning to recognize the proposals of the contraband items).

4.5. Training via Meta-Transfer Learning

In order to generalize the proposed tensor-shot detector to extract the contraband items irrespective of the scanner specifications, and also to avoid the requirement of large and well-annotated data for fine-tuning the pretrained weights of the backbone, we adopted meta-transfer learning [17] strategy as described in Algorithm 1. Here, "task" refers to the correct identification of each suspicious item category, and the meta-transfer learning (for the proposed tensor-shot detector) is performed in an iterative manner where, in the first iteration, we trained the backbone model on the dual-energy tensors (obtained from the joint GDXray [15] and SIXray [14] datasets) to suppress the contours of normal baggage content while retaining the edges of the prohibited items. Moreover, in the second iteration, we take the model weights (θ) (updated in the first iteration) and fine-tune them through the single-shot training to classify the localized proposal categories. The network weights (θ) learned in the first iteration enables the network to effectively recognize the baggage items (within each proposal) without retraining the whole network again for each dataset separately. Even fine-tuning on the single example of each category (in the second iteration) is optional as the proposed detector also produces decent performance with the zero-shot classifier (please see Section 6.5 for more details). Apart from this, the complete implementation details of the proposed detection framework are presented in Section 5.2.

Algorithm 1: Meta Transfer Learning Algorithm

```

1 Input: model with  $\theta$  weights ( $\mathcal{M}_\theta$ ), epochs ( $\mathcal{E}$ ), data batch ( $\mathcal{D}_b$ ), learning rate ( $l_r$ )
2 Output: updated weights
3  $\mathcal{S}, \mathcal{T} = \text{preprocessData}(\mathcal{D}_b)$  // get the samples and tasks from  $\mathcal{D}_b$ 
4 foreach epoch in  $\mathcal{E}$  do
5    $\mathcal{T}_l = 0$  // cumulative loss for all the tasks in each epoch
6   foreach task  $t$  in  $\mathcal{T}$  do
7      $\mathcal{S}_t = \text{filter samples } (\mathcal{S}) \text{ w.r.t } t$ 
8     Compute loss for the current task  $\mathcal{L}_t(\mathcal{M}_\theta)$  using  $\mathcal{S}_t$  examples
9     Evaluate  $\nabla_\theta \mathcal{L}_t(\mathcal{M}_\theta)$ 
10    Compute adapted weights:  $\theta_t = \theta - l_r \nabla_\theta \mathcal{L}_t(\mathcal{M}_\theta)$ 
11     $\mathcal{T}_l = \mathcal{T}_l + \mathcal{L}_t(\mathcal{M}_{\theta_t})$ 
12  end
13  Update:  $\theta = \theta - l_r \nabla_\theta \mathcal{T}_l$ 
14 end
15 return  $\theta$ 

```

4.6. Loss Function

The dual-tensor map contains imbalanced ratio of normal and threatening items contours. Therefore, penalizing the backbone model through the conventional cross-entropy loss function would make it biased towards producing more false positives (and false negatives as well). Therefore, in order to effectively train the model to distinguish between normal and threatening baggage content, we employ a focal loss function [41] within the proposed tensor-shot framework, as expressed below:

$$\mathcal{L} = -\frac{1}{b_s} \sum_{i=0}^{b_s-1} \sum_{j=0}^{c-1} \alpha(1 - p(l_{i,j}))^\gamma t_{i,j} \log(p(l_{i,j})) \quad (8)$$

where b_s is the batch size, c denotes the total number of classes, $t_{i,j}$ indicates whether or not the i th training example is from the j th class, $p(l_{i,j})$ represents the probability of the logit $l_{i,j}$ (generated by the network) for the i th training example belonging to the j th class, the expression $\alpha(1 - p(l_{i,j}))^\gamma$ depicts the scaling factor [41]. The values for the focal loss parameters are determined empirically through rigorous experimentation, as reported in the ablation study (Section 6.1.1).

5. Experimental Setup

In this section, we present a detailed description of the datasets, the implementation details as well as the evaluation metrics, which we used to compare the proposed framework with the state-of-the-art solutions.

5.1. Datasets

The proposed framework has been extensively evaluated on publicly available GRIMA X-ray database (GDXray) [15] and Security Inspection X-ray (SIXray) [14] dataset. GDXray [15] is the widely used dataset containing high resolution texture-less grayscale X-ray scans [15]. Moreover, SIXray [14] is the recently introduced large-scale dataset for baggage threat detection and to date it contains the most challenging colored X-ray scans. The detailed description of each dataset is presented below:

5.1.1. GRIMA X-ray Database

GDXray [15] is the widely used public dataset for baggage threat detection and also for the nondestructive testing (NDT) [15]. GDXray [15] is unique as it is the only public dataset containing the

19,407 texture-less grayscale scans in which it contains the baggage items that are heavily occluded as shown in Figure 1. Moreover, the scans within GDXray [15] are highly annotated and arranged within five categories, i.e., *welds*, *baggage*, *nature*, *casting*, and *settings*. The baggage groups (which is the only relevant category for this study) contain 8150 grayscale X-ray scans in which the suspicious items such as *handguns*, *razors*, *shuriken*, and *knives* have been marked by the security experts. Apart from this, we have marked the suspicious items in the original dataset (like *chip* and *mobile phones*) to further validate the performance of the proposed framework. To make things even more challenging, we have separated the original *handgun* category as *pistol* and *revolver*, to further test the capacity of the proposed detection framework in individually recognizing these items.

5.1.2. Security Inspection X-ray Dataset

SIXray [14] is the largest and, to the best of our knowledge, the most challenging dataset for the extraction and identification of baggage items from the colored X-ray images [14]. The dataset contains 1,059,231 scans having heavily occluded and cluttered items in which the suspicious items are grouped into six categories, i.e., *knives*, *guns*, *wrenches*, *scissors*, *pliers*, and *hammers*. Furthermore, the dataset has been organized into various subsets containing a highly imbalanced combination of positive and negative scans (positive means scan having one or more suspicious item and negative means scan having no suspicious item) to meet the real-world scenario. These subsets are named as SIXray10, SIXray100, and SIXray1000, respectively [14]. Apart from this, the dataset contains highly detailed annotations of baggage items that were marked by the security experts. These annotations served as ground truth for us to validate the performance of the proposed framework.

Here, we further want to highlight that both of these datasets contain a wide range of forbidden items that have been identified by the European Commission in this report [59].

5.2. Implementation Details

The proposed detection framework has been implemented on MATLAB R2020a using the deep learning, computer vision, and image processing toolbox on a machine with Intel Core i5, 16 GB RAM, and NVIDIA RTX 2080 GPU (with compute compatibility v7.5). For a fair comparison with the existing solutions, the scans used for training and testing the proposed tensor-shot detector were honored as per each dataset standard. First of all, we trained the salient network for 5 epochs on each dataset. Afterward, we conducted meta-transfer learning for 10 epochs (in the first iteration) to generalize the backbone network in distinguishing the edge representation of the normal and threatening baggage content based upon the 848,172 dual-energy tensors obtained from the training scans of combined GDXray [15] and SIXray [14] datasets. Moreover, in the second iteration, the meta-transfer learning was conducted for 2 epochs in which we trained the meta-one-shot classifier (with a single training example of each contraband item proposal) to effectively recognize them. Apart from this, we employed the stochastic gradient descent as an optimizer with a momentum of 0.9 and a static learning rate (l_r) of 0.001. These hyperparameters are determined empirically for both datasets through conventional grid search optimizations [60,61], where the learning rate was varied from 0.1 to 0.0001 by the drop factor of $\frac{1}{10}$, and momentum was varied from 0.5 to 0.95 in the step of 0.05.

5.3. Evaluation Metrics

To evaluate the performance of the proposed framework and also to compare it with the state-of-the-art solution, we have used the following metrics:

5.3.1. Intersection-over-Union

Intersection over Union (*IoU*), also known as Jaccard's similarity index measures the capacity of the framework that how well it has extracted the object of interest as compared to its ground truth. The *IoU* is computed through [62]:

$$\text{IoU} = \frac{T_p}{T_p + F_p + F_n} \quad (9)$$

where T_p denotes the true positives, F_p represents false positives, and F_n denotes the false negatives. Moreover, the mean *IoU* score, showcasing the overall object extraction performance of the proposed framework, is computed by taking an average of *IoU* score for each suspicious item category.

5.3.2. Dice Coefficient

Dice Coefficient (*DC*) also illustrates how accurately the proposed framework can extract the object regions and it is computed by measuring a degree of similarity between the extracted object regions with respect to their ground truths as expressed in Equation (10) [63]:

$$DC = \frac{2T_p}{2T_p + F_p + F_n} \quad (10)$$

Moreover, the mean *DC* is computed by taking the average of *DC* scores for all the suspicious items categories. The difference between *IoU* and *DC* is that *DC* gives more weight towards the accurate extraction of the contraband items (true positives) as compared to the *IoU*.

5.3.3. Mean Average Precision

The performance of the proposed framework for accurately detecting the prohibited items is measured by the mean average precision (*mAP*) scores. Here, the *mAP* scores are measured by taking the mean of average precision (*AP*) scores computed at an $\text{IoU} \geq 0.5$ for each suspicious item category.

$$mAP = \sum_{k=0}^{n_c-1} AP(k) \quad (11)$$

where n_c denotes the total number of contraband item categories.

5.3.4. Confusion Matrix

Apart from evaluating the detection performance of the proposed framework using *mAP*. We also validated its capacity to classify the baggage scan as threatening or nonthreatening using the confusion matrix and standard classification metrics such as accuracy, true positive rate (*TPR*), false positive rate (*FPR*), positive predicted value (*PPV*), and the *F₁* score as expressed below:

$$\text{Accuracy} = \frac{T_p + T_n}{T_p + F_n + F_p + T_n} \quad (12)$$

$$TPR = \frac{T_p}{T_p + F_n} \quad (13)$$

$$FPR = \frac{F_p}{F_p + T_n} \quad (14)$$

$$PPV = \frac{T_p}{T_p + F_p} \quad (15)$$

$$F_1 = 2 \times \frac{TPR \times PPV}{TPR + PPV} \quad (16)$$

where T_n denotes the true negatives.

5.3.5. Mean Squared Error

To further show the statistical significance of the proposed framework compared to the state-of-the-art solutions on both GDXray [15] and SIXray [14] dataset. We have used the mean squared error (*MSE*) scores. *MSE*, in this study, is computed for each contraband item class through Equation (17) [64]:

$$MSE = \frac{1}{n_t} \sum_{i=1}^{n_t} (y_i - \hat{y}_i)^2 \quad (17)$$

where y_i denotes the ground truth values for each item, \hat{y}_i denotes the predicted values of each contraband item, and n_t denotes the total number of instances of the respective item within the dataset. Here, it should be noted that the predicted values for each item are taken as their *mAP* scores, and their ground truth values represent ideal *mAP* performances, i.e., 1.

5.3.6. Qualitative Evaluations

Apart from quantitative evaluations, we also demonstrate the capacity of the proposed framework for accurately detecting the cluttered, concealed, and overlapping contraband items through extensive qualitative examples.

6. Results

In this section, we present a thorough evaluation of the proposed framework on two publicly available datasets. Furthermore, we showcase its detailed comparison with the state-of-the-art frameworks against various metrics. We also present an ablation study here through which we determined the optimal parameters for the focal loss function [41] and the backbone model for detecting the baggage threats.

6.1. Ablation Study

Before discussing the experimental results of the proposed framework, we present an ablation study where we determined the optical parameters for the focal loss function [41] and best backbone network for edge suppression and object proposals classification.

6.1.1. Determining the Focal Loss Parameters

The scaling factor within the focal loss function [41] consists of two hyperparameters, i.e., the α and the γ parameter. α represents the balancing factor that penalizes the network towards accurately recognizing the imbalanced classes, and γ is the focusing parameter that allows the network to down-weight the accurate recognition of easy examples to emphasize on the hard one. Here, we varied the value of α as $0.25 \leq \alpha \leq 0.75$, and the value of γ as $1 \leq \gamma \leq 5$ for both GDXray [15] and SIXray [14] datasets according to the grid search scheme [60,61]. From Table 2, we can observe that for GDXray [15] dataset, varying the value of α and β does not affect much the overall detection performance of the proposed framework. This is because GDXray [15] does not contain highly imbalanced contraband item classes. However, on SIXray [14], we see significant variations in the detection performance while varying α and γ , i.e., we achieved the maximum *mAP* score of 0.6457 on SIXray [14] dataset with $\alpha = 0.25, \gamma = 2$, and a minimum *mAP* score of 0.4926 with $\alpha = 0.75, \gamma = 1$. Here, it should also be noted that increasing the value of γ penalizes the proposed framework to focus more on the hard examples, whereas decreasing the value of α ensures high resistance to the imbalanced classes.

Table 2. Effect of varying focal loss parameters on the detection performance (in terms of *mAP*) on GDXray [15] and SIXray [14] dataset. Bold indicates the optimal performance.

GDXray [15]		α		
		0.25	0.5	0.75
γ	1	0.9059	0.8742	0.8693
	2	0.9162	0.8916	0.8869
	3	0.9143	0.8882	0.8807
	4	0.9017	0.8834	0.8762
	5	0.9064	0.8763	0.8691
SIXray [14]		α		
		0.25	0.5	0.75
γ	1	0.5483	0.5140	0.4926
	2	0.6457	0.6283	0.6182
	3	0.6283	0.6036	0.5874
	4	0.6156	0.5709	0.5370
	5	0.6083	0.5472	0.5198

Scores for the SIXray dataset represent the average of SIXray10, SIXray100, and SIXray1000 subset.

6.1.2. Determining the Classification Backbone

To determine the best backbone model, we tested the tensor-shot detector with ResNet-50 [46], ResNet-101 [46] and VGG-16 [48], where the detection performance with each of the backbones is reported in Table 3. We can observe here that although the best detection results are achieved with ResNet-101 [46] on both datasets, the choice of backbone does not significantly affect the overall detection performance of the proposed framework, e.g., the worse detection performance with VGG-16 [46] only lags by 5.14% on GDXray [15] and 5.83% on SIXray10 [14] from the best performing ResNet-101 [46] backbone.

Table 3. Performance of proposed detection framework in terms of *mAP* on GDXray [15] and SIXray [14] datasets using different classification backbones. Bold indicates the best performance.

Network	GDXray [15]	SIXray10 [14]	SIXray100 [14]	SIXray1000 [14]	SIXray [14] *
VGG-16 [48]	0.8691	0.7583	0.5721	0.4126	0.5810
ResNet-50 [46]	0.8917	0.7826	0.6284	0.4392	0.5915
ResNet-101 [46]	0.9162	0.8053	0.6791	0.4527	0.6457

* Average of SIXray10, SIXray100 and SIXray1000 subset.

6.2. Evaluations on GDXray Dataset

The first dataset on which we evaluated the proposed framework is the GDXray [15] dataset. The detection performance of the proposed framework on GDXray [15] is shown in Table 4. Here, we can observe that the proposed framework achieved the mean *IoU*, mean *DC* and the *mAP* score of 0.9118, 0.9536, and 0.9162, respectively. Furthermore, it outperformed [33] by achieving 1.49% improvements in terms of *PPV* and 0.573% improvements in terms of *F₁* score. However, it lags from [33] by 0.397% in terms of *TPR* and 2.90% in terms of accuracy. With that said, since *F₁* is a better score than accuracy especially for the imbalanced data and considering the fact that the proposed framework is also validated using standard *mAP* metric (where it achieved the score of 0.9162), we believe that the performance of the proposed framework is significant.

Table 4. Detection performance of the proposed framework on the GDXray [15] and the SIXray [14] dataset. Bold indicates the best performance while the second-best performance is underlined. ‘-’ denotes that the metric is not computed.

Dataset	Metric	Proposed	[14]	[51]	[33]	[31]	[65]	[66]
GDXray [15]	mean <i>IoU</i>	0.9118	-	-	-	-	-	-
	mean <i>DC</i>	0.9536	-	-	-	-	-	-
	<i>mAP</i>	0.9162	-	-	-	-	-	-
	Accuracy	<u>0.9554</u>	-	-	0.9840	0.9500	-	-
	<i>TPR</i>	<u>0.9761</u>	-	-	0.9800	-	0.8900	0.9430
	<i>TNR</i>	<u>0.9305</u>	-	-	-	0.9140	-	0.9440
	<i>FPR</i>	<u>0.0694</u>	-	-	-	0.0860	-	0.0560
	<i>PPV</i>	0.9441	-	-	<u>0.9300</u>	-	0.9200	-
	<i>F₁</i>	0.9598	-	-	<u>0.9543</u>	-	0.9047	-
SIXray [14] *	mean <i>IoU</i>	0.9238	-	-	-	-	-	-
	mean <i>DC</i>	0.9603	-	-	-	-	-	-
	<i>mAP</i>	<u>0.6457</u>	0.5938	0.8500	-	-	-	-
	Accuracy	<u>0.8949</u>	<u>0.4577</u>	-	-	-	-	-
	<i>TPR</i>	0.8127	-	-	-	-	-	-
	<i>TNR</i>	0.8956	-	-	-	-	-	-
	<i>FPR</i>	0.1043	-	-	-	-	-	-
	<i>PPV</i>	0.0621	-	-	-	-	-	-
	<i>F₁</i>	0.1153	-	-	-	-	-	-

* Average of SIXray10, SIXray100, and SIXray1000 subset.

In addition to this, Table 5 reports the statistical analysis of the proposed framework in terms of MSE scores. Here, to make the fair comparison with state-of-the-art frameworks, we only extracted the originally marked contraband items from the dataset i.e., *handguns*, *knives*, *razors*, and *shuriken*. We can observe from Table 5 that the proposed framework statistically outperforms the second-best [5] by 40.05% that is quite significant, especially because the framework in [5] is a fully supervised framework trained via conventional fine-tuning. However, the proposed framework employs meta-transfer learning for detecting suspicious baggage items.

Table 5. Statistical significance of the proposed framework compared to the state-of-the-art solutions in terms of MSE. Bold indicates the best results, while the second-best performances are underlined. ‘-’ indicates that the metric is not computed. Scores for the SIXray dataset represent the average of SIXray10, SIXray100, and SIXray1000 subset.

Dataset	Items	Proposed	[14]	[51]	[5]
GDXray [15]	Handguns	0.001436	-	-	<u>0.008082</u>
	Knives	<u>0.002683</u>	-	-	0.000030
	Razors	0.007586	-	-	<u>0.013782</u>
	Shuriken	<u>0.001459</u>	-	-	0.000068
	Mean	0.003291	-	-	<u>0.005490</u>
	STD	0.002530	-	-	<u>0.005802</u>
SIXray [14]	Guns	<u>0.021874</u>	0.018496	0.006400	0.000079
	Knives	0.030905	<u>0.021432</u>	0.044100	0.004264
	Wrenches	<u>0.060614</u>	0.101251	-	0.000072
	Scissors	<u>0.134762</u>	0.166219	-	0.000038
	Pliers	<u>0.087971</u>	0.030241	-	0.005372
	Mean	0.067225	0.067528	<u>0.025250</u> *	0.001965
	STD	0.041015	0.057959	<u>0.018850</u> *	0.002355

* These results are computed by only considering *guns* and *knives* items from SIXray10 [14] subset.

Apart from this, we also present the qualitative evaluation of the proposed detection framework in Figure 4 where we can observe how effectively the proposed tensor-shot detector recognizes the

concealed and cluttered contraband items. For example, see the detection of concealed pistol in (B and F), concealed *pistol* and a *laptop* (*chip*) in (J), the cluttered *pistol* and *knife* in (R), cluttered *revolver* in (L), and low contrasted *razor* in (T).

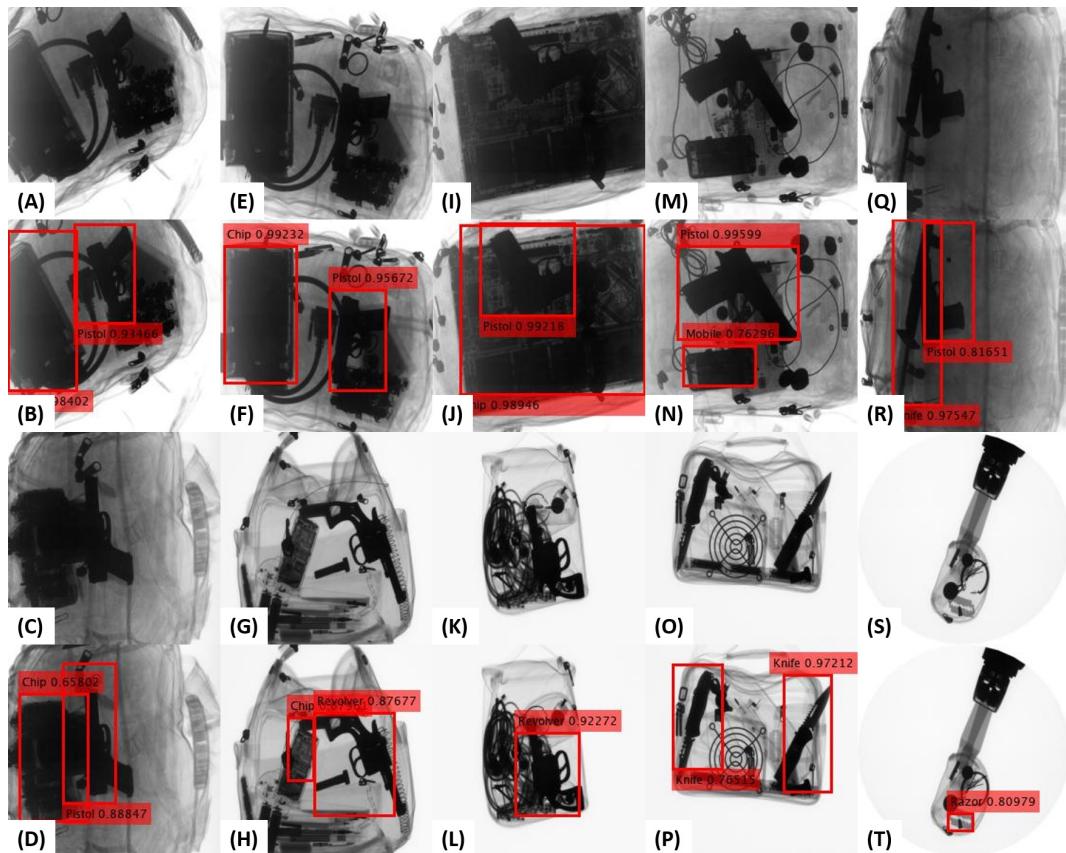


Figure 4. Qualitative performance of the proposed framework on the GDxray [15] dataset for detecting contraband items such as *pistols*, *revolvers*, *razors*, *chips*, *mobile phones*, and *knives* as shown in (A–T). Moreover, first and third row show the original scans. We can appreciate the detection of concealed pistol in (B,F), concealed pistol and a laptop (*chip*) in (J), the cluttered pistol and knife in (R), cluttered revolver in (L), and low contrasted razor in (T).

6.3. Evaluations on SIXray Dataset

The second dataset on which we have evaluated the proposed framework is the SIXray [14] dataset. SIXray [14] to the best of our knowledge is the largest and most challenging baggage X-ray dataset to date [14]. The detection performance of the proposed framework can be seen in Table 4 where we can observe that the proposed detector achieved an *mAP* score of 0.6457, outperforming [14] by 8.03%. Although it lags from [51] by 24.03%. However, this comparison is not fair because the authors in [51] only utilized SIXray10 subset of the SIXray dataset for extracting only the *guns* and *knives*. However, we evaluated the proposed framework on all the three subsets of the SIXray [14] dataset for extracting all the originally marked prohibited items [14]. Apart from this, we achieved an *F₁* score of 0.1153 on SIXray [14] dataset. We can notice here the substantial gap of 87.11% between the performance of the proposed framework in terms of accuracy and the *F₁* score. This is due to the fact that all the subsets of SIXray [14] dataset are extremely imbalanced [14]; therefore, we got an excessive number of false positives compared to the true positives (causing a very low precision and *F₁* score).

In another experiment, we quantified the capacity of the proposed framework to detect contraband items under various degrees of clutter and concealment. For this, we divided the positive scans within SIXray [14] dataset into three disjoint sets. The first set contains only those examples which contain contraband items under the low concealment. The second set contains examples with partially cluttered

suspicious objects, and the third set contains extremely cluttered and concealed contraband items. Please note that these sets are prepared by us just to give the quantitative representation on how well the proposed framework is resistant to the level of clutter, and we also want to highlight these sets are not present within the original SIXray [14] dataset. Furthermore, we performed this experiment only on the SIXray [14] dataset because SIXray [14] is, to the best of our knowledge, the largest and most challenging dataset designed for detecting baggage threats under the highly imbalanced scenario. GDXray [15], although, contains texture-less grayscale scans making the detection of contraband items (in some scans) difficult. However, overall, comparing the complexity of GDXray [15] with SIXray [14], SIXray [14] presents more challenging cases. The quantitative evaluation of the proposed framework for this experiment is shown in Table 6. Here, we can observe how effectively the proposed framework recognizes the suspicious items regardless of the clutter, occlusion, or concealment. Even in an extremely cluttered scenario, the performance of the proposed framework only deteriorates by 33.45%, which is 4.40% better than [14].

Table 6. Quantitative evaluation of the proposed framework and other state-of-the-art frameworks on SIXray [14] dataset towards detecting contraband items exhibiting different level of clutter and concealment.

Level of Clutter and Concealment	Proposed	[14]
Low	0.7816	0.7453
Partial or mild	0.6593	0.5918
Full or extreme	0.5201	0.4632

Moreover, Table 5 reports the statistical significance of the proposed framework in terms of MSE. Here, we have excluded the extraction of *hammers* to maintain consistency with the dataset standard [14] and the CHR [14] framework. From Table 5, we can see that although the proposed framework lags from [5]. However, because it utilizes meta-one-shot learning to recognize contraband items and still able to achieve comparable performance with the fully supervised frameworks (trained on the large-scale datasets), we believe that the performance of the proposed framework is promising. In addition, it should be noted that the comparison of the proposed framework with second-best [51] is not fair because they only studied SIXray10 [14] subset of the SIXray dataset [14] in their study for extracting only the *guns* and *knives*.

Apart from this, the capacity of the proposed framework to localize and detect the baggage threat can be seen in Figure 5. Here, we can observe how remarkably the extremely cluttered contraband items have been detected, e.g., see the detected *gun*, *knife*, and *wrenches* in (D), a cluttered *knife* in (H and L), the overlapping *guns* and a *knife* in (J). This is due to the fact that the proposed tensor-shot detector can suppress the unwanted edges while emphasizing the threatening regions within the candidate scan through its backbone.

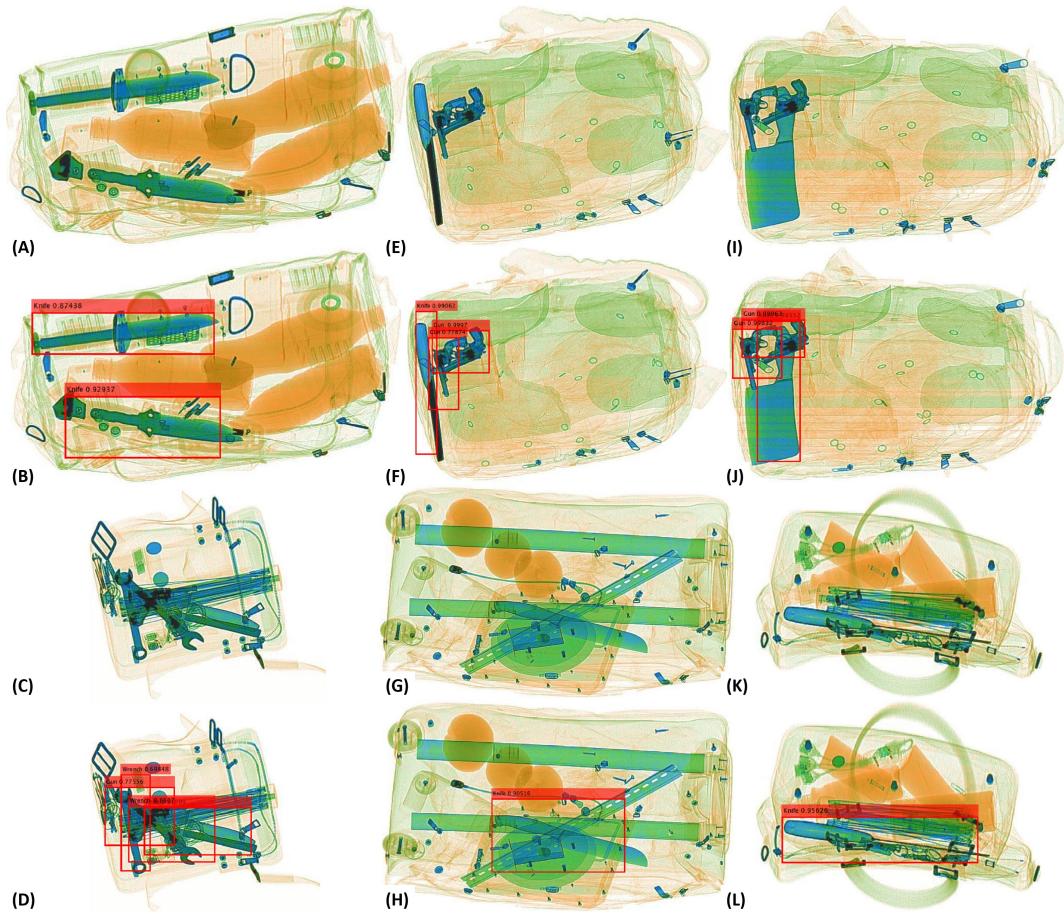


Figure 5. Qualitative performance of the proposed framework on the SIXray [14] dataset for detecting contraband items such as *knives*, *guns*, and *wrenches* as shown in (A–L). Moreover, first and third row show the original scans. These samples showcase the effectiveness of the proposed framework in detecting *gun*, *knife*, and *wrenches* in (D), a cluttered *knife* in (H,L), and the overlapping *guns* and a *knife* in (J).

6.4. Qualitative Analysis

Figure 6 shows the saliency maps obtained from the proposed salient network. Although, due to its shallow architecture, the salient model cannot generalize well against the diverse ranging X-ray scanners. Nevertheless, it can robustly pick the high transitional objects and suppress them for generating the low-energy tensors, e.g., see the extracted *knives* and *guns* in Figure 6 (L, P, and T) despite the extreme clutter. Moreover, the proposed dual-energy tensor scheme can reveal the boundaries of the low and high spectral threatening items; it also amplifies the transitions of normal baggage content (e.g., see the second and fifth columns in Figure 7). Here, to suppress the irrelevant edges, we employ a meta-transfer learning-driven backbone network (as discussed in Section 4.5) that is trained on the large-scale generalized dual-tensor representations of the grayscale and color X-ray scans. Furthermore, this backbone network is fine-tuned via single-shot training to recognize different contraband item proposals. The suppressed edges (computed by the generalized backbone model) can be seen in Figure 7. Here, we can appreciate its capacity to effectively strain the irrelevant edges regardless of the scanner specifications. Although, compared to GDXray [15], the backbone model produces better edge representations for the SIXray [14] dataset scans. This is because the backbone network is more biased towards SIXray [14] scanner as compared to the GDXray [14] due to the imbalanced ratio of the training scans within both datasets. However, this situation can be easily handled by employing different binarization thresholds (for each dataset) during postprocessing.

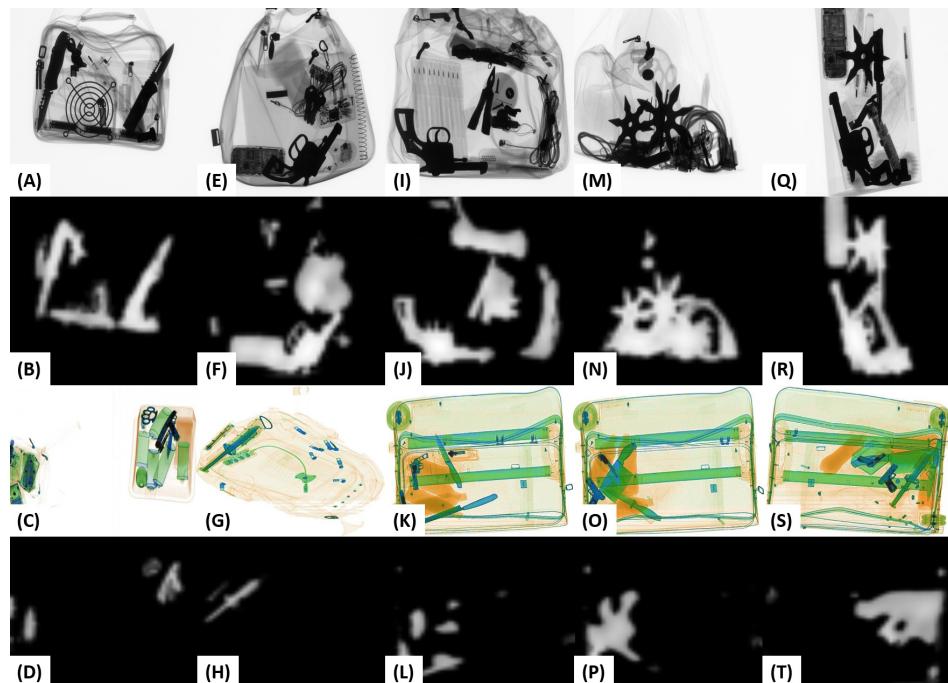


Figure 6. Saliency maps produced by the proposed salient feature extractor on both GDXray [15] and SIXray [14] datasets are shown in (A–T). Also, the first and third row show the original scans.

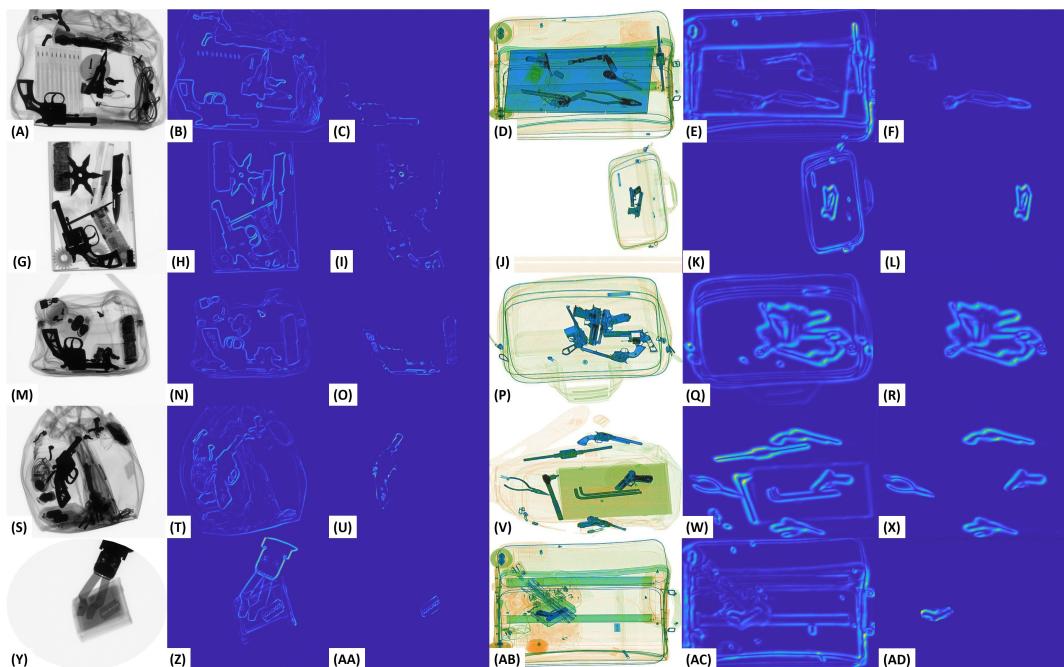


Figure 7. Irrelevant edge suppression through the proposed meta-transfer learning-driven tensor-shot detection framework are shown in ((A)–(AD)). Also, the first and fourth column show the original scans.

Despite the weak edge representations obtained for the GDXray [15] scans, the capacity of the generalized backbone model for edge suppression can be appreciated in Figure 7 (AA), where it has effectively retained the *razor* while suppressing all the irrelevant edges regardless of their prominence in the scan. Moreover, in Figure 8, we report some of the failures cases of the proposed tensor-shot detector on both datasets. The first failure is related to the incapacity of the edge suppression backbone to eliminate the irrelevant boundaries of the baggage content that produces multiple bounding

boxes for the same item e.g., see the twice detected *shuriken* in (B). Although, we handled such failures by applying nonmaximum suppression [55] as a postprocessing step. Still, because of the fixed overlapping threshold (in the nonmaximum suppression [55]), we rarely observed these errors. Although, we can avoid them by further decreasing the overlapping threshold. In addition, the other failure related to nonmaximum suppression [55] is the generation of loose bounding boxes, e.g., see the bounding box of a cluttered *knife* in (J). These loose boxes are generated by merging the multiple overlapping boxes (representing the same item). Although such errors are not drastic (as the framework is correctly detecting the item), such loose boxes can lead towards a low quantitative performance when compared to the ground truth. Moreover, the proposed framework also misses some extremely dulled and occluded objects, e.g., the *razor* in (B and F) (also in Figure 4P), and a *gun* in (H). These types of failures are related to the inability of the saliency model to accurately differentiate the low contrasted (and overlapping) objects within the low-energy tensor. Although we observed very few of these cases, they can be easily addressed by amplifying the dual-energy tensors before edge suppression. The last failure which we observed is the inability of the proposed tensor-shot detector to accurately detect all the overlapping instances of the same time in extremely challenging scenario, e.g., see the missed *knife* on top of *chopper* in (D), and the missed wrenches in (L), even the bounding box of the detected *knife* is not accurate. While we concur that the proposed framework is limited towards these false negatives (only if the scans are extremely challenging like Figure 8C,K), we can still appreciate the overall detection performance of the proposed framework given the fact that it is highly generalizable, and yet, produces decent detection results (even in cluttered scenarios), e.g., see the detected *knife* in (L) and the cluttered *knife* in (D).

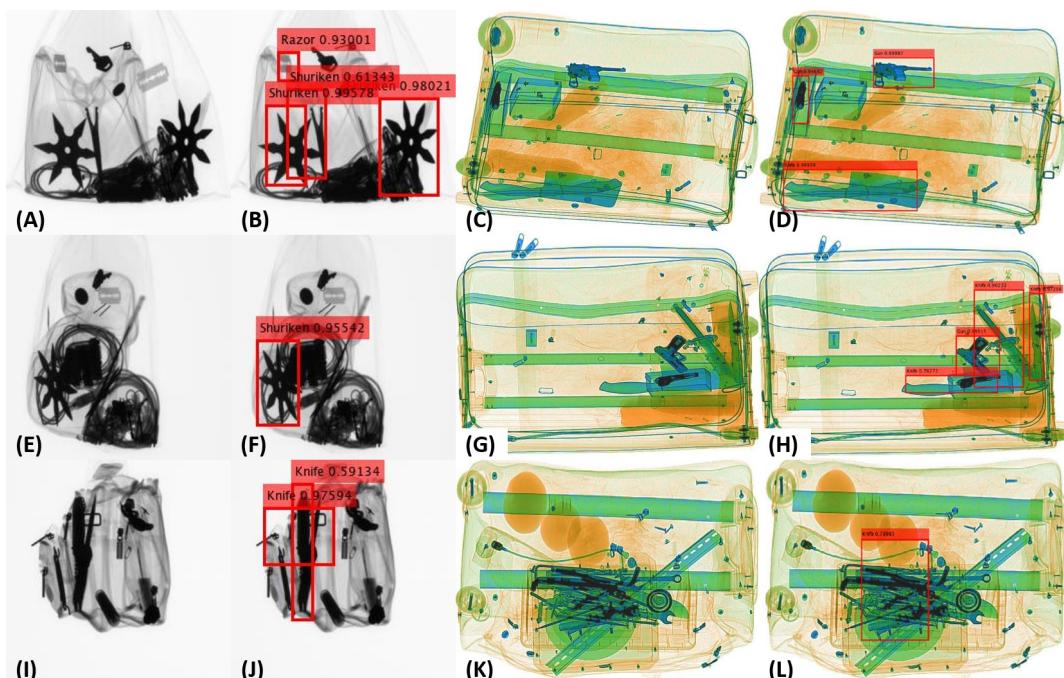


Figure 8. Failure cases on the GDXray [15] and SIXray [14] dataset are shown in (A–L). Moreover, the first and third column show the original scans.

6.5. Generalizability Assessment

To further test the generalizability of the proposed tensor-shot detector, we conducted another experiment where we trained the edge suppression backbone network (ResNet-101 [46]) on the dual-tensor representations of the training scans (of both datasets) and utilized a zero-shot classifier (driven through the generalized backbone model) to classify the proposals of contraband items. The detection performance of the proposed framework for this experimentation is shown in Table 7. Here, we can see that the proposed zero-shot tensor-shot driven detector achieved an *mAP*

score of 0.8069 on GDXray [15], and 0.4690 on SIXray [14] using ResNet-101 [46] as a backbone. In addition, the proposed framework achieved an *mAP* score of 0.6528, 0.4379, and 0.3164 on SIXray10, SIXray100, and SIXray1000 subset, respectively. Although, on average, the performance of the zero-shot detector lags by 11.50% on GDXray [15] and 27.36% on SIXray [14] dataset as compared to the one-shot detector but still the performance of the zero-shot detector is appreciable given the fact that the classifier does not require any fine-tuning even on single training examples.

Table 7. Detection performance of a zero-shot classifier (in terms of *mAP*) driven through the generalized edge suppression backbone.

Network	GDXray [15]	SIXray10 [14]	SIXray100 [14]	SIXray1000 [14]	SIXray [14]
ResNet-101 [46]	0.8069	0.6528	0.4379	0.3164	0.4690

7. Discussion and Conclusions

This paper presents a meta-transfer learning-based tensor-shot detection framework that can recognize highly concealed and cluttered baggage threats from the security X-ray scans. The proposed framework has been thoroughly tested on the two publicly available datasets (i.e., the SIXray [14] and the GDXray [15]). In addition, it has been extensively compared with the existing state-of-the-art solutions where it achieved 0.573% improvements (in terms of F_1 score) over [33] on GDXray [15] dataset and 8.03% improvements (in terms of *mAP*) over [14] on the SIXray [14] dataset.

Furthermore, through both quantitative and qualitative evaluations, we have demonstrated the capacity of the proposed framework for detecting the extremely cluttered contraband items on both grayscale and colored X-ray scanners. For instance, see the extraction of cluttered (and occluded) *pistol* and *revolver* in Figure 4D,F,J,L. In addition, in Figure 5, see the extraction of extremely occluded *gun*, *wrench* and *knife* in (D), a *knife* in (H and L). Moreover, Tables 4 and 6 further showcase the capacity of the proposed framework towards recognizing contraband items regardless of the occlusion, baggage clutter, and concealment.

Apart from this, the proposed framework is, to the best of our knowledge, the first baggage threat detector that is invariant to the scanner specifications and can work on any grayscale or colored X-ray scan for recognizing the potential threats. This is due to its capacity to transform the candidate scan into novel dual-energy tensors from which it identifies the threatening items even in extreme clutter and concealment. In addition, the proposed framework can be practically deployed in the real world for mass screening baggage threats (including the cluttered ones which, although, modern X-ray scanners can reveal, yet they can be missed by the security officers during the manual inspection due to the rush hours and tiring work schedule).

In future, the proposed tensor-shot framework can be utilized in detecting 3D printed and dismantle items from the baggage X-ray scans which are barely visible even to the human observers. Furthermore, it can also be tested on normal RGB scans for detecting concealed, cluttered, and occluded objects.

Supplementary Materials: The following are available online at www.mdpi.com/xxx/s1, Table S1. Summary of existing works related to autonomous baggage threat detection.

Author Contributions: Conceptualization, T.H., M.B., and N.W.; methodology, T.H., S.A. and S.K.; software, T.H. and M.S.; validation, T.H., M.S., and S.A.; resources, N.W. and E.D.; writing—original draft preparation, T.H. and M.S.; writing—review and editing, T.H., N.W., and E.D.; supervision, M.B., S.K., and N.W.; funding acquisition, N.W. and E.D. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by a research fund from Khalifa University. Ref: CIRA-2019-047 and the Abu Dhabi Department of Education and Knowledge (ADEK), Ref: AARE19-156.

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

References

1. National Research Council. *Airline Passenger Security Screening: New Technologies and Implementation Issues*; National Academies Press: Washington, DC, USA, 1996.
2. Cargo Screening. Aviation Security International. Available online: <https://www.asi-mag.com/cargo-screening-improvement/> (accessed on 4 December 2019).
3. Sterchi, Y.; Hattenschwiler, N.; Schwaninger, A. Detection measures for visual inspection of X-ray images of passenger baggage. *Atten. Percept. Psychophys.* **2019**, *81*, 1297–1311. [CrossRef] [PubMed]
4. Wells, K.; Bradley, D. A Review of X-ray Explosives Detection Techniques for Checked Baggage. *Appl. Radiat. Isot.* **2012**, *70*, 1729–1746. [CrossRef] [PubMed]
5. Hassan, T.; Bettayeb, M.; Akçay, S.; Khan, S.; Bennamoun, M.; Werghi, N. Detecting Prohibited Items in X-ray Images: A Contour Proposal Learning Approach. In Proceedings of the 27th IEEE International Conference on Image Processing (ICIP), Abu Dhabi, UAE, 25–28 October 2020; pp. 2016–2020.
6. Bilsen, V.; Rademaekers, K.; Berden, K.; Zane, E.B.; Voldere, I.D.; Jans, G.; Mertens, K.; Regeczi, D.; Slingenbergh, A.; Smakman, F.; et al. *Study on the Competitiveness of the EU Eco-Industry*; ECORYS Research and Publishing: Brussels, Belgium, 2009.
7. Wells, K.; Bradley, D. *Rethinking Checked Baggage Screening*; Reason Public Policy Institute Policy Study: Los Angeles, CA, USA, 2002; p. 297.
8. Bastan, M.; Byeon, W.; Breuel, T. Object Recognition in Multi-View Dual Energy X-ray Images. In Proceedings of the British Machine Vision Conference, Bristol, UK, 9–13 September 2013; p. 11.
9. Akçay, 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]
10. Gaus, Y.F.A.; Bhowmik, N.; Akçay, S.; Guillén-Garcia, P.M.; Barker, J.W.; Breckon, T.P. Evaluation of a Dual Convolutional Neural Network Architecture for Object-wise Anomaly Detection in Cluttered X-ray Security Imagery. In Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 14–19 July 2019; pp. 1–8.
11. Hassan, T.; Akçay, S.; Bennamoun, M.; Khan, S.; Werghi, N. Trainable Structure Tensors for Autonomous Baggage Threat Detection Under Extreme Occlusion. *arXiv* **2020**, arXiv:2009.13158.
12. Akçay, S.; Kundegorski, M.E.; Devereux, M.; Breckon, T.P. Transfer learning using convolutional neural networks for object classification within X-ray baggage security imagery. In Proceedings of the IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 1057–1061.
13. 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. *arXiv* **2020**, arXiv:2004.08656.
14. 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 IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 18–20 June 2019; pp. 2119–2128.
15. Mery, D.; Riffó, 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*, 42. [CrossRef]
16. Finn, C.; Abbeel, P.; Levine, S. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. *arXiv* **2017**, arXiv:1703.03400.
17. Sun, Q.; Liu, Y.; Chua, T.S.; Schiele, B. Meta-Transfer Learning for Few-Shot Learning. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 18–20 June 2019; pp. 403–412.
18. Mery, D.; Svec, E.; Arias, M.; Riffó, V.; Saavedra, J.M.; Banerjee, S. Modern Computer Vision Techniques for X-Ray Testing in Baggage Inspection. *IEEE Trans. Syst. Man Cybern. Syst.* **2017**, *4*, 682–692. [CrossRef]
19. Mery, D.; Saavedra, D.; Prasad, M. X-Ray Baggage Inspection With Computer Vision: A Survey. *IEEE Access* **2020**, *8*, 145620–145633. [CrossRef]
20. Akçay, S.; Breckon, T. Towards Automatic Threat Detection: A Survey of Advances of Deep Learning within X-ray Security Imaging. *arXiv* **2020**, arXiv:2001.01293.

21. Wang, G.; Luo, C.; Sun, X.; Xiong, Z.; Zeng, W. Tracking by Instance Detection: A Meta-Learning Approach. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 16–18 June 2020; pp. 6288–6297.
22. Hsu, K.; Levine, S.; Finn, C. Unsupervised Learning via Meta-Learning. *arXiv* **2018**, arXiv:1810.02334.
23. Vinyals, O.; Blundell, C.; Lillicrap, T.; Kavukcuoglu, K.; Wierstra, D. Matching Networks for One Shot Learning. In Proceedings of the Neural Information Processing Systems (NIPS), Barcelona, Spain, 6–12 December 2016; pp. 3630–3638.
24. Oreshkin, B.N.; Rodriguez, P.; Lacoste, A. TADAM: Task dependent adaptive metric for improved few-shot learning. In Proceedings of the Neural Information Processing Systems (NIPS), Montreal, QC, Canada, 3–8 December 2018; pp. 721–731.
25. 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 (ICIT), Cape Town, South Africa, 25–28 February 2013; pp. 1140–1145.
26. Heitz, G.; Chechik, G. Object Separation in X-ray Image Sets. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, CA, USA, 13–18 June 2010; pp. 2093–2100.
27. Baştan, M. Multi-view Object Detection In Dual-energy X-ray Images. *Mach. Vis. Appl.* **2015**, *26*, 1045–1060. [[CrossRef](#)]
28. Baştan, M.; Yousefi, M.R.; Breuel, T.M. Visual Words on Baggage X-ray Images. In Proceedings of the 14th International Conference on Computer Analysis of Images and Patterns, Seville, Spain, 29–31 August 2011; pp. 360–368.
29. Kundegorski, M.E.; Akçay, S.; Devereux, M.; Mouton, A.; Breckons, T.P. On using Feature Descriptors as Visual Words for Object Detection within X-ray Baggage Security Screening. In Proceedings of the IEEE International Conference on Imaging for Crime Detection and Prevention (ICDP), Madrid, Spain, 23–25 November 2016; pp. 1–6.
30. Mery, D.; Svec, E.; Arias, M. Object Recognition in Baggage Inspection Using Adaptive Sparse Representations of X-ray Images. In *Pacific-Rim Symposium on Image and Video Technology*; Springer: Cham, Switzerland, 2016; pp. 709–720.
31. Riffo, V.; Mery, D. Automated Detection of Threat Objects Using Adapted Implicit Shape Model. *IEEE Trans. Syst. Man Cybern. Syst.* **2016**, *46*, 472–482. [[CrossRef](#)]
32. Liu, Z.; Li, J.; Shu, Y.; Zhang, D. Detection and Recognition of Security Detection Object Based on YOLO9000. In Proceedings of the 2018 5th International Conference on Systems and Informatics (ICSAI), Nanjing, China, 10–12 November 2018; pp. 278–282.
33. 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.
34. Xu, M.; Zhang, H.; Yang, J. Prohibited Item Detection in Airport X-Ray Security Images via Attention Mechanism Based CNN. In Proceedings of the Chinese Conference on Pattern Recognition and Computer Vision, Guangzhou, China, 23–26 November 2018; pp. 429–439.
35. Jaccard, N.; Rogers, T.W.; Morton, E.; Griffin, L.D. Detection of Concealed Cars In Complex Cargo X-ray Imagery Using Deep Learning. *J. X-ray Sci. Technol.* **2017**, *25*, 323–339. [[CrossRef](#)]
36. Griffin, L.D.; Caldwell, M.; Andrews, J.T.A.; Bohler, H. “Unexpected Item in the Bagging Area”: Anomaly Detection in X-Ray Security Images. *IEEE Trans. Inf. Forensics Secur.* **2019**, *14*, 1539–1553. [[CrossRef](#)]
37. An, J.; Zhang, H.; Zhu, Y.; Yang, J. Semantic Segmentation for Prohibited Items in Baggage Inspection. In Proceedings of the International Conference on Intelligence Science and Big Data Engineering Visual Data Engineering, Nanjing, China, 17–20 October 2019; pp. 495–505.
38. Zou, L.; Yusuke, T.; Hitoshi, I. Dangerous Objects Detection of X-Ray Images Using Convolution Neural Network. In *Security with Intelligent Computing and Big-data Services*; Springer: Cham, Switzerland, 2018.
39. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 779–788.

40. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
41. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal Loss for Dense Object Detection. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2980–2988.
42. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In Proceedings of the 29th Conference on Neural Information Processing Systems (NIPS 2015); Montreal, Canada, 7–12 December 2015; pp. 91–99.
43. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going Deeper with Convolutions. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1–9.
44. Xiao, H.; Zhu, F.; Zhang, R.; Cheng, Z.; Wang, H.; Alesund, N.; Dai, H.; Zhou, Y. R-PCNN Method to Rapidly Detect Objects on THz Images in Human Body Security Checks. In Proceedings of the IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation, Guangzhou, China, 8–12 October 2018; pp. 1777–1782.
45. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2961–2969.
46. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
47. Iandola, F.N.; Han, S.; Moskewicz, M.W.; Ashraf, K.; Dally, W.J.; Keutzer, K. SqueezeNet: AlexNet-Level Accuracy with $50 \times$ Fewer Parameters and <0.5 MB Model Size. *arXiv* **2016**, arXiv:1602.07360.
48. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.
49. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. In Proceedings of the 26th Annual Conference on Neural Information Processing Systems (NIPS 2012), Lake Tahoe, NV, USA, 3–8 December 2012; pp. 1106–1114.
50. Dai, J.; Li, Y.; He, K.; Sun, J. R-FCN: Object Detection via Region-based Fully Convolutional Networks. In Proceedings of the 30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, 5–10 December 2016; pp. 379–387.
51. Gaus, Y.F.A.; Bhowmik, N.; Akcay, S.; Breckon, T. Evaluating the Transferability and Adversarial Discrimination of Convolutional Neural Networks for Threat Object Detection and Classification within X-Ray Security Imagery. *arXiv* **2019**, arXiv:1911.08966.
52. Akçay, S.; Atapour-Abarghouei, A.; Breckon, T.P. GANomaly: Semi-Supervised Anomaly Detection via Adversarial Training. In *Asian Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 622–637.
53. Akçay, S.; Atapour-Abarghouei, A.; Breckon, T.P. Skip-GANomaly: Skip Connected and Adversarially Trained Encoder-Decoder Anomaly Detection. *arXiv* **2019**, arXiv:1901.08954.
54. 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 European Conference on Computer Vision (ECCV 2016), Amsterdam, The Netherlands, 8–16 October 2016; pp. 21–37.
55. Bodla, N.; Singh, B.; Chellappa, R.; Davis, L.S. Soft-NMS – Improving Object Detection With One Line of Code. In Proceedings of the International Conference on Computer Vision (ICCV 2017), Venice, Italy, 22–29 October 2017; pp. 5561–5569.
56. Bigun, J.; Granlund, G. Optimal Orientation Detection of Linear Symmetry. In Proceedings of the 1st International Conference on Computer Vision (ICCV), London, UK, 8–11 July 1987; pp. 433–438.
57. Sun, Y.; Zuo, W.; Liu, M. RTFNet: RGB-Thermal Fusion Network for Semantic Segmentation of Urban Scenes. *IEEE Robot. Autom. Lett.* **2019**, *4*, 2576–2583. [[CrossRef](#)]
58. Hazirbas, C.; Ma, L.; Domokos, C.; Cremers, D. FuseNet: Incorporating Depth into Semantic Segmentation via Fusion-based CNN Architecture. In Proceedings of the Asian Conference on Computer Vision, Taipei, Taiwan, 22–24 November 2016; pp. 213–228.

59. European Commission. List of Prohibited Articles in your Cabin Baggage. *Mobil. Transp.* Available online: https://ec.europa.eu/transport/sites/transport/files/modes/air/security/doc/info_travellers_hand_luggage.pdf (accessed on 1 October 2020).
60. Chui, K.T.; Liu, R.W.; Zhao, M.; Pablos, P.O.D. Predicting Students' Performance With School and Family Tutoring Using Generative Adversarial Network-Based Deep Support Vector Machine. *IEEE Access* **2020**, *8*, 86745–86752. [[CrossRef](#)]
61. Fayed, H.A.; Atiya, A.F. Speed up grid-search for parameter selection of support vector machines. *Appl. Soft Comput.* **2019**, *80*, 202–210. [[CrossRef](#)]
62. Tan, P.N.; Steinbach, M.; Kumar, V. *Introduction to Data Mining*; Pearson: London, UK, 2005. ISBN 0-321-32136-7.
63. Murguia, M.; Villasenor, J.L. Estimating the effect of the similarity coefficient and the cluster algorithm on biogeographic classifications. *Ann. Bot. Fenn.* **2003**, *40*, 415–421.
64. Pishro-Nik, H. *Introduction to Probability, Statistics, and Random Processes*; Kappa Research LLC: Sunderland, MA, USA, 2014. ISBN: 0990637204.
65. Riffo, V.; Mery, D. Active X-ray testing of complex objects. *Insight Non Destr. Test. Cond. Monit.* **2012**, *54*, 28–35. [[CrossRef](#)]
66. Mery, D. Automated detection in complex objects using a tracking algorithm in multiple X-ray views. In Proceedings of the IEEE CVPR 2011 Workshops, Colorado Springs, CO, USA, 20–25 June 2011; pp. 41–48.

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).