Multimedia Tools and Applications
https://doi.org/10.1007/s11042-023-16057-7

Programmable broad learning system for baggage threat recognition

Muhammad Shafay[1] · Abdelfatah Ahmed[1] · Taimur Hassan[2] · Jorge Dias[1] · Naoufel Werghi[1]

Received: 5 September 2022 / Revised: 26 May 2023 / Accepted: 18 June 2023

Abstract
Detecting illegal and harmful objects in baggage at airports, subways, and bus stations has always been a difficult task that requires intense focus and concentration. Despite recent advances, developing systems for robust autonomous threat recognition still remains a challenge. In this paper, we propose a novel CNN-driven Broad Learning System, dubbed Programmable BLS, for identifying threat objects from the security X-ray scans. The proposed framework first extracts latent features from the input scan utilizing the CNN backbone. Then, the BLS model uses these features to assess whether or not the candidate scan contains the threat items. Unlike existing approaches, the design adaptation of the BLS architecture (within the proposed framework) is fully autonomous, requiring no human efforts to formulate the optimized combination of layers that gives the best classification performance for the given application. This unique design adaption is based on heuristics and greedy searches that measure the relevance of fusing adjacent node pairs in order to improve the overall network performance. Apart from this, across three datasets, namely, GDXray, SIXray, and COMPASS-XP, we rigorously tested the proposed framework on which it outperforms the state-of-the-art by 0.996%, 4.82%, and 0.934%, respectively, in terms of accuracy, and by 3.56%, 1.71%, and 1.30%, respectively, in terms of F1-score.

Keywords Broad learning systems · Greedy search · Baggage classification · Baggage X-ray scans

1 Introduction

Baggage scanning is essential for ensuring the safety of international travelers and reducing the risk of terrorist activities. An estimated 1.5 million passengers in the United States are

✉ Abdelfatah Ahmed
100059689@ku.ac.ae

[1] Department of Electrical Engineering and Computer Science, Khalifa University, Abu Dhabi, UAE

[2] Department of Electrical, Computer and Biomedical Engineering, Abu Dhabi University, Abu Dhabi, UAE

Published online: 13 July 2023

inspected daily for potentially harmful objects, according to a recent research [1]. Furthermore, nearly 4.5 billion individuals are estimated to have traveled by airplane in 2019 [2], and the movement of freight and cargo across international borders has been given a further boost to globalization and online commerce. In order to ensure safety in the aviation processes, a number of different security measures have been implemented in different parts of the world. Moreover, the manual inspection of the baggage to check the presence of prohibited items is a process that is both time-consuming and inefficient. Also, it is affected by the fatigue and tiring schedules of the human operators. Therefore, there is a need to develop an efficient and trustworthy baggage screening system capable of identifying contraband data on the fly. To address this, many researchers have proposed automated baggage screening solutions to aid the security staff in lowering the risk of aviation threats. The majority of these frameworks, however, are based on traditional RGB detectors, which have limited performance in locating occluded suspicious objects due to their region-based proposal generating methods [3] and the intrinsic discrepancies between RGB and X-ray images [4]. To address this, many works have presented clutter-aware methods that can detect hidden and occluded threats independent of acquisition errors [5], imbalance distribution of the utilized data [6], or scanner limitations [3, 7, 8]. Moreover, early studies on 2D luggage X-ray imaging depend on image pre-processing techniques like noise reduction and enhancement to help identify contraband items by revealing additional details. To enhance threat detection performance, the majority of these studies used shape-based estimation methods (2D and 3D) of threat items incorporated with the bag-of-visual-words (BoVW) architecture [9, 10]. Several studies employed region-based approaches and the material differentiability feature of dual-energy X-rays to detect dangerous objects from baggage scans. The limitations of traditional techniques were due to their reliance on hand-crafted features generated from images, which was solved with the development of deep learning algorithms. Conventional neural net (CNN)-based algorithms have improved the process of baggage screening by employing their ability to extract compact, comprehensive, and many layers of feature representation. Early work focused on denoising 3D suitcase CT (Computed Tomography) pictures for threat (non-explosive) identification, which was followed by machine learning approaches in 3D item categorization and segmentation. Recent research using 3D CNNs has yielded encouraging results [11, 12].

Despite the strategies proposed to perform the threat detection under occlusion [7, 8, 13], and imbalanced [6] data nature, the classification and detection of highly cluttered and concealed threat objects from non-threat objects is still an open problem. One of the key reasons for this is the inherent complexity and variability of occluded and cluttered scenes. The presence of multiple objects, various orientations, and diverse lighting conditions make it challenging to accurately identify and differentiate between threat and non-threat objects. Another major challenge is the imbalanced nature of the available data. In threat detection scenarios, the number of non-threat objects typically far exceeds the number of threat objects. This class imbalance can lead to biased models that perform well in identifying non-threat objects but struggle with detecting rare threat objects. Imbalanced data can hinder the learning process of machine learning algorithms, resulting in reduced performance when it comes to detecting concealed threat objects.

In this paper, regardless of the scanner specifications, we present a novel greedy search-based Broad Learning System (BLS) optimization in conjunction with CNN to identify threats from cluttered and concealed baggage. Using this framework (Programmable BLS), the methodology is leading among other state-of-the-art in terms of scan level classification. To incorporate the benefits of both BLS and CNN, we integrate the BLS optimization into the architecture of a CNN. The CNN serves as the backbone of the model, leveraging its ability to extract hierarchical features from input images. The BLS optimization complements the

CNN by fine-tuning its parameters and structure, allowing it to adapt to the complexities of the threat detection task.

2 Related work

This section first provides a quick introduction to some of the traditional baggage threat detection systems before getting into the state-of-the-art deep learning-based approaches. Also, it gives an overview of Broad Learning based-Systems. For a comprehensive survey, we direct the reader to [14, 15] work.

2.1 Traditional methods

The initial wave of threat recognition approaches employed machine learning techniques for classification, detection, and segmentation. These methods are based on hand-crafted features [16]. Support Vector Machine (SVM) [9] [17], Random Forest (RF) [18], and K-Nearest Neighbor (K-NN) [19] classifiers are considered evaluated for the purpose of classification of X-ray images to detect illegal objects utilizing Bag-of-Visual-Words (BoVW) representation [20]. In addition, SURF [21], and Fast SURF [22] are also utilized along with basic machine learning-based classifiers for extracting key-point descriptions that are used to classify the prohibited items. Heitz et al. [23] developed a scheme different than the aforementioned to use SURF characteristics with the region-growing technique for baggage threat recognition. Moreover, creating a dictionary that contains the feature descriptors from cropped X-ray scans patches is another BoVW-based approach for classification [24]. Kundegorski et al. [25] proposed a binary classification based on BoVW for classifying two different gun types. The author combined key descriptors with the Fast-SURF features and used SVM as the binary classifier to achieve optimal accuracy.

2.2 Deep learning methods

Deep learning algorithms have significantly improved the detection abilities of threat screening frameworks to the point that they have the ability to identify contraband items in both grayscale and colored X-ray scans despite the scanner's specifications. Moreover, Akcay et al. [26] proposed to employ GAN for separating anomalous data from the regular one, where regular data includes normal scans with no threat objects. Mery et al. [27] introduced the publicly available dataset named as GDXray for the purpose of evaluating different techniques for detecting, classifying, and segmenting contraband items from regular ones. Furthermore, Miao et al. [6] introduced the Security Inspection X-ray (SIXray) dataset for recognizing five different classes of threat objects. SIXray dataset is utilized to deal with the class imbalance problem, along with including greatly cluttered images which make it the hardest dataset to work on. The author proposed a CHR framework to deal with the class imbalance in the SIXray dataset. Besides, Wei et al. [13] created the De-occlusion Attention Module (DOAM), a plug-and-play configuration that can be used in conjunction with object detectors to identify and locate obscured contraband objects in luggage X-ray scans. DOAM has been thoroughly checked on the occluded prohibited items X-ray (OPIXray) dataset [13]. Hassan et al. developed two contour-driven object detectors, namely, Cascaded Structure Tensors (CST) [3], and Dual-Tensor Shot Detector (DTSD) [8]. An et al. [28] designed a segmentation approach by employing encoder-decoder models leveraging dual attention mechanisms. In addition to

that, the first-ever contour instance segmentation framework [7] was exclusively designed to detect the cluttered prohibited objects from the security X-ray scans.

2.3 Broad Learning System (BLS)

Numerous studies have been made to propose different variants of BLS. Robust BLS (RBLS) has been developed by [29] to deal with data modeling by addressing the outliers. The objective function for RBLS is generated by considering that the regression residual and output weights follow their corresponding distributions, and the output weights for robust modeling may be calculated via bayesian statistics. The resilience of RBLS can then be further strengthened by using regularization techniques and validating the work on various UCI datasets. Similar to this, in [30], Graph regularized BLS (GBLS) has been proposed for BLS regularization. The manifold learning is included in the objective function of the classic BLS, taking into consideration the regionally consistent characteristic of data, which means that comparable images may have similar characteristics. The output weights of GBLS are limited to learn more discriminatory features, and the classifier's performance may be improved further. Moreover, the employment of BLS for the classification of hyperspectral images (HSI) has been introduced in [31]. A hierarchical guidance filter is applied to HSI to obtain spatial and spectral characteristics. The probability-based class hierarchy is then included in the broad learning method to provide a semi-supervised broad learning version, allowing for the concurrent use of a few labeled examples and many unlabeled examples. Then, using the ridge regression estimate, the linking weights of the broad structure may be simply determined. Despite this, these methods proposed different variants of BLS. However, to the best of our knowledge, there is no research work that addresses the hyperparameter optimization of BLS. Therefore, in this research, we propose the greedy search-based optimization of the links inside BLS, along with coupling it with the CNN network for detecting the prohibited objects.

2.4 Contributions

This paper presents a novel greedy search-based optimization of the Broad Learning System in conjunction with CNN for the classification of threat objects from non-threat objects regardless of the scanner specifications. This proposed approach (named Programmable BLS) can extract deep features from candidate images and then use such characteristics to identify luggage content using a programmatic BLS model design. The distinctive features of the presented research work are listed below:

- A hybrid Programmable BLS to detect and classify threat objects in X-ray scans
- A methodology for optimizing the model architecture employing a greedy search approach
- To the best of our knowledge, this study is the first attempt to solve the baggage threat identification problem using a broad learning system.
- A thorough evaluation of three publicly available datasets showing the dominance of the proposed framework over the current state-of-the-art threat detection frameworks.

The rest of the paper is organized as follows: Sect. 3 presents the proposed framework, Sect. 4 showcases the experimental setup, Sect. 5 presents the detailed evaluation results, and Sect. 6 discusses the prospects of the proposed framework and concludes the paper.

3 Approach and methodology

X-ray imagery consists of extremely cluttered images and consists of such cases where even human observers fail to detect the items inside the scans. The goal of the proposed method is to classify the contraband items into respective classes. BLS can learn incrementally without the need for retraining when new nodes or input data is introduced. It has been demonstrated that BLS can address the flaws in gradient-based deep learning algorithms induced by training a high degree of complexity. Hence, BLS is used for the purpose of scan-level classification. The block diagram of the proposed system can be seen in Figure 1. As seen, the system is considered a hybrid with a two-step procedure. Initially, the pre-trained convolutional network serves as a backbone for extracting the abstract features from the images. The backbone is responsible for returning network activation for a specific layer from the net. This set of network activation represents the feature maps in the matrix form and serves as the input for BLS. The BLS consists of five trainable parameters namely, feature nodes weight matrix and bias (W_f, β_f), enhancement nodes weight matrix (W_h,β_h) and collective weight matrix W. Following the original BLS methodology, $\phi_i(.)$ is generated using the features that are based on sparse auto-encoders. It is possible to use a different mapping function other than the one based on sparse auto-encoder such as Radial Bias kernel or Gaussian kernel mapping. But in the case of cluttered X-ray scans, we preferred to apply the sparse auto-encoder-based kernel mapping as used in the original work. Using the CNN backbone for generating feature nodes Z_i has the following benefits.

- When it comes to data extraction using different CNN layers, there are a variety of options, which gives us more ways to improve the model's efficiency.
- CNN backbones pre-trained on X-ray imagery dataset tends to extract localized features leading to better performance of the model.

More details about the system components will be provided in the following sections.

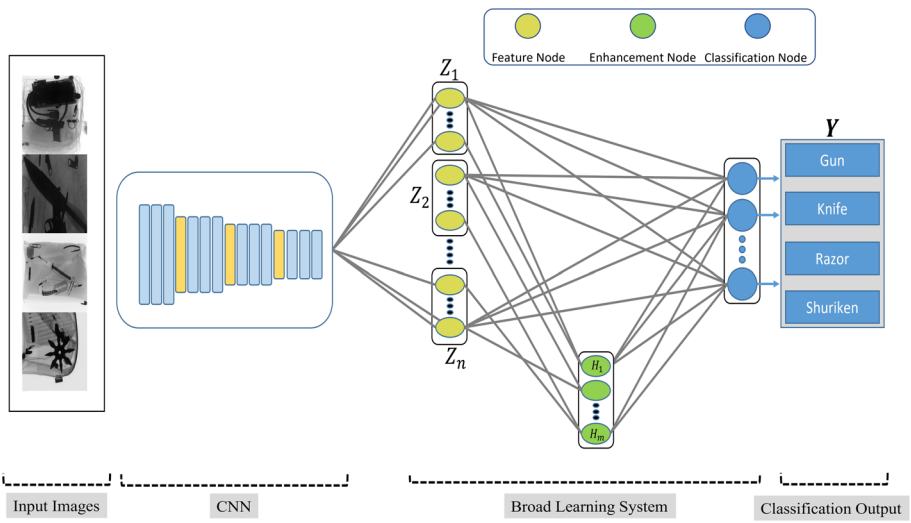

Fig. 1 Proposed system block diagram. CNN works as feature extractor while BLS works as a classifier in the proposed methodology

3.1 Broad Learning System (BLS)

BLS is a flat network that maps the original input into an array of feature nodes [32]. This first structure is extended into a second structure named enhancement nodes. The nodes in both structures are mapped to the output classification layer. The system is trained to determine the optimal parameters (i.e. weights) of the connections mapping the feature nodes and enhancement nodes to the classification layer. Given the training data set $(X, Y) \mid X \in R^{N\times M}$, $Y \in R^{N\times C}$, N, M, C are the number of training samples, the size of the training sample, and the number of classes, respectively. The feature nodes Z_i, and the enhancement nodes H_j are calculated through the following projection system:

$$Z_i = \phi_i(XW_{fi} + \beta_{fi}), i = 1, 2, 3, ...n \tag{1}$$

$$H_j = \xi_j(Z^n W_{hj} + \beta_{hj}), j = 1, 2, 3, ...m \tag{2}$$

$$Z^n = [Z_1, Z_2, ...Z_n] \tag{3}$$

where ϕ_i is a mapping function, and ξ_j is an activation function, (W_f, B_f) and (W_h, B_h) are the random weights, and bias values used to generate the mapped features and the enhancement features, respectively. The mapping function ϕ_i is a transformation function associated with a sparse autoencoder optimized based on a lasso regressor. The size p of the feature nodes Z_i is controlled by the size of the random weight W_f. The output of the BLS is given by:

$$\begin{aligned} Y &= [Z_1, Z_2, ...Z_n, H_1, H_2, ...H_m]W \\ &= [Z^n, H^m]W \\ &= AW \end{aligned} \tag{4}$$

where A represents the transformation features and W is the weight matrix of the connections trussing the feature nodes and enhancement nodes to the output layer. W is optimized by solving the following minimization

$$\min_W ||Y - AW||_2^2 + \lambda||W||_2^2 \tag{5}$$

where the first term denotes the training error, and the second term is a regularization term controlling the complexity and over-fitting. λ is the trade-off regularization parameter. A solution of the equation (5) can be formulated as

$$W = (A^T A + \lambda I)^{-1} A^T Y \tag{6}$$

If λ is set to 0, then this problem degenerates to the least square problem and leads the solution to the original pseudo-inverse solution, whereas, on the other end, when λ approaches infinity, the solution tends to the null vector. Practically, λ is set empirically to a small value, and the above equation is solved iteratively using a sparse autoencoder based on lasso regression [33].

3.2 BLS optimization

BLS consists of five trainable parameters namely, feature nodes weight matrix and bias (W_f, β_f), enhancement nodes weight matrix (W_h,β_h) and collective weight matrix W. In addition, the number of windows of features nodes n, the size of feature node window p, and the number of enhancement nodes m are the hyper-parameters we want to optimize. There are two types of connections in the BLS, W_f that maps the input into feature nodes,

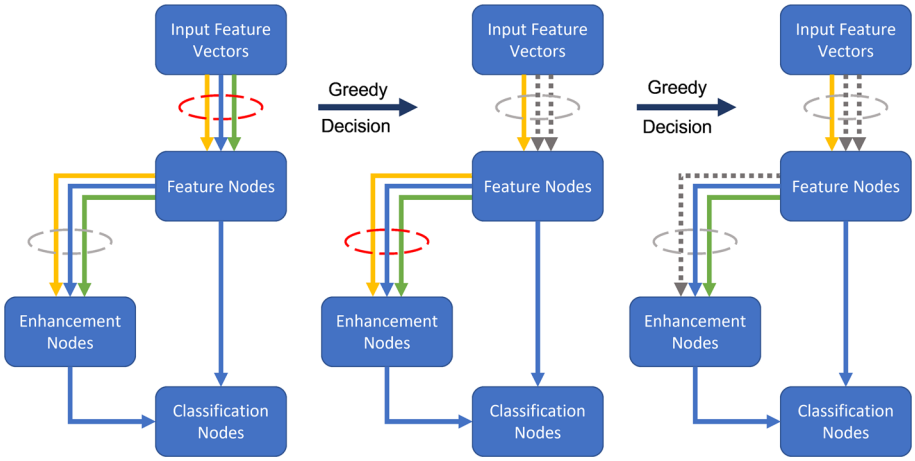

Fig. 2 A representation of the greedy-based selection method. Red-colored encircling the connections shows the existing greedy epoch. The greedy choice, where a connection is either maximized or repressed, is determined based on the selection criteria. The dashed line illustrates the suppressed connection, while other colors represent the selected connections in the right-most figure, which comes out as the final architecture

and W_h, which maps the feature nodes into the enhancement nodes. We proposed to search for the optimal links among the mentioned connections by employing a greed-based search algorithm to further strengthen the connections between input and output feature vectors. The algorithm can be seen in figure 2. Rather than determining a fixed optimal number, the algorithm starts with a fixed number of hyperparameters and searches its way through the pre-built links in an optimized manner, and prunes the rest of the connections. Greedy search is a technique to find a local optimum by employing a heuristic at each level. It finds its use in several graph-based algorithms such as Dijkstra shortest path algorithm [34], maximum spanning tree [35], etc. The solution in the greedy algorithm is assembled bit by bit as it makes the selection by weighing out the most apparent and critical benefit, leading to a globally optimum solution by making the locally optimum choices. Furthermore, because this method does not use back-propagation and is just a feed-forward algorithm, the decisions made cannot be changed after they have been made.

In this work, as shown in Algorithm 1, we propose a novel methodology to employ the greedy algorithm on BLS to find the optimal internal connections. To implement the greedy algorithm, a selection criterion is required that will make the most optimal choices on a sublayer level. The proposed criteria is based on the between input and output based upon the sparse autoencoder optimization. Afterward, the greedy selection can be made based on predefined selection criteria to further strengthen the connections between input and output feature vectors and enhance the performance of the system.

3.3 Selection criteria

To implement the greedy algorithm, a selection criterion is required that will make the optimal choices on a sublayer level. An illustration of how selection criteria are calculated is depicted in figure 2. There are two unique selection criteria that are employed for this purpose, as discussed below;

Algorithm 1 Greedy Search based BLS Optimization.

1: **Begin**
2: Initialize the system weights, W_f and W_h.
3: Fine tune the random weights via sparse-autoencoder optimization.
4: **for** $a = 1$ to n **do**
5: Evaluate the selection criteria based upon the set of nodes.
6: Based on selection criteria, determine the weight by replacing the
7: weight with argmax(W_{fa}) for feature nodes.
8: Suppress the weights with lower criteria by replacing the weights with
9: W_{fa}
10: **end for**
11: **for** $b = 1$ to m **do**
12: Evaluate the selection criteria based upon the set of nodes.
13: Based on selection criteria, determine the weight by replacing the
14: weight with argmax(W_{hb}) for enhancement nodes.
15: Suppress the weights with lower criteria by replacing the weights with
16: W_{hb}.
17: **end for**
18: Generate the broader weights using the optimized weights.
19: Optimize the broader weights using Ridge Regression Optimization.
20: **End**

(i) Edge Importance (EI): Edge importance is used for searching for the operations from the search space, where each operation represents some function [36]. This addresses the scalability challenge of architecture search by formulating the task in a differentiable manner. In the case of BLS, the same technique can be applied to grade the weighted connection. It shows the significance of a particular connection or a node as it evaluates the softmax probability of each node with the rest to the total number of nodes. EI is evaluated using the equation below;

$$EI_{\alpha,\beta} = \frac{e^{w_{\alpha,\beta}}}{\sum_{k=1}^{O} e^{w_{\alpha,k}}} \tag{7}$$

where O is the total number of connections of one node with the connecting nodes. In the case of example provided in figure 2, m is representing the total number of connecting nodes.

(ii) Selection Certainty (SC): The uncertainty measurement in a distribution is SC, and it is evaluated using the EI. It is used along with EI in neural architecture search [37] for searching the suitable operation. In the case of BLS, the same can be employed to calculate the connections entropy.

$$SC_{\alpha,\beta} = 1 - \frac{-EI_{(\alpha,\beta)} log(EI_{(\alpha,\beta)})}{|log(O) - 1)|} \tag{8}$$

(iii) Criteria: Edge importance tells about the implications of a particular weight in selecting the node, while SC indicates the correctness of the chosen weight. Hence, the selection criterion is developed on the basis that the edge with higher Edge Importance and higher Selection Criteria will be selected while the rest of the connections will be suppressed. The normalization applied is a min-max normalization.

$$Criteria_{\alpha,\beta} = normalize(EI_{\alpha,\beta} * SC_{\alpha,\beta}) \tag{9}$$

The decision of enhancing or pruning a connection is made upon comparing to criteria value in equation 9 to a pre-defined threshold. Connection with Criteria value higher than the set

threshold will be replaced by the maximum value of the connection, while the rest will be pruned.

4 Experimental setup

4.1 Datasets

We tested our approach on three publicly available datasets: GDXray, SIXray, and COMPASS-XP. In the next subsections, we will go through these datasets in further depth.

4.1.1 GDXray

Grima Dataset Xray (GDXray) [27] dataset was introduced in 2015 for object detection in X-ray scans. This dataset is divided into five categories. Besides, there are a total of 19,407 x-ray scans, out of which only one category out of five can be used for treat-object detection.

4.1.2 SIXray

Security Inspection X-ray (SIXray) [6] dataset is one the biggest publicly available dataset for the detection of contraband items in extremely obscure and cluttered baggage X-ray imagery. This dataset is known for introducing a class imbalance as in the total of approximately 1 million X-ray scans (1,059,231), only 8,929 scans are marked as positive, i.e. these scans consist of contraband items, while the rest of the scans are negative. There are six major categories of prohibited items, namely, guns, knives, pliers, wrenches, scissors, and hammers. In addition, for the classification into two categories, positive and negative scans, we used all positive scans, while for negative scans, we used 89,000 images to mimic SIXray10. Furthermore, an 80/20 split is used for training and testing purposes.

4.1.3 COMPASS-XP

This dataset was introduced in 2019 and consists of 501 object types [38]. In each scan, the dataset includes synchronized photographic and X-ray imagery that represents a single object. Contrary to the aforementioned datasets, the COMPASS-XP dataset is intended for image classification in which the ground truths are labeled scan-by-scan to indicate the presence of a common or harmful object in each scan. From the total of 11568 scans, we split the dataset into an 80/20 ratio so that there are 9,254 images for training and 2,314 for testing purposes.

4.2 Implementation details

The proposed system is implemented using MATLAB R2021a with deep learning and computer vision toolbox on a machine with an Intel Core i9-10940@3.30GHz processor, 132 GB RAM, and NVIDIA Quadro RTX 6000 GPU. The utilized optimizer for the framework during the training was stochastic gradient descent [39], with a number of epochs, batch size, and learning rate of 20, 128, and 0.001, respectively. [1]

[1] The code used for implementing the proposed system will be made available upon acceptance of the paper.

4.3 Performance metrics

To evaluate the presented technique and compare it to state-of-the-art works, the framework has been tested utilizing a variety of evaluation metrics such as Accuracy, Precision, Recall, Average True Positive (AvTP), and F1-score.

5 Results

This section presents the results acquired through the proposed framework's evaluation of three publicly available datasets. Moreover, we report an ablation study through which we determined the optimal covnet backbone and the backbone layer employed as a feature extractor. Furthermore, we report a detailed comparative analysis.

5.1 Ablation study

We conducted analytical experimentation that aims to investigate the performance of the proposed system with respect to: (i) The convnet backbone and (ii) The backbone layer employed as a feature extractor. In general, as the depth of the CNN model increases, the complexity of features learned by convolution layers increases. Therefore, for this experiment, middle and deep layers are considered, along with benchmarking them on the GDXray, SIXray, and the COMPASS-XP. Moreover, VGG19 [40] and ResNet50 [41] have been utilized to perform the test. Results are reported in Tables 1 and 2 showing the performance metrics obtained for the ResNet50 and the VGG19, respectively. We can see here that for ResNet50, the best performances spread over the last six layers. Besides, the highest accuracy, F1-score, and AvTP have been obtained when the features are extracted from the last layer for the GDXray, whereas, for both COMPASS-XP and SIXray, the average pooling layer is the optimum choice. Furthermore, it can be seen that in Table 2 the best performances are confined in the last four layers. For GDXray and COMPASS-XP, the accuracy, F1-score, and AvTP of the last layer FC7 have surpassed the other layers. However, for the SIXray, the first convolution layer has obtained the highest performance. In addition, the distributions of the best performance across the three datasets indicate a better generalization of the VGG19. We also notice that in this model, the FC7 layer exhibits the best performance for the accuracy, F1-score, and the AvTP metrics, across the three datasets. The above observations concur towards confirming the VGG19 model with the FC7 as the layer from which the features are being extracted is the best combination.

5.2 Comparative analysis

We analyzed the performance of the proposed framework as compared with other existing works. As mentioned in the ablation study, we concluded that for the best results, the backbone used for feature extraction should be VGG19, and the layer from which the features are being extracted should be the FC-7 layer. Besides, the BLS hyper-parameters settings are set to have 30, 600, and 30 feature nodes, enhancement nodes, and the size of feature nodes, respectively. This section discusses the results as well as presents a comparative analysis of the results from the presented methods with the current framework in terms of the three publicly available datasets.

Table 1 Model performance evaluation for the different feature maps of the ResNet50

ResNet50 Layer	GDXray Accuracy	F1	AvTP	SIXray Accuracy	F1	AvTP	COMPASS-XP Accuracy	F1	AvTP
res2a_branch2b	0.9681	0.9705	0.9710	0.9558	0.8286	0.7610	0.8686	0.5293	0.5323
res2c_branch2c	0.9590	0.9603	0.9608	0.9616	0.8568	0.7933	0.8674	0.5397	0.5380
res3c_branch2c	0.9863	0.9874	0.9874	0.9690	0.8895	0.8345	0.8807	0.6344	0.6031
res4c_branch2c	0.9894	0.9901	0.9911	0.9722	0.9030	0.8537	0.8980	0.7159	0.6723
res4e_branch2c	0.9939	0.9947	0.9949	0.9755	0.9166	0.8727	0.9017	0.7308	0.6863
res4f_branch2c	0.9848	0.9853	0.9872	0.9805	0.9351	0.8990	0.8988	0.7222	0.6792
res5b_branch2a	0.9878	0.9882	0.9898	*0.9829*	*0.9456*	*0.9271*	*0.9063*	*0.7503*	*0.7063*
res5b_branch2c	0.9894	0.9896	0.9911	0.9805	0.9384	0.9240	0.8948	0.7126	0.6723
res5c_branch2b	0.9894	0.9901	0.9911	0.9717	0.9034	0.8584	0.8980	0.7250	0.6841
avg_pool	*0.9924*	*0.9926*	*0.9936*	**0.9866**	**0.9574**	**0.9393**	**0.9112**	**0.7703**	**0.7282**
FC1000	**0.9939**	**0.9944**	**0.9949**	0.9043	0.4752	0.5001	0.8663	0.4663	0.6833

Bold shows the best performance while blue in italics shows the second highest

Table 2 Model performance evaluation for the different feature maps of the VGG19

VGG19 Layer	GDXray Accuracy	F1	AvTP	SIXray Accuracy	F1	AvTP	COMPASS-XP Accuracy	F1	AvTP
Conv1_2	0.9498	0.9520	0.9536	**0.9621**	**0.8603**	**0.7994**	0.8646	0.4954	0.5136
Conv2_2	0.9863	0.9872	0.9885	*0.9615*	*0.8571*	*0.7952*	0.8790	0.6122	0.5857
Conv3_3	0.9833	0.9853	0.9856	0.9593	0.8465	0.7819	0.8896	0.6671	0.6274
Conv4_1	0.9818	0.9836	0.9843	0.9572	0.8366	0.7713	0.8939	0.6859	0.6426
Conv4_4	0.9863	0.9873	0.9874	0.9531	0.8147	0.7458	0.9026	0.7348	0.6905
Conv5_2	0.9909	0.9920	0.9923	0.9527	0.8125	0.7434	0.9159	0.7732	0.7227
Conv5_4	0.9894	0.9907	0.9911	0.9496	0.7976	0.7290	0.9107	0.7621	0.7161
FC6	*0.9924*	*0.9928*	*0.9936*	0.9490	0.7918	0.7216	*0.9435*	*0.8633*	*0.8252*
FC7	**0.9939**	**0.9940**	**0.9949**	0.9485	0.7901	0.7208	**0.9516**	**0.8841**	**0.8472**

Bold shows the best performance while blue in italics shows the second highest

Table 3 Performance Results on GDXray dataset

Approaches	Accuracy	Precision	Recall	F1-Score
DSTD [43]	0.9554	0.9305	0.9761	.9598
CST [3]	0.9683	.9890	0.8856	0.9178
F-RCNN [44]	.9840	-	0.9800	0.9543
AISM [42]	-	0.9500	**0.9975**	-
Optimized BLS	**0.9939**	**0.9932**	.9949	**0.9940**

The best and the second-best performance is bold and underlined, respectively. In addition, the uncomputed metric is indicated by '-'

5.2.1 GDXray

In this section, we provide a comprehensive evaluation of our proposed system's performance on the GDXray dataset and compare it with state-of-the-art approaches. As shown in Table 3, our framework, which combines features from the Convolutional Neural Network (CNN) with the Broad Learning System (BLS), demonstrates superior results. Specifically, we have achieved a 3.56%, 0.996%, and 0.425% improvement over the state-of-the-art scheme in terms of F1-score, accuracy, and precision, respectively. This substantial enhancement in performance can be attributed to the synergy between the CNN features and BLS. The former adeptly extracts intricate features from the input data, while the latter efficiently uses these features to make precise and accurate predictions. Furthermore, the integration of the greedy search strategy has proven instrumental in improving the system's adaptability, contributing significantly to the observed performance. However, we acknowledge that our approach slightly trails the AISM [42] approach by 0.26% in terms of recall. We can conclude from these results that the performance of the proposed BLS on the GDXray dataset is quite appreciable.

5.2.2 SIXray

The assessment of the suggested method on the SIXray dataset is reported in Table 4. Here, it should be noted that the SIXray10 subset has been utilized only for this experiment.

Table 4 Performance Results on SIXray dataset

Approaches	Accuracy	Precision	Recall	F1-Score
Approaches	**Accuracy**	**Precision**	**Recall**	**F1-Score**
ResNet50+CST [3]	0.7685	-	-	**-**
ResNet50+CHR [6]	0.7794	-	-	-
Inceptionv3+CHR [6]	0.7949	-	-	-
DenseNet+CHR [6]	0.7956	-	-	-
SXMNet [46]	-	**0.9678**	-	-
HRNet+BA [45]	.9172	-	-	.8986
Optimized BLS	**0.9614**	.9582	0.8804	**0.9140**

The best and the second-best performance is bold and underlined, respectively. In addition, the uncomputed metric is indicated by '-', also BA stands for Balanced Affinity

Table 5 Performance Results on COMPASS-XP dataset

Approaches	F1-Score	Accuracy	Recall	Precision
ResNet50 [41]	0.5930	0.9161	0.8446	0.4569
VGG-19 [40]	0.7765	.9418	0.7977	.7565
MobileNet [47]	0.5917	0.9121	**0.9238**	0.4763
HRNet+Balanced Affinity [45]	.8714	0.8916	-	-
Optimized BLS	**0.8827** $\pm$ **0.0029**	**0.9506** $\pm$ **0.0012**	.8490 $\pm$ .0031	**0.9278** $\pm$ **0.0035**

The best and the second-best performance is bold and underlined, respectively

Also, It is worth noting that the available state-of-the-art approaches did not evaluate using the same set of metrics as us. Therefore, to ensure a fair and comprehensive comparison, we present the results in Table 4. It is noticed that the suggested technique has attained an accuracy of 96.14%, leading the state-of-the-art method, namely HRNet injected by Balanced Affinity loss function [45] by 4.819%. Moreover, it can also be seen that our framework outperformed the second-best approach by achieving an F1-score of 91.40%. In terms of precision, the proposed method slightly lags behind by 1.00%. In addition, the proposed framework performance has reached up to 88.04% in terms of recall.

5.2.3 COMPASS-XP

To assess the effectiveness of our proposed system, we have compared its performance with different baseline methods in terms of the COMPASS-XP dataset, and this is because there are no state-of-the-art methods for this aspect. Furthermore, Table 5 emphasizes the results of the proposed framework and the other baseline models, namely, ResNet50 [41], VGG19 [40], MobileNet [47], and HRNet [45]. It can be shown that, while the performance of multiple baseline models and the suggested BLS architecture is competitive in terms of accuracy and recall, the proposed model has achieved slightly less than the best scheme. However, regarding the F1-score, which is a better criterion than accuracy on unbalanced data, the suggested system surpassed the second-best technique by 1.30%. Similarly, the presented method outperforms the state-of-the-art approach by 22.64% in terms of precision.

5.3 Statistical significance

To assess the statistical significance of the performance improvements observed with our proposed Optimized BLS system, we performed a t-tests comparing the performance of our system against the various state-of-the-art approaches for each of the datasets used in our study. The methodology for performing these t-tests was guided by the approach outlined by Qureshi et al. [48]. The results of these t-tests yielded a p-value of 0.003, which is well below the conventional significance level of 0.05. This low p-value indicates that the differences in performance between our Optimized BLS system and the other models are statistically significant. Hence, we reject the null hypothesis that there is no significant difference in performance between our model and the state-of-the-art approaches. This reinforces our confidence in the efficacy of the proposed system and its superior performance in the task of baggage scanning.

6 Conclusion

In this work, we have presented a novel Programmable BLS framework for identifying contraband items, which we applied to three different datasets. This innovative system, unlike traditional BLS models, is auto-tuned during run-time via greedy searches, achieving optimal threat detection performance during the inference stage without the need for human intervention. Notably, this marks a groundbreaking attempt to address the problem of threat detection using a broad learning system. We evaluated our proposed framework using different metrics and it demonstrated superior performance in comparison to existing methods. Particularly, our system has shown a significant enhancement in detecting and recognizing concealed and highly imbalanced prohibited items in cluttered X-ray imagery.

7 Limitations and future work

Despite these promising outcomes, this study is not without its limitations. The approach, while novel, is in its nascent stages and therefore requires further refinement and validation. These limitations, however, present opportunities for future work. We plan to further investigate the potential of the proposed system in solving recognition-related tasks across a wider range of applications. We also aim to conduct a more comprehensive analysis of the system's capabilities, especially in terms of its scalability and robustness in handling more diverse and challenging scenarios.

Acknowledgements This work is supported by a research fund from Khalifa University, Ref: CIRA-2019-047 and CIRA-2021-052, and the Abu Dhabi Department of Education and Knowledge (ADEK), Ref: AARE19-156.

Data Availability The proposed method has been tested on three X-ray threat detection datasets, on which all of these three datasets are publicly available.

Declarations

Conflicts of interest All the authors declare that there are no competing interests related to this article.

References

1. Council, N.R.: Airline Passenger Security Screening: New Technologies and Implementation Issues. The National Academics Press (1996)
2. Air Transport, passengers carried. The World Bank. https://data.worldbank.org/indicator/IS.AIR.PSGR
3. Hassan, T., Bettayeb, M., Akçay, S., Khan, S., Bennamoun, M., Werghi, N.: Detecting Prohibited Items in X-ray Images: A Contour Proposal Learning Approach. 27th IEEE International Conference on Image Processing (ICIP) (2020)
4. Akçay 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 Transactions on Information Forensics and Security 13(9):2203–2215
5. Tao, R., Wei, Y., Li, H., Liu, A., Ding, Y., Qin, H., Liu, X.: Over-sampling De-occlusion Attention Network for Prohibited Items Detection in Noisy X-ray Images. Under Review in IEEE Transactions on Multimedia (2021)
6. 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 (2019)
7. Hassan, T., Werghi, N.: Trainable Structure Tensors for Autonomous Baggage Threat Detection Under Extreme Occlusion. Asian Conference on Computer Vision (ACCV) (2020)
8. Hassan, T., Shafay, M., Akçay, S., Khan, S., Bennamoun, M., Damiani, E., Werghi, N.: Meta-Transfer Learning Driven Tensor-Shot Detector for the Autonomous Localization and Recognition of Concealed Baggage Threats. MDPI Sensors (2020)
9. Baştan, M., Yousefi, M.R., Breuel, T.M.: Visual words on baggage x-ray images. Computer Analysis of Images and Patterns Lecture Notes in Computer Science, 360–368 (2011). https://doi.org/10.1007/978-3-642-23672-3_44
10. Megherbi, N., Flitton, G.T., Breckon, T.P.: A classifier based approach for the detection of potential threats in ct based baggage screening. In: Proceedings - International Conference on Image Processing, ICIP, pp. 1833–1836 (2010). https://doi.org/10.1109/ICIP.2010.5653676
11. Wang, Q., Bhowmik, N., Breckon, T.P.: On the Evaluation of Prohibited Item Classification and Detection in Volumetric 3D Computed Tomography Baggage Security Screening Imagery (2020)
12. Wang, Q., Bhowmik, N., Breckon, T.: Multi-class 3d object detection within volumetric 3d computed tomography baggage security screening imagery. (2020)
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. Proceedings of the 28th ACM International Conference on Multimedia (2020). https://doi.org/10.1145/3394171.3413828

14. Akçay, S., Breckon, T.: Towards Automatic Threat Detection: A Survey of Advances of Deep Learning within X-ray Security Imaging. http://arxiv.org/abs/2001.01293arXiv:2001.01293 (2020)
15. Mery D, Svec E, Arias M, Riffo V, Saavedra JM, Banerjee, S.: Modern Computer Vision Techniques for X-Ray Testing in Baggage Inspection. (2017) IEEE Transactions on Systems. Man, and Cybernetics: Systems 47(4):682–692
16. Zhang, J., Zhang, L., Zhao, Z., Liu, Y., Gu, J., Li, Q., Zhang, D.: Joint shape and texture based x-ray cargo image classification. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 266–273 (2014). https://doi.org/10.1109/CVPRW.2014.48
17. Turcsany, D., Mouton, A., Breckon, T.P.: Improving feature-based object recognition for x-ray baggage security screening using primed visualwords. 2013 IEEE International Conference on Industrial Technology (ICIT) (2013). https://doi.org/10.1109/icit.2013.6505833
18. Jaccard, N., Rogers, T.W., Morton, E., Griffin, L.D.: Using deep learning on x-ray images to detect threats. (2016)
19. Riffo V, Mery D (2016) Automated detection of threat objects using adapted implicit shape model. IEEE Transactions on Systems, Man, and Cybernetics: Systems 46(4):472–482. https://doi.org/10.1109/TSMC.2015.2439233
20. Bastan, M., Yousefi, M.R., Breuel, T.: Visual words on baggage x-ray images. In: CAIP (2011)
21. Bay H, Ess A, Tuytelaars T, Van Gool L (2008) Speeded-up robust features (surf). Comput. Vis. Image Underst. 110(3):346–359. https://doi.org/10.1016/j.cviu.2007.09.014
22. Huijuan, Z., Qiong, H.: Fast image matching based-on improved surf algorithm. In: 2011 International Conference on Electronics, Communications and Control (ICECC), pp. 1460–1463 (2011). https://doi.org/10.1109/ICECC.2011.6066546
23. Heitz, G., Chechik, G.: Object separation in x-ray image sets. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 2093–2100 (2010). https://doi.org/10.1109/CVPR.2010.5539887
24. Mery, D., Svec, E., Arias, M.: Object recognition in baggage inspection using adaptive sparse representations of x-ray images. Image and Video Technology Lecture Notes in Computer Science, 709–720 (2016). https://doi.org/10.1007/978-3-319-29451-3_56
25. Kundegorski, M.E., Akcay, S., Devereux, M., Mouton, A., Breckon, T.P.: On using feature descriptors as visual words for object detection within x-ray baggage security screening. In: 7th International Conference on Imaging for Crime Detection and Prevention (ICDP 2016), pp. 1–6 (2016). https://doi.org/10.1049/ic.2016.0080
26. Akcay, S., Atapour-Abarghouei, A., Breckon, T.P.: GANomaly: Semi-Supervised Anomaly Detection via Adversarial Training (2018)
27. Mery, D., Riffo, V., Zscherpel, U., Mondragón, G., Lillo, I., Zuccar, I., Lobel, H., Carrasco, M.: Gdxray: The database of x-ray images for nondestructive testing. Journal of Nondestructive Evaluation **34**(4) (2015). https://doi.org/10.1007/s10921-015-0315-7
28. An, J., : Semantic segmentation for prohibited items in baggage inspection. In: Int. Conf. Intelligence Science and Big Data Engineering. Visual Data Engineering, pp. 495–505 (2019)
29. Jin J-W, Chen CP (2018) Regularized robust broad learning system for uncertain data modeling. Neurocomputing 322:58–69
30. Jin J, Liu Z, Chen CLP (2018) Discriminative graph regularized broad learning system for image recognition. Science China Information Sciences 61(11):112209. https://doi.org/10.1007/s11432-017-9421-3
31. Kong, Y., Wang, X., Cheng, Y., Chen, C.L.P.: Hyperspectral imagery classification based on semi-supervised broad learning system. Remote Sensing **10**(5) (2018). https://doi.org/10.3390/rs10050685
32. Chen CP, Liu Z (2017) Broad learning system: An effective and efficient incremental learning system without the need for deep architecture. IEEE transactions on neural networks and learning systems 29(1):10–24
33. Chen CLP, Liu Z (2018) Broad learning system: An effective and efficient incremental learning system without the need for deep architecture. IEEE Transactions on Neural Networks and Learning Systems 29(1):10–24. https://doi.org/10.1109/TNNLS.2017.2716952
34. Golden B (1976) Shortest-path algorithms: A comparison. Operations Research 24(6):1164–1168
35. Ozeki K, Yamashita T (2011) Spanning trees: A survey. Graphs and Combinatorics 27(1):1–26
36. Liu, H., Simonyan, K., Yang, Y.: Darts: Differentiable architecture search. arXiv preprint http://arxiv.org/abs/1806.09055arXiv:1806.09055 (2018)
37. Li, G., Qian, G., Delgadillo, I.C., Muller, M., Thabet, A., Ghanem, B.: Sgas: Sequential greedy architecture search. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1620–1630 (2020)
38. Griffin: COMPASS-XP. https://zenodo.figshare.com/articles/dataset/COMPASS-XP/9249791/1

39. Kingma, D.P., Ba, J.: ADAM: A Method for Stochastic Optimization. International Conference for Learning Representations, 2015
40. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: International Conference on Learning Representations (2015)
41. He, K., Zhang, X., Ren, S., Sun, J.: Deep Residual Learning for Image Recognition (2015)
42. Riffo V, Mery D (2016) Automated detection of threat objects using adapted implicit shape model. IEEE Transactions on Systems, Man, and Cybernetics: Systems 46(4):472–482. https://doi.org/10.1109/TSMC.2015.2439233
43. Hassan, T., Shafay, M., Akçay, S., Khan, S., Bennamoun, M., Damiani, E., Werghi, N.: Meta-Transfer Learning Driven Tensor-Shot Detector for the Autonomous Localization and Recognition of Concealed Baggage Threats. Multidisciplinary Digital Publishing Institute (2020). https://www.mdpi.com/1424-8220/20/22/6450
44. Jain, D.K., : An evaluation of deep learning based object detection strategies for threat object detection in baggage security imagery. pattern recognition letters 120, 112–119 (2019)
45. Ahmed, A., Velayudhan, D., Hassan, T., Bennamoun, M., Damiani, E., Werghi, N.: Highly imbalanced baggage threat classification. In: Proceedings of the 15th International Conference on Machine Learning and Computing (2023)
46. Hu, B., Zhang, C., Wang, L., Zhang, Q., Liu, Y.: Multi-label x-ray imagery classification via bottom-up attention and meta fusion. In: Proceedings of the Asian Conference on Computer Vision (ACCV) (2020)
47. Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H.: Mobilenets: Efficient convolutional neural networks for mobile vision applications. ArXiv **abs/1704.04861** (2017)
48. Qureshi SA, Rehman AU, Mir AA, Rafique M, Muhammad W (2022) Simulated annealing-based image reconstruction for patients with covid-19 as a model for ultralow-dose computed tomography. Frontiers in Physiology 12:2324