Automatic Threat Detection in Single, Stereo (Two) and Multi View X-Ray Images

Abhinav Tuli
Department of Computer Science
Birla Institute of Technology
and Science, Pilani
Pilani, Rajasthan
f20170048@pilani.bits-pilani.ac.in

Rohit Bohra
Department of Computer Science
Birla Institute of Technology
and Science, Pilani
Pilani, Rajasthan
f20170225@pilani.bits-pilani.ac.in

Tanmay Moghe
Department of Computer Science
Birla Institute of Technology
and Science, Pilani
Pilani, Rajasthan
f20170184@pilani.bits-pilani.ac.in

Nitin Chaturvedi
Department of Electrical Engineering
Birla Institute of Technology
and Science, Pilani
Pilani, Rajasthan
nitin80@pilani.bits-pilani.ac.in

Domingo Mery
Department of Computer Science
Pontificia Universidad
Católica de Chile
Santiago, Chile
domingo.mery@uc.cl

Dhiraj
Cognitive Computing Group
CSIR-Central Electronics Engineering
Research Institute,
Pilani, Rajasthan
dhiraj@ceeri.res.in

2020 IEEE 17th India Council International Conference (INDICON) | 978-1-7281-6916-3/20/$31.00 ©2020 IEEE | DOI: 10.1109/INDICON49873.2020.9342253

***Abstract*—Accurate X-ray screening systems are of paramount importance in the present day. Most existing systems predict only on the basis of a single image, which could lead to false positives and false negatives due to limited information present. We implemented several approaches using single, two and multiple X-Ray views to make a reliable and practical model with varying levels of success in threat object detection. These approaches include long-established methods such as Bag of Visual Words (BOVW), 3D Object Recognition, Adaptive Implicit Shape Model and Deep Neural Networks. The approaches took in dual inputs to make more informed predictions. Varying levels of success are obtained in these methods ranging from 73% using BOVW to 87% using Deep CNN. It was observed that, when two views of an object are considered, an improvement of 5% to 15% in performance took place (considering various approaches) compared to a single view.**

***Index Terms*—Threat Detection, X-Ray, Bag of Visual Words, Deep Learning, Adaptive Implicit Shape Model**

I. Introduction

In recent years, especially after the New York plane crashes in 2001, X-Ray screening systems are being used worldwide to protect and secure many places like airports, railway stations, malls, stadiums, government buildings, etc. The main aim of these screening systems is to detect dangerous objects such as knives, handguns and explosives in baggage items being carried by the public during their travel.

All over the world, this detection is done manually by security officers present at the physical location. This is a very demanding job and requires a lot of concentration by the security officers as only a minuscule percentage of bags would contain weapons or explosives. Some threat objects are bound to be missed by the elaborate security systems on account of human error. The detection rates of humans as well as computers are severely affected because of the complexity - all types of objects are present in the bags of passengers. High traffic further reduces the time spent by the security officers per person.

Other than airport baggage security, there are a lot of areas where X-Ray image inspection has been automated and has proven useful to the task like detection of diseases, even Covid-19, through the chest X-Ray of a patient, food quality inspection [1] and inspection of welds and cargo [2]. Considering all this, an automated system to detect the threat objects in passenger baggage and cargo would greatly streamline the process of airport security and reduce the workload of security officers. A lot of research has been going on towards automatic detection of threat objects [3] [4] [5]. Most of these methods improve the quality of the image through novel pseudo-colour algorithms and image segmentation and use methods like Bag of Visual Words(BOVW) [6], SVMs [7], etc. to detect threat objects. In all of these diagnoses, only a single view of an image is taken into consideration.

In all of the above approaches, the chances of misinterpretation are high as there is just a single view. However, nowadays two-view X-Ray machines are becoming more common. Having multiple views will always enhance the capability to detect a threat object compared to having a single view. The scope of this paper is to analyze and compare existing models for threat detection and develop an efficient automatic threat detection technique using single,dual or multiple views of an object. We have experimented with the methods of 3D Space Carving [8], BOVW [6], Dual Input CNN and Adaptive Implicit Shape Model [9] and analyzed the impact of multiple views vs single view in the process of threat detection and the feasibility to implement these considering the current

infrastructural bottlenecks.

A deep-learning-based multi-view approach for threat detection was proposed by Steitz et al [10] in 2018. A multi-view pooling layer was introduced in the network to perform a 3D aggregation of 2D image features. These features were extracted from all the views of the image (three in this case). This pooling layer also took into account the known features of common threat objects. Secondly, an end-to-end pipeline which was based on Faster-RCNN was introduced to perform the final detection using the previously extracted aggregated 3D features. There were significant improvements in the accuracy (10 to 30 percent depending on the object) compared to just single-view detection.

In order to detect prohibited items and not just threat objects, Maoshu Xu et al. proposed an approach [11] in 2018 based on enhancing the CNN classifier using a top-down attention mechanism. The attention maps are very useful in localizing the detected object. The experiments have been done on the GDXRay dataset [17] (same as the one used in this paper) and have proved successful in the detection of single and multiple targets.

Another tensor-based framework to detect normal, as well as threat objects, has been proposed by Hassan et al. [12] in 2019. Contour based translational information was extracted from different orientations and this information was used by just a simple feed-forward convolutional neural network for object detection. An accuracy of almost 99 percent was achieved in detecting normal as well as suspicious objects.

Among the various approaches we have proposed, one is a dual-CNN approach. This is a very promising approach in itself and yet there is a lot of scope for improvement. The above mentioned three papers show that with slight improvements in feature extraction and initial network structure, the already stellar accuracy achieved can be improved further.

In section II, we have analyzed the method of 3D space carving [8]. A concept 3D model is generated with the help of X-Ray images from multiple views and features are extracted from this model. Threat detection is done by comparing features of the 3D model. The feasibility of this method is also addressed. Following this, in section III, we have analyzed the BOVW method using multiple features extractors and compared the performance when a single view image is given as input and when two views are given. In section IV, we have designed various dual-input Convolutional neural network architectures on top of well known image classification networks like InceptionV3 [13], VGG19 [14], ResNet50 [15], etc. and reported their performance. In section V, we study the Adaptive Implicit Shape Method with two images fed simultaneously instead of one. Finally in section VI, we offer conclusions on our experiments.

II. 3D Object Recognition

This method was originally proposed by Riffo et al. in [16]. This method proposes the creation of 3D models of the threat object in the training stage and the extraction of descriptors from these. In the testing phase, 3D models of the bags of interest are created and their descriptors are extracted. Subsequently the two descriptors are compared to find matches. The 3D reconstruction in both the training and testing phases is done by using a voxel cutting technique called ”Space Carving” [8]. For both training and testing, we have used X-ray images selected from the publicly available GDXray database [17]. In the subsequent sections we have described the method in detail.

A. Training

For training purposes, X-Ray images of guns taken by rotating the gun across all the 3 axes were used [16]. Training using these images however required the projection matrices of each image, which wasn't available to us in the public domain. We have instead used two orthogonal images of guns taken from freely available online stock images for training. Using these we were able to train without projection matrices. The drawback to this approach however was the fact that the gun model that we used was different from the one present in the test bags.

Training consists of creating 3D model(Figure 1) from the images acquired using Space Carving. Space Carving works by carving out a 'virtual sculpture' from a cube shaped voxel array. The 3D object to be constructed is enclosed within this cube. Space Carving proceeds to iteratively carve/remove portions of the cube that do not match the silhouettes of the 2D images. At the end, the sculpture of the 3D model is left and the background has been carved out. Subsequently, we extract keypoints from the reconstructed object. These points are the locations where invariant features about the 3D model can be described. 3D local descriptors around these keypoints are then extracted using OpenCV library [22]. These are local characteristics that describe the object's characteristics and they must have some invariance, such as rotation or scaling. We separately experimented with four 3D descriptors, namely 3DSC [18], USC [19], FPFH [20] and SHOT [21].

Fig. 1. 3D model of gun generated from 2 gun images

B. Testing

For testing purposes, we used 19 distinct bags from GDXray that contained 0, 1 or 2 handguns. Each bag had 90 X-Ray images that were taken by rotating the vertical axis from 0 to 178 degrees. All these images were present along with their respective projection matrix. From these images, the three darkest contours were selected and the background was ignored. Doing this we were able to segment all objects of interest and ignore the insignificant objects. For each bag,

as we had 90 images with their projection matrices, complete space carving was done. As multiple objects are present in the bags, each significant object was clustered separately to obtain multiple 3D models from each bag. A generated 3D models for one bag are shown in Figure 2. For each object obtained, key points and descriptors were extracted, similar to training.

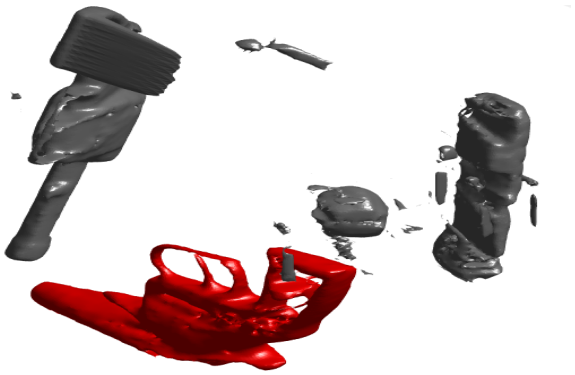

Fig. 2. Separate 3D objects obtained by clustering

The 3D descriptors of the obtained test models were compared with the descriptors of the training model. Euclidean distance [23] was used as the difference metric. The descriptors matched if Euclidean distance was smaller than a manually specified threshold. The Euclidean distance(d(p,q)) between n-dimensional vectors p and q) formula is given below.

$$d(p,q) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2} \tag{1}$$

C. Results

Across the 19 bags, we achieved an overall accuracy of 72.22% for gun detection using 3DSC Descriptor [18] which is less than the proposed accuracy in the original paper [16]) of 97%. We were unable to replicate the results shown in the original paper due to non availability of projection matrix data for the gun model. Despite this, the method demonstrated the advantages of using multiple views over a single view and gave us insights that we used in subsequent experiments. Here accuracy was defined as the numbers of guns detected out of all guns present in the nineteen bags. Some bags contained more than one guns giving a total number of 18 guns. As this system could potentially be deployed in real time systems, speed is another important factor to be considered. Using SHOT as the descriptor was the fastest, we were able to bring down the time per bag from 33 seconds in 3DSC to 18 seconds although the accuracy obtained went down to 66.67%. Below is a table showing accuracy for different descriptors. All these experiments were timed on a MacBook Pro 2019 Model. Refer to table I for complete overview on results for this method.

TABLE I
3D OBJECT RECOGNITION RESULTS

Metric	3DSC	SHOT	USC	FPFH
Accuracy	72.22%	66.67%	55.56%	55.56%
Time	33s	18s	42s	24s

III. BAG OF VISUAL WORDS

We use the Bag of visual words (BOVW) approach [6] to tackle this image classification problem. This novel technique is a modification to the Bag of Words(BOW) [24] approach, used in Natural Language Processing and information retrieval. In BOW approach, we count the number of words that appear in a document and then we use the frequency of each word to know the important keywords in the document. Then we create a frequency histogram from it. We treat each document as a bag of words [24].However, in BOVW approach, instead of words, we use image features as the words to create the frequency histogram. Image features are unique patterns that we can find in an image. Also, key-points are the "stand out" points in an image. Even if the image is rotated, shrinked, or expanded, its key-points will always be the same. The descriptor can be considered as the description of the key-point.

Following a similar work of Turcsany et al. [25], we use dual-view single-energy X-Ray Images. Both the front and side view of the image is considered. We use the GDXray Dataset [17]. We treat this problem as a standard binary classification problem. Our aim is to differentiate bags containing guns from bags without guns. We train our classifier using four different detectors:- SIFT [26], BRISK [27], SURF [28] and ORB [29].

A. Single Input Approach

1) Training: The training set consisted of 700 cropped gun images from GDXRay. These included 307 front views, 302 side views and 91 top and bottom views. Examples of side and bottom images are shown in Figure 5. We then extracted keypoints and descriptors for all the images in the dataset. Since the number of the key-point, descriptor pairs obtained for most images were more than 100, we applied k-means to obtain the visual codebook for the main keypoints (top 30) in the image. Once we are done with this across all images, we take an average of the 30 descriptors extracted from all images to essentially get a histogram of the average features of a gun. We find the Euclidean distance between the average histogram and histogram of each training image. We the manually set a threshold value such that the distance of the majority of training images lies below this.

2) Testing: The Test set consisted of twenty images of different bags(taken from GDXray). Ten of these bags contained 1 or more guns while the rest had no guns. These bags contains multiple objects and the guns were partially obstructed from view in many images. Figure 4 shows the two

Fig. 3. Side and Bottom-View Images of Gun from GDXray Dataset

TABLE II
BOVW SINGLE-VIEW RESULTS

Measure	SIFT	SURF	ORB	BRISK
Precision	54.54%	53.84%	41.6%	42.86%
Recall	60%	70%	50%	60%
F1 Score	0.57	0.61	0.45	0.50

types of bags used for this method. We apply a sliding window method on each bag individually to check if a portion of the bag has a gun or not. We take a window size of 256x256. Similar to training, the top 30 Keypoints and descriptors were then extracted from each window. We then calculated the Euclidean distance of that window's descriptors from the average histogram obtained in training. If the distance is less than the predetermined threshold, the model predicts the presence of a gun. Once this is done for all windows of a particular image, if all the distances were more than the threshold, the model predicts that no gun is present. Otherwise the model detects a gun.

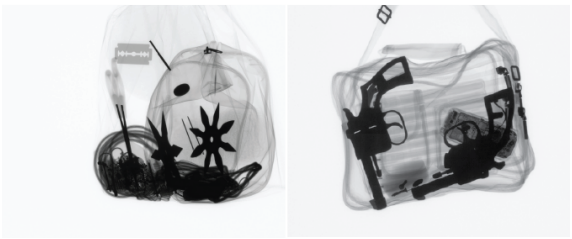

Fig. 4. Bags from GDXray Dataset

3) Results: The accuracy obtained using different descriptors is tabulated in table II.

B. Dual Input Approach

A single image of a bag is not sufficient to accurately make a prediction due to possible obstructions. Hence we used the same method with a two input approach.

1) Training: The same dataset mentioned in the Single Input Approach was used however the 91 images of the top and bottom view were removed. This was done because we were unable to get a accurate histogram for this because of limited number of top and bottom view images in our dataset. Similar to the Single Input Approach, we again created the visual codebook by calculating the average histogram, however this time we created 2 codebooks, one for the front view and one for the side view. This was done by segregating the dataset and calculating histograms separately.

TABLE III
BOVW DUAL-VIEW RESULTS

Measure	SIFT	SURF	ORB	BRISK
Precision	66.67%	62.5%	50%	50%
Recall	80%	50%	70%	60%
F1 Score	0.73	0.55	0.58	0.54

2) Testing: The Test set consisted of 2 orthogonal images from 20 Bags. 10 of these bags contained 1 or more guns while the rest had no guns. These bags contained multiple objects and the guns were partially obstructed from view in many images. Similar to Single Input Approach, we again used the sliding window method for each bag. However this time we took 2 images from each bag into consideration before making the prediction about the presence of a gun. Each window from both images was compared to both histograms for front and side view matching with the gun.

3) Results: The accuracy obtained using different descriptors is tabulated in table III.

Here also, we can conclude that SIFT gives the best overall result.

C. Limitations of the Approach

Figuring out the threshold to be used for gun detection is a manual process that requires a lot of trial and error. Moreover the same threshold might not give good results for images from a slightly different dataset. This method also gives a lot of false positives even for the two input model.This is because it simply detects a gun if even a single view detects a gun and does not verify it with the other view.

IV. DUAL INPUT CNN

A. Overview

With tremendous progress in the field of Deep Learning since the creation of the VGG network [14], we had to test if a deep neural network could give us a better result than the methods discussed above. We developed three dual-input architectures that used 2 orthogonal X-Ray images for detecting threats. In most past studies, dual images have been used with single input CNNs [30] in which 2 separate CNN outputs are combined to make a final prediction. Such models predict the presence of threat objects if it is detected in either of the views. These methods get better precision than the single input baseline method, however by not combining both images before making predictions, these models usually suffer from many false positives.

Our method in contrast, takes into account both the images before making predictions. This is done in order to reduce the possibility of false positives. Our model architectures were

built on top of InceptionV3, VGG19 and ResNet50. We have previously tested our models only on guns. Here we used Gun, Shuriken and Knife as our target labels. For all of these architectures instead of using a standard single input CNN, we used a dual input CNN.

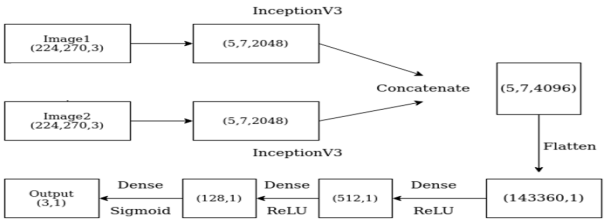

Fig. 5. Model Diagram of Dual Input CNN built on top of InceptionV3

B. Dataset

From the GDX-Ray dataset, we have X-Ray images of 19 bags, each of the bags having different threat objects - gun, blade, shuriken and knife. The dataset has images from 19 bags, 14 of them have guns, 10 of them have blades and 5 of them have shurikens present in them. For the scope of this problem, we are detecting whether the bag contains a gun or not, whether it contains a blade or not and whether it contains a shuriken or not. This problem is very specific, but it can be generalized to all types of threat objects on the acquisition of suitable datasets of dual-view X-Ray images. Each of these 19 bags have 180 images, each of the same bag but from a different angle. The angles range from 0 to 360 with an interval of 2 degrees. We have generated 180 pairs from the given 180 images. Thus we have a total of 19*180 i.e. a total of 3420 pairs. For training the network, we have used 174 pairs from 16 bags i.e. a total of 2784 pairs. For testing the network, we have used the 6 remaining pairs from these 16 bags and 30 pairs from the three remaining bags which the network has not seen, giving a total size of 186 bags. Figure 6 shows a pair of orthogonal images.

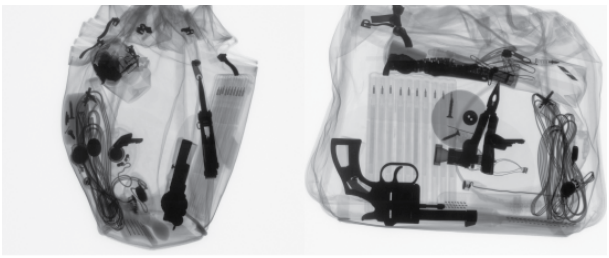

Fig. 6. A pair of orthogonal images

C. Baseline Model

Before experimenting with the dual input architecture, we implemented a vanilla Inception-v3 [13] architecture (with a single input) as our baseline model. We fed our network 2784 images for training. Then we applied our model on 186 images for testing (same number as dual-input, however single images taken instead of pairs). This CNN architecture gave us a decent baseline accuracy of around 65%.

TABLE IV
DUAL-INPUT CNN RESULTS

Baseline	InceptionV3	ResNet50	VGG19
65.09%	78.07%	87.72%	77.19%

D. Dual Input Model

Consider the Inception model - each of the two orthogonal input images was fed to one Inception-v3 module with its final connected layer omitted and all the weights fixed as those used in the 'imagenet' competition. These weights were frozen. The outputs of the modules were concatenated and two dense relu [31] layers of 512 and 128 neurons were added on top of that. Finally the sigmoid output with loss as 'binary-crossentropy' and optimizer as 'Adam' [32] had three neurons - each neuron giving the prediction whether the bag contains a gun, a blade or a shuriken. To illustrate this, if the output vector for a given test data set input is [1 0 1], this means that the bag contains a gun, does not contain a blade and contains a shuriken. Even if one of the predictions is zero, this means that a threat is detected. Same architecture and layer-parameters were used for the other two models (VGG19 and ResNet50). The models were run for a total of 50 epochs.

E. Results

Due to the limited dataset, accuracy is a slightly flawed metric but still the best way to gauge performance. Since all of the data points in the training set have threat objects, the job of the model is to accurately predict the presence of three predefined threat objects on the basis of which the models were trained. The result is correct only when the presence of each threat object is detected correctly. Considering this, the models have a validation accuracy between 75-90 percent in identifying the correct threats, which is promising since the test set also includes images of bags which the network has never seen before. Out of the three, the best accuracy is seen on the ResNet50 model. Since the number and types of threat objects is not very large (different types of guns, explosives, etc.), this method can be extrapolated to detect all types of threat objects in dual-view and if possible, multi-view X-Ray images. The exact results (the measure being accuracy) are formulated in the table IV.

V. ADAPTIVE IMPLICIT SHAPE MODEL

A. Overview

This approach is inspired from the work of Riffo and Mery in [9], because it has been reported in [33] that compared with other known methods it achieves high performance. The AISM (Adaptive Implicit Shape Model) is inspired from the famous Implicit Shape Model (ISM) approach [34]. The

implicit shape model works on two basic ideas - learning an appearance codebook and learning a star-topology structural model [35] i.e. the features are considered as independent given the object center. The model for a given object category consists of a codebook of local appearances that are representative for the given object category. The codebook can be thought of as a class-specific alphabet for the given category. The model also contains a spatial probabilistic distribution. This distribution specifies where (the displacement from the central position) each codebook entry (alphabet) may be found on the object (a word made from the alphabet). The probabilistic generalized Hough Transform algorithm [36] is used to detect images.

B. Training, Testing and Detection

For codebook generation, a training database of 200 images each of a handgun, a blade and a shuriken is used. These images are selected from GDXray database and the SIFT approach [26] is used to extract key-points. For each test image, probable centers of target objects are extracted from the image. These points are called interest points. Now each interest point is matched to the closest keyword in the codebook and a probability is calculated. This probability corresponds to whether the interest point corresponds to the center of mass of the target object. Now each matched interest point generated polls for specific instances of the given object category at different points on the image according to the learnt spatial probability distribution. After this step, many subwindows are collected and overlapping subwindows among these are merged. Each sub window over a pre-decided threshold probability is now selected. If no sub-window in the target image meets this conditional requirement, then no potential target object is detected. Such detection is carried out for each threat object (gun, blade and shuriken) separately. Figure 7 shows in detail the three stages during the sub-part of detection of a blade. Detection of a gun and shuriken is performed after this. All the training images for the codebook generation and the starter AISM Matlab code are available at Domingo Mery's university page.

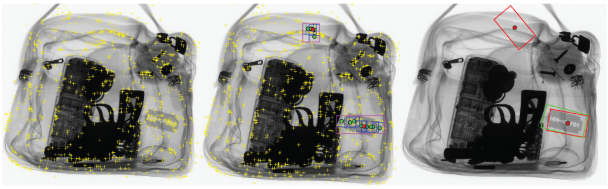

Fig. 7. Stage I - extraction of keypoints, Stage II - merging of overlapping subwindows, Stage III - final detection of blade

C. Results

We created a custom test set of 180 pairs of orthogonal images from the GDXray dataset. In our test set, each one of the target images has one or more of the three threats - gun, blade and shuriken. We are measuring the score in which the correct threat is detected at the correct position in the image. We test the target images for each of the three threat objects individually. A false detection is counted when the place where the threat is detected in the image has actually no threat object.

TABLE V
AISM RESULTS

Measure	Single-Input	Dual-Input
Accuracy	80.55%	91.66%
False Detections	43	60

In the 180 single view images, through AISM, threat objects were correctly classified in 145 images. However, the number of false detections were high, 43 of them were detected. In the 180 image pairs, AISM correctly detected the threats in 165 images. Along with this, the amount of false positives also increased. 60 false detections were recorded. Refer to table V for complete overview.

VI. CONCLUSION

In the desire to improve Threat Detection using X-Ray images, we explored 4 different approaches.

Although practically we can't completely automate the baggage screening process since human security is at stake, machines could help humans speed the process and make it more robust. We can improve human vision accuracy by using new models based on human perception and combining it with state-of-the-art Computer Vision Algorithms.

Before we draw conclusions on the approaches we used, we should also understand the need to have standard evaluation protocols to make fair comparisons among our different algorithms. In our paper, we have used precision,recall and accuracy to evaluate our models.

First, we experimented with a 3D object recognition approach. Due to the unavailability of projection matrices of gun images, we created our own implementation of 3D space carving. Using this, we generated gun models using only two gun images and matched them with objects in the bag under inspection to detect guns.

Secondly, using the Bag of Visual Words Technique, we made a classifier to detect guns in a bag. We experimented with both single view and two-view approaches and got much better precision and recall for the two-view approach. However, even the dual input approach resulted in false positives as both views were being considered separately and no combined inferences were taken.

Next, to take proper advantage of both views, we shifted our focus to CNNs. We experimented with a variants of Inception-V3, VGG-19 and ResNet50 models with transfer learning. We used a Dual Input CNN model with custom top layers to consider two views of an Object. For this process, we segregated orthogonal pairs from the original dataset. The model built on top of ResNet50 performed the best.

Finally, we discussed the AISM approach, which performs better in terms of detecting threats when we use dual-view instead of single-view. Accurate threat detection increased from 80% to 91% on moving from single view to the two-view approach. Even the dual-input mode resulted in a high number of false positives in the detections.

Depending upon the feasibility and applications, each method has it's pros and cons. 3D object recognition can give excellent results if we have a versatile dataset of threat objects and we have the ability to take a high number (ideally 180) of different views of the bag under inspection. This however is not possible in real life scenarios. BOVW approach takes less training time however it doesn't give very good results on single as well dual input images. Also, the method to estimate classification threshold is manual. If we can automate this process and provide more training classes(instead of just front and side views), we can expect better results. AISM gives decent results but too many false postives which can hamper our main purpose of automating threat detection which was to reduce waiting time at security checks. Out of all approaches, Dual Input CNN was the most promising and unlike other methods, it successfully took into account the two views and gave better results compared to a standard single input CNN.

REFERENCES

[1] Brosnan, Tadhg, and Da-Wen Sun. "Improving quality inspection of food products by computer vision—a review." Journal of food engineering 61.1 (2004): 3-16.

[2] Mery, Domingo. "Computer vision technology for X-ray testing." Insight-non-destructive testing and condition monitoring 56.3 (2014): 147-155.

[3] Mery, D., Saavedra, D., & Prasad, M. (2020). X-Ray Baggage Inspection With Computer Vision: A Survey. IEEE Access, 8, 145620-145633.

[4] Akcay, S., & Breckon, T. (2020). Towards Automatic Threat Detection: A Survey of Advances of Deep Learning within X-ray Security Imaging. arXiv preprint arXiv:2001.01293.

[5] Gaus, Y. F. A., Bhowmik, N., & Breckon, T. P. (2019, November). On the Use of Deep Learning for the Detection of Firearms in X-ray Baggage Security Imagery. In 2019 IEEE International Symposium on Technologies for Homeland Security (HST) (pp. 1-7). IEEE.

[6] C.-F. Tsai, "Bag-of-words representation in image annotation: A review,"ISRN Artif. Intell., vol. 2012, 2012, Art. no. 376804.

[7] Chapelle, Olivier, Patrick Haffner, and Vladimir N. Vapnik. "Support vector machines for histogram-based image classification." IEEE transactions on Neural Networks 10.5 (1999): 1055-1064.

[8] Kutulakos, Kiriakos N., and Steven M. Seitz. "A theory of shape by space carving." International journal of computer vision 38.3 (2000): 199-218.

[9] V. Riffo and D. Mery, "Automated detection of threat objects using adapted implicit shape model," IEEE Trans. Syst., Man, Cybern., Syst.,vol. 46, no. 4, pp. 472–482, Apr. 2016.

[10] Jan-Martin O Steitz, Faraz Saeedan, and Stefan Roth. Multi-view X-Ray R-CNN. In German Conference on Pattern Recognition (GCPR), pages 153–168. 2019.

[11] Maoshu Xu, Haigang Zhang, and Jinfeng Yang. Prohibited Item Detection in Airport X-Ray Security Images via Attention Mechanism Based CNN. In Chinese Conference on Pattern Recognition and Computer Vision (PRCV), Lecture Notes in Computer Science, pages 429–439. Springer International Publishing, 2018.

[12] Taimur Hassan, Salman H. Khan, Samet Akcay, Mohammed Bennamoun, and Naoufel Werghi. Deep CMST Framework for the Autonomous Recognition of Heavily Occluded and Cluttered Baggage Items from Multivendor Security Radiographs. CoRR, dec 2019.

[13] Szegedy, Christian, et al. "Rethinking the inception architecture for computer vision." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[14] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014).

[15] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[16] Riffo, Vladimir, Ivan Godoy, and Domingo Mery. "Handgun Detection in Single-Spectrum Multiple X-ray Views Based on 3D Object Recognition." Journal of Nondestructive Evaluation 38.3 (2019): 66.

[17] Mery, D., Riffo, V., Zscherpel, U., Mondragón,G., Lillo, I., Zuccar, I., Lobel, H., Carrasco, M.: GDXray: The database of X-ray images for non-destructive testing. Journal of Nondestructive Evaluation 34(4) (2015) 1–12

[18] Frome, A., Huber, D., Kolluri, R., Bülow, T., & Malik, J. (2004, May). Recognizing objects in range data using regional point descriptors. In European conference on computer vision (pp. 224-237). Springer, Berlin, Heidelberg.

[19] Tombari, F., Salti, S., & Di Stefano, L. (2010, October). Unique shape context for 3D data description. In ACM workshop on 3D object retrieval (pp. 57-62).

[20] Rusu, R. B., Blodow, N., & Beetz, M. (2009, May). Fast point feature histograms (FPFH) for 3D registration. In 2009 IEEE international conference on robotics and automation (pp. 3212-3217). IEEE.

[21] Tombari, F., Salti, S., & Di Stefano, L. (2010, September). Unique signatures of histograms for local surface description. In European conference on computer vision (pp. 356-369). Springer, Berlin, Heidelberg.

[22] Bradski, G., & Kaehler, A. (2008). Learning OpenCV: Computer vision with the OpenCV library. " O'Reilly Media, Inc.".

[23] Danielsson, P. E. (1980). Euclidean distance mapping. Computer Graphics and image processing, 14(3), 227-248.

[24] Zhang, Yin, Rong Jin, and Zhi-Hua Zhou. "Understanding bag-of-words model: a statistical framework." International Journal of Machine Learning and Cybernetics 1.1-4 (2010): 43-52.

[25] Turcsany, Diana, Andre Mouton, and Toby P. Breckon. "Improving feature-based object recognition for X-ray baggage security screening using primed visual words." 2013 IEEE (ICIT). IEEE, 2013.

[26] Lowe, David G. "Distinctive image features from scale-invariant keypoints." International journal of computer vision 60.2 (2004): 91-110.

[27] Leutenegger, Stefan, Margarita Chli, and Roland Y. Siegwart. "BRISK: Binary robust invariant scalable keypoints." 2011 International conference on computer vision. Ieee, 2011.

[28] Bay, Herbert, Tinne Tuytelaars, and Luc Van Gool. "Surf: Speeded up robust features." European conference on computer vision. Springer, Berlin, Heidelberg, 2006.

[29] Rublee, Ethan, et al. "ORB: An efficient alternative to SIFT or SURF." 2011 International conference on computer vision. Ieee, 2011.

[30] Liang, K. J., Sigman, J. B., Spell, G. P., Strellis, D., Chang, W., Liu, F., ... & Carin, L. (2019). Toward automatic threat recognition for airport X-ray baggage screening with deep convolutional object detection. arXiv preprint arXiv:1912.06329.

[31] Agarap, A. F. (2018). Deep learning using rectified linear units (relu). arXiv preprint arXiv:1803.08375.

[32] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." arXiv preprint arXiv:1412.6980 (2014).

[33] Mery, Domingo, et al. "Modern computer vision techniques for x-ray testing in baggage inspection." IEEE Transactions on Systems, Man, and Cybernetics: Systems 47.4 (2016): 682-692.

[34] Leibe, Bastian, Ales Leonardis, and Bernt Schiele. "Combined object categorization and segmentation with an implicit shape model." Workshop on statistical learning in computer vision, ECCV. Vol. 2. No. 5. 2004.

[35] Habbab, I. S. A. M., Mohsen Kavehrad, and C. Sundberg. "Protocols for very high-speed optical fiber local area networks using a passive star topology." Journal of Lightwave Technology 5.12 (1987): 1782-1794.

[36] Ballard, Dana H. "Generalizing the Hough transform to detect arbitrary shapes." Readings in computer vision. Morgan Kaufmann, 1987. 714-725.