



Automated detection of inorganic powders in X-ray images of airport luggage

Danijela Vukadinovic¹ · Miguel Ruiz Osés¹ · David Anderson¹ 

Received: 9 March 2023 / Accepted: 5 April 2023 / Published online: 31 May 2023
© The Author(s) 2023

Abstract

At the checkpoint, the detection of illicit inorganic powders in passenger luggage using conventional X-ray can be challenging. An algorithm is presented for the automated detection of inorganic powder-like substances from complex X-ray images of highly cluttered passenger bags using computer vision. The proposed method utilizes support vector machine (SVM) classifiers built from local binary patterns (LBP) texture features. When tested on a dataset created in-house, the algorithm achieves a detection precision of 97% and a false positive rate of 3%. This is the first study performed on a realistic dataset, including different amounts and shapes of powders and electronic clutter, and where the success of the automated method is compared with inter-observer variability.

Keywords Detection · Explosives · Inorganic powders · X-ray · Aviation security

Introduction

Since air passengers' cabin baggage can be accessed during flight, objects that can be used for both hijacking and sabotaging the aircraft are prohibited (Vukadinovic and Anderson 2022). At the security checkpoint prior to boarding, X-ray images of cabin baggage are visually inspected by human screeners for prohibited items such as guns, knives, and improvised explosive devices (IEDs), as well as other items such as self-defence gas sprays or electric shock devices (Hancock and Hart 2002). The list of prohibited items is long and increasing (European Commission 2021; Transport Security Administration 2021). However, explosives are still considered to be the most dangerous prohibited articles in passenger baggage. Bare explosives are especially challenging to detect because – in the absence of other IEDs components – they can look like a shapeless mass, and are hence difficult to spot even by well-trained observers (Huegli et al. 2020). Bare explosives in cabin baggage pose a

✉ David Anderson
david.anderson@ec.europa.eu

¹ Joint Research Centre, European Commission, Geel, Belgium

threat because they can be combined with other IED components after passing an airport security checkpoint (Hättenschwiler et al. 2018). Furthermore, the target prevalence of real explosives (IEDs and bare explosives) is extremely low which makes the tasks of the screeners more difficult (Vukadinovic and Anderson 2022). Many explosives of concern traditionally fall into the organic range because of their effective atomic number (Z_{eff}). Therefore, they appear orange in pseudo-coloured dual energy X-ray images, and that is how human screeners are taught to recognize them. Inorganic substances, on the other hand, appear green and easily blend in with electronic devices that also appear green or blue, and that are increasingly prevalent in passengers' luggage (Fig. 1).

Contraband materials, including powders, can be automatically detected with other methods with varying success. For example, infrared hyperspectral imaging can be used for detection of traces of explosive manipulation on passengers' fingers as described in Fernández de la Ossa et al. (2014). It took 5 min to successfully analyse fingerprints of volunteers previously manipulating typical substances used to make an IED, such as ammonium nitrate, black powder, single- and double-base smokeless gunpowder and dynamite. Although it requires a long processing time, this is a potentially promising forensic tool for the detection of explosive residues and other related samples. Another method is X-ray diffraction (XRD) (Harding et al. 2012). Such systems achieve a high detection rate and a low false alarm rate for crystalline substances detection, including inorganic powders. These systems are comparatively slow and expensive and therefore tend to be used only as a third tier for hold baggage false alarm clearing (Vukadinovic and Anderson 2022). Faster detection is achieved using automated detection systems (EDS) that analyses X-ray attenuation data for potential explosives before the X-ray image is displayed to the X-ray screener. However, EDS with high detection rates (close to 90%) have false alarm rates in the range of 15–20% (Hättenschwiler et al. 2018).

The automatic detection of illicit inorganic powders in passenger luggage is clearly a challenging problem, both in terms of accuracy and speed. In X-ray imagery, bare explosives cannot be detected using shape, and colour is not a reliable

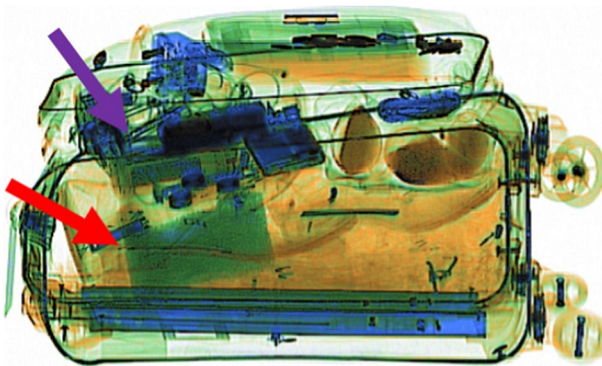


Fig. 1 An inorganic substance (marked with red arrow) appears similar to consumer electronics (marked with purple arrow) on X-ray pseudo-coloured image

feature since it changes due to overlapping with other materials. However, powdery texture could be a promising feature to enable more accurate detection of inorganic powders.

In this paper, a computer-vision based method developed for fully automated detection of inorganic powders from dual-source X-ray images of airport passenger bags packed with realistic electronic clutter is presented. The algorithm achieves both a high level of automation and accuracy. The algorithm consists of several steps; firstly, the region of interest (ROI) where inorganic powder could be located is automatically segmented. Secondly, LBP texture features are extracted from a sliding window that slides over the ROI and the features are fed into a binary classifier that classifies each pixel in the ROI as belonging to a powder or not. Finally, post processing of the classification probabilities results in a segmented area of the target.

This paper is structured as follows: in Sect. 2 related work is presented and discussed, followed by a description of the methodology in Sect. 3. Subsequently, in Sect. 4, the experimental setup is described, including the datasets used, and the evaluation methodology. In Sect. 5, the results are presented, followed by a discussion in Sect. 6 and a conclusion in Sect. 7.

Related work

An overview of the principles of X-ray technology for screening of baggage is given in Vukadinovic and Anderson (2022). In the published literature, threat detection using X-ray and CT images focus on the prohibited items with a characteristic shape; methods dealing with contraband materials are very few. Additionally, no research was published on the detection of powder-like inorganic substances using X-ray images of cluttered airport bags. Therefore, in this section an overview is given of published computer vision methods applied to X-ray images of materials with similar characteristics to contraband materials in airport baggage, although these methods are not necessarily representative of realistic airport checkpoint situations where materials are placed in cluttered bags. Additionally, computer vision methods for the detection of prohibited items such as guns, knives are presented.

Benedykciuk et al. (2020) published work on material classification using one million patches of dual-energy X-ray images, using machine learning with pixel values as features, and classifying between light organic materials, heavy organic materials, mixed materials, and heavy metals. A year later, the same authors published their work on the same topic, and using the same database, but with a CNN approach instead of conventional machine learning (Benedykciuk et al. 2021). The average accuracy was equal to 95%, and not higher than in Benedykciuk et al. (2020). Both algorithms (Benedykciuk et al. 2021; 2020) were trained and tested on patches of different materials and not on cluttered bags. Morris et al. (2018) tested several CNNs on the Passenger Baggage Object Database (PBOD) (Gittinger et al. 2018) consisting of dual-energy X-ray images of bags containing explosives. The best results were obtained using a hybrid model consisting of VGG19 (Simonyan and Zisserman 2015) convolutional layers and Xception (Chollet 2017) and InceptionV3 (Szegedy et al. 2016) top layers, achieving an area under curve (AUC) of 0.95, a

false positive rate (FPR) of 0.06 and a false negative rate (FNR) of 0.24. The types of explosives used in the study were not described for security reasons. Chouai et al. (2020) used a modified YOLO V3 (Redmon and Farhadi 2018) to detect all objects in dual-energy X-ray images of baggage, an adversarial autoencoder (Makhzani et al. 2016) to extract features, and a SVM classifier to detect explosives and firearms among selected objects. The dataset consisted of 10,000 objects. The types of explosives used in this study are not described, nor the creation of the dataset, but based on example images shown in Chouai et al. (2020), low-complexity to high-complexity images were used. The accuracy of the presented method was 96.5% and the F1 score was 0.94, while the time needed to perform object classification varied between 350 and 425 ms, based on the image size. Kayalvizhi et al. (2022) developed a CNN-based algorithm for the automated detection of threat materials in dual-energy X-ray images. The dataset consisted of grey and pseudo-coloured images of threat materials (explosive simulants) and non-threat organic materials having a similar Z_{eff} . In contrast to the dataset used in this paper, imaged bags in Kayalvizhi et al. (2022) were sparsely packed and with no overlapping objects. The initial dataset was expanded using a deep convolutional generative adversarial network (DCGAN) (Radford et al. 2016) and classical image augmentation (Bloice et al. 2019) to a final dataset comprising 20,000 grey images and 12,000 pseudo-coloured ones. On grey and pseudo-coloured images, the method achieved an accuracy of 97% and 98%, respectively, and a speed of 12 s and 6 s, respectively.

Regarding computer vision methods for object detection in X-ray security images, an extensive review of the application of machine learning in X-ray security screening was published by (Vukadinovic and Anderson 2022) where it was shown that most of the methods for prohibited items detection rely on the bag of visual words (BoVW) concept. BoVW was used for handgun detection (Baştan et al. 2011), firearm classification (Turcsany et al. 2013), and for 4-class classification of guns, shuriken, razor blades and clips (Mery et al. 2016). The typical approach for BoVW was to use different feature detectors to encode features of images in a vector quantized representation. For example, Baştan (2015) used several feature detectors (Harris–Laplace, Harris-affine, Hessian–Laplace, Hessian-affine), combined with SIFT and SPIN feature descriptors, and used an SVM classifier to discriminate several classes (guns, bottles and laptops). There are also several methods that don't employ BoVW. Franzel et al. (2012) used histogram oriented gradient (HOG) features and machine learning applied on all four X-ray views (Dalal and Triggs 2005), Schmidt-Hackenberg et al. (2012) used features inspired by the human visual cortex for the detection of guns, and Roomi (2012) used shape context descriptors (Belongie et al. 2002) and Zernike moments for features (Khotanzad and Hong 1990) fed into a fuzzy k-NN classifier (32). For deep learning methods, it is normally necessary to use transfer learning due to a lack of data in this field. Akçay et al. (2016) used AlexNet (Krizhevsky et al. 2012) as a base model which they further optimized for handgun image classification. Training and testing datasets consisted of image patches cropped from 6,997 X-ray images created by the authors. They compared their transfer learning method directly with the approach of Turcsany et al. (2013), and the transfer learning method showed superior results. The same authors published more elaborate studies on threat detection (Akçay and Breckon 2017; Akçay

et al. 2018), with more data, comparing traditional sliding-window-based CNN detection with region-based object detection techniques (Ren et al. 2017), region-based fully convolutional networks (R-FCN) (Dai et al. 2016) and the YOLOv2 model (Redmon & Farhadi). Hassan et al. (2019) proposed a contraband-object-detection algorithm with complex, non-deep learning based ROIs extraction based on the variations of the objects directions and applied on heavily occluded and cluttered baggage using GDXray (Mery et al. 2015) and SIXray (Miao et al. 2019) datasets. ROIs were then fed into a CNN model using a pre-trained ResNet50 model (He et al. 2016), achieving a very high accuracy of 99.5% and a FPR of 4.3%.

Methodology

The method used to solve the difficult task of automated detection of inorganic powder in cluttered bags is based on pixel-based machine learning approach, with colour-texture features extracted using a sliding window approach (Benedykciuk et al. 2020; Franzel et al. 2012; Schmidt-Hackenberg et al. 2012; Akçay and Breckon 2017). Overall, the method can be separated into three parts:

- a pre-processing step with ROI candidates extraction
- texture feature extraction using a sliding-window approach
- classification and post-processing.

Pre-processing: ROI candidates extraction

In order to extract the initial region of interest (ROI) where the threat might be located, the RGB images are pre-processed. First, the bag area of the image is extracted from the original image by thresholding the local entropy image. The local entropy image is calculated using a grey-scale image of the original RGB X-ray image. The local entropy is related to the complexity contained in a given neighbourhood, typically defined by a structuring element. The entropy filter can detect subtle variations in the local grey level distribution and it is calculated using the following formula within the structuring element:

$$H = - \sum_{i=0}^{255} p_i \log_2 p_i, \quad (1)$$

where p_i is the probability (obtained from the normalized local histograms of the image) associated with the grey level, i . In this work, the structuring element, chosen by educated guess, was a disk-shaped structuring element of radius = 10 pixels. The *scikit-image* Python library was used to perform morphology operations. The final ROI depicting the bag is the result of applying an Otsu threshold (Otsu 1979) on the local entropy image (Fig. 2). Inorganic powders appear in the X-ray images as green or green-bluish areas. Even if there is an organic object overlapping it, there will be a trace of green-blue colour visible in the orange object area. In order to exclude all the pure orange and white areas from the segmented bag, and keep the remaining

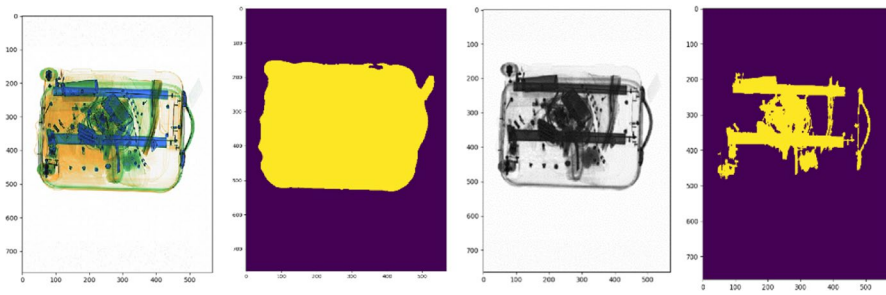


Fig. 2 **a** RGB X-ray image of luggage containing an inorganic powder; **b** a bag mask resulting from Otsu thresholding of the local entropy image. **c** red component of the bag area of the image on which threshold is applied to extract the ROI within the bag mask, **d** connected components smaller than 500 pixels removed from (c) to create the initial ROI mask

part of the image for further analysis, colour-based thresholding was performed. First, the red channel of the RGB image is blurred using a Gaussian kernel of 5×5 pixels, and is thresholded with the fixed threshold, $th = 100$. Filtering out connected components (Rosenfeld and Pfaltz 1966) smaller than 500 pixels results in the area of interest where inorganic powders could be located. The resulting area is a heavily overestimated region where the powder of interest, if in the bag, is located with high certainty, while many false positive pixels are included. This area is referred to as the *Initial ROI*, and the process of deriving it from an image is illustrated in Fig. 2.

During training, positive and negative samples were extracted from the *Initial ROI* area. Positive samples were extracted from an area created by dilating the ground truth (GT) ROIs with a disk-shaped element with a radius of 10 pixels. This area is referred to as *Positive ROI*. Negative samples were extracted from an area defined with the following equation:

$$\text{NegativeROI} = \text{InitialROI} - \text{PositiveROI} \quad (2)$$

where *Negative ROI*, *Positive ROI* and *Initial ROI* are sets consisting of pixels, and the—operator is a difference operator applied to two sets. The thresholds and the minimum connected component sizes used in this process were chosen after a few trial-and-error educated guesses on the tuning set. In the training phase, pixels belonging to the *Positive ROI* area represent positive samples, and pixels belonging to the *Negative ROI* represent negative samples. Every 10th pixel selected from the *Initial ROI* represented test samples. For each sample (positive, negative or test), the same set of features is extracted.

LBP features extraction

In case of inorganic powders, it is very difficult for a human screener to see powders in a cluttered X-ray image because of their arbitrary form. The issue of green–blue clutter is a growing problem, due to the increasing amounts of electronic items carried in passengers' bags, such as cables, phones, tablets,

headphones and laptops (see Fig. 1). Perhaps the only distinguishable feature, visible to human screeners, besides colour, is texture. One of the most widely used texture descriptors are local binary patterns (LBPs), first presented by Ojala et al., in 1996 (Ojala et al. 1996). Lately, several studies claim an improved performance of CNNs if the LBP transformation of input images is performed before the convolution (de Souza et al. 2017; Sharifi et al. 2020; Xi et al. 2016). In a recent survey on texture feature extraction methods (Ghalati et al. 2021), among rotation and scale invariant methods, one of the most successful texture descriptors were LBPs. In aviation security applications, LBP texture descriptors have not been used often. One study where they showed good performance was Mery et al. (2016), where they were successfully used within a BoVW framework.

The original version of LBP (Ojala et al. 1996) encodes the local structure around each pixel with LBP patterns calculated in its 3×3 neighbourhood. Each image pixel is compared with its eight neighbours in a 3×3 neighbourhood by subtracting the centre pixel grey value from the neighbourhood pixels grey values. The resulting negative values are encoded with a 0, and the positive ones with 1 (Huang et al. 2011) forming a binary LBP code (see Fig. 3).

Using variable neighbourhood sizes was necessary to capture multiscale textures. For this reason, and to be able to apply modifications that would allow extension to rotation invariant version, circular neighbourhoods were used for LBP code calculation. LBP code is computed by sampling evenly distributed p pixels on a circle of radius r from a central pixel, c , and comparing their grey values $\{g_i\}_{i=0}^{p-1}$ with the central pixel grey value, g_c .

$$LBP_{r,p}(c) = \sum_{i=0}^{p-1} s(g_i - g_c) 2^i, s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (3)$$

If the coordinates of g_c are $(0, 0)$, then the coordinates of g_p are given by $(-r \sin(\frac{2\pi p}{P}), r \cos(\frac{2\pi p}{P}))$ – see Fig. 4. In practice, the neighbouring pixels grey values that do not fall exactly in the centre of pixels, are interpolated. The LBP patterns resulting from Eq. (3) is a binary code that is, due to the sign function $s(x)$, invariant against any monotonic transformation of image brightness.

If LBP patterns are calculated for each part of an $N \times M$ image, then that image texture could be characterized by the distribution of LBP patterns presented in the form of a histogram vector, H (see Eq. (4)).

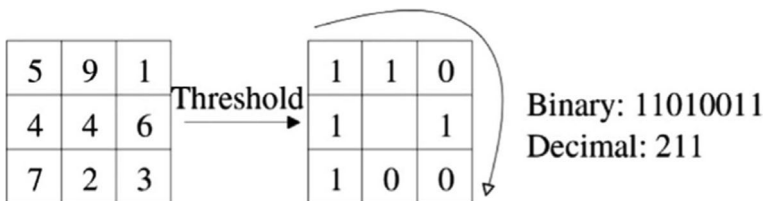


Fig. 3 Example of the basic LBP pattern (Huang et al. 2011)

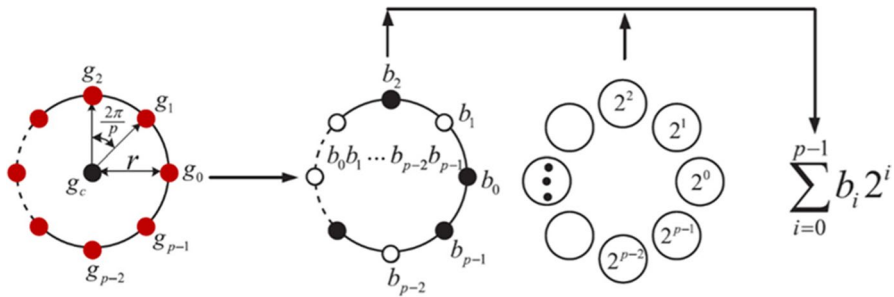


Fig. 4 A typical (r, p) neighbourhood type used to derive a LBP-like operator: central pixel g_c and its p circularly and evenly spaced neighbours g_0, g_1, \dots, g_{p-1} , on a circle of radius r . (Liu et al. 2017)

$$H(k) = \sum_{i=1}^N \sum_{j=1}^M \delta(LBP_{r,p}(i,j) - k) \tag{4}$$

where $0 \leq k < d = 2p$ is the number of LBP patterns (Liu et al. 2017).

For the problem addressed in this study, a rotationally invariant texture detector is needed. The basic LBP approach is easy to implement, invariant to monotonic illumination changes, and has low computational complexity. However, there are number of shortcomings, such as sensitivity to image rotation and sensitivity to noise (since the slightest fluctuation above or below the value the central pixel changes the LBP code). Furthermore, producing large histograms even for small neighbourhoods leads to decreased distinctiveness and large storage requirements (Liu et al. 2017). A number of modifications were introduced to the basic LBP implementation to overcome these shortcomings.

The rotation of the image, inevitably move the values g_p along the circle around g_c . Given that value g_0 is always assigned to the $(0, r)$ coordinate, right from the central pixel (see Fig. 4), rotation of the image changes the $LBP_{r,p}$ value. To remove the effect of rotation, a rotationally invariant version of LBP was obtained by grouping together LBPs that are rotated versions of the same patterns and mapping them to the minimum value LBP code of that pattern (see Eq. (5)).

$$LBP_{r,p}^i = \min\{ROR(LBP_{r,p}, i) \mid i = 0, 1, \dots, p - 1\} \tag{5}$$

where $ROR(x, i)$ performs a circular bit-wise right-shift on the p -bit number x , i times to the right ($i < p$). In terms of image pixels, this operation corresponds to rotating the neighbour set clockwise so many times that, the maximum number of the most significant bits, starting from g_{p-1} is 0. Besides ensuring rotational invariance, this modification significantly reduces feature dimensionality. However, the number of LBP codes still increases rapidly with p .

Some LBP patterns occur more frequently than others. It was observed in Ojala et al. (2002) that certain LBPs are fundamental properties of texture, providing the vast majority of all patterns present. These are called “uniform” patterns since they have one thing in common—uniform circular structure that contains very few spatial

transitions. Uniform patterns are illustrated in the upper part of Fig. 5. They function as templates for microstructures such as a bright spot and flat area or a dark spot (upper row), and edges of varying positive and negative curvature (bottom row).

“Uniform” patterns are defined using a uniformity measure, U , which corresponds to the number of spatial transitions (bitwise 0/1 changes) in the given LBP pattern (Ojala et al. 2002). U is formally defined as:

$$U(LBP_{r,p}) = \left| s(g_{p-1} - g_c) - s(g_0 - g_c) \right| + \sum_{i=1}^{p-1} \left| s(g_i - g_c) - s(g_{i-1} - g_c) \right| \quad (6)$$

where s , and g_c are defined as in Eq. (3), g_0 is always assigned to the $(0, r)$ coordinate right from the g_c (see Fig. 4 and Fig. 5). Furthermore, a grey-scale and rotation invariant texture descriptor consisting of “uniform” LBP patterns is defined as:

$$LBP_{r,p}^{riu} = \begin{cases} \sum_{i=0}^{p-1} s(g_i - g_c), & \text{if } U(LBP_{r,p}) \leq 2 \\ p + 1, & \text{otherwise} \end{cases} \quad (7)$$

The final texture features that are used in this study are calculated as histograms of the $LBP_{r,p}^{riu}$ texture descriptors. The reason why the histogram of “uniform” patterns provides better discrimination compared to the histogram of all individual patterns is due to differences in their statistical properties. The relative proportion of “non-uniform” patterns of all patterns accumulated into a histogram is so small that their probabilities cannot be estimated reliably. The inclusion of their noisy estimates in the dissimilarity analysis of sample and model histograms would deteriorate performance (Ojala et al. 2002).

Pixel-based classification is used, i.e. any pixel that belongs to *Initial ROI* described in Sect. 3.1, represents a sample whose features are extracted in order to classify them. In order to extract texture features that describe those pixels, a sliding

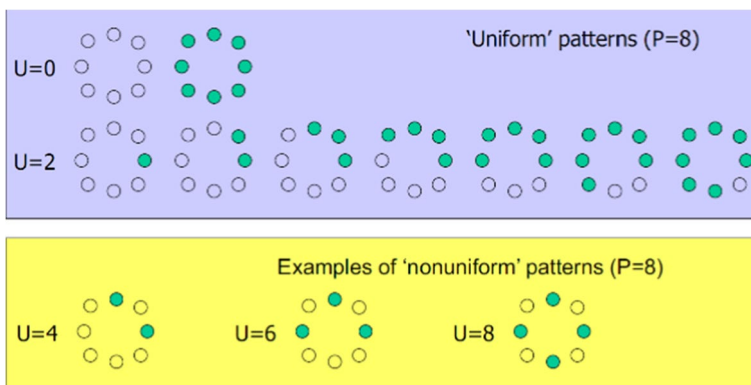


Fig. 5 “Uniform” vs “non-uniform” LBP patterns, with P representing number of sampled points. (Pietikäinen and Heikkilä, 2011)

window centered at the pixel is utilized. In the training phase, pixels are randomly selected within *Positive ROI* area and *Negative ROI* area, representing positive and negative samples respectively. An example of randomly selected 100 positive samples from *Positive ROI* (green) and 200 negative samples from *Negative ROI* (red) is shown in Fig. 6.

Each sample is represented with several LBPs with different values of (r, p) pairs in $LBP_{r,p}^{riu}$ as defined in Eq. 7. Each LBP pattern is calculated for the sliding window image patch, and consequently represented as a histogram vector of LBP patterns. Each LBP histogram vector has $p+2$ bins, making a total number of features of an $LBP_{r,p}^{riu}$ set extracted from one image patch equal to:

$$Nm = \sum_{i=1}^{i=\text{numofLBPs}} (p_i + 2) \quad (8)$$

In the presented method, several image modes are used to extract features: all channels of RGB and Lab images and the greyscale image, for a total number of features of:

$$Nt = Nm * \sum_{i=1}^{i=\text{numofLBPs}} (p_i + 2) \quad (9)$$

Classification and post-processing

After normalizing the data to zero mean and unit variance, a classifier was trained on the training set. Several classification methods were investigated, namely, support vector machines (Vapnik 1999), AdaBoost (Freund and Schapire 1997; Friedman et al. 2000; Schapire 2013), and decision trees (Rokach and Maimon 2005). All three classification methods were compared as described in the experimental set up (see Sect. 4.2). In the testing phase, features were extracted for every 10th pixel within *Initial ROI* and fed into a trained classifier for each of these pixels to be classified as belonging to the powder area or not. Sparse sampling was performed because calculating complete set of features for each pixel was time consuming, and

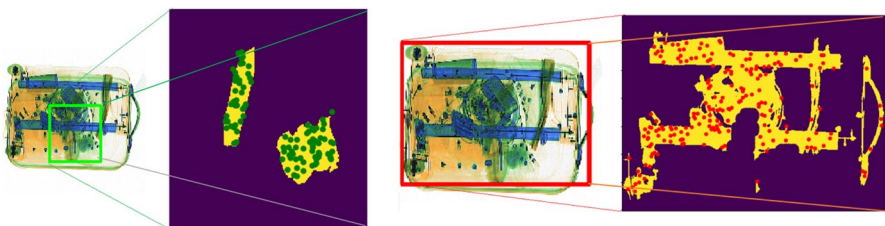


Fig. 6 Example extraction of (a) 100 positive training samples; (b) 200 negative training samples

there was no benefit given that the pixels representing inorganic powders were anyway grouped in spatially compact clusters reflecting either positive or negative classification result.

The result of the classification that is assigned to the each pixel is the distance to the separation plane in feature space. Positive values mean that the given pixel belongs to the powder area and negative values that it does not. The higher the absolute value of the pixel, the more confidence in the classification result. The resulting images can be considered classification confidence images. Some examples of confidence images are shown in Fig. 7. Morphological closing was performed on positive values, and connected components smaller than 300 pixels were removed (Fig. 7c). The performance of the powder detection algorithm was analysed for each image separately and per bag.

Data and experimental set up

An in-house dataset was created to train and evaluate the proposed algorithm. The dataset comprises X-ray images of realistically packed bags with electronic clutter and different shapes and amounts of powders. Two human observers were involved in the labelling process to assess inter-observer variability.

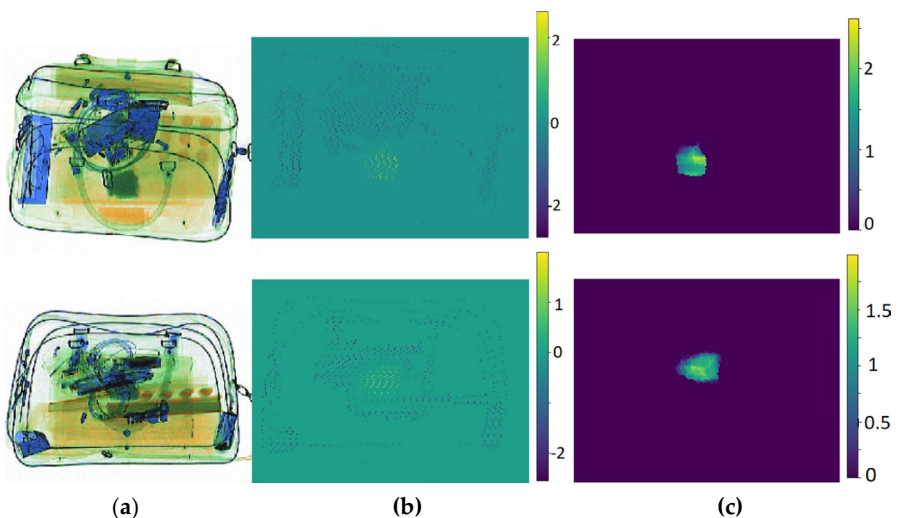


Fig. 7 Example of morphological closing of confidence images, which was the first step in post processing: (a) two RGB X-ray views of a bag; (b) classification values for each testing sample (i.e. confidence images); (c) morphological closing of positive values

Data

The bag set consisted of cabin bags and hold bags supplied by Iconal Ltd, UK. Five different sets of electronic clutter were assembled and placed inside different bags during the imaging process (Fig. 8).

Inorganic powder threats are simulated by placing different amounts of coarse table salt, NaCl, in plastic bags, in masses of 50 g, 100 g and 200 g. Each bag with a simulated threat was always imaged with one of the sets of clutter.

Each threat was shaped in three different forms: spherical, cylindrical and thinly spread:

- 200 g amount was imaged thinly spread in each cabin and hold bag, amounting to 40 bags in total.
- 200 g amount was imaged in a cylindrical shape with every second cabin and hold bag, amounting to 20 bags in total.
- 200 g amount was imaged in a spherical shape with every second cabin and hold bag amounting to 20 bags in total.
- 100 g amount was imaged only in a spherical shape with every cabin and hold bag amounting to 40 bags in total.
- 2×50 g amount was imaged one package in cylindrical and the other one in thinly spread shape with every cabin and hold bag (15 bags) amounting to 40 bags in total.

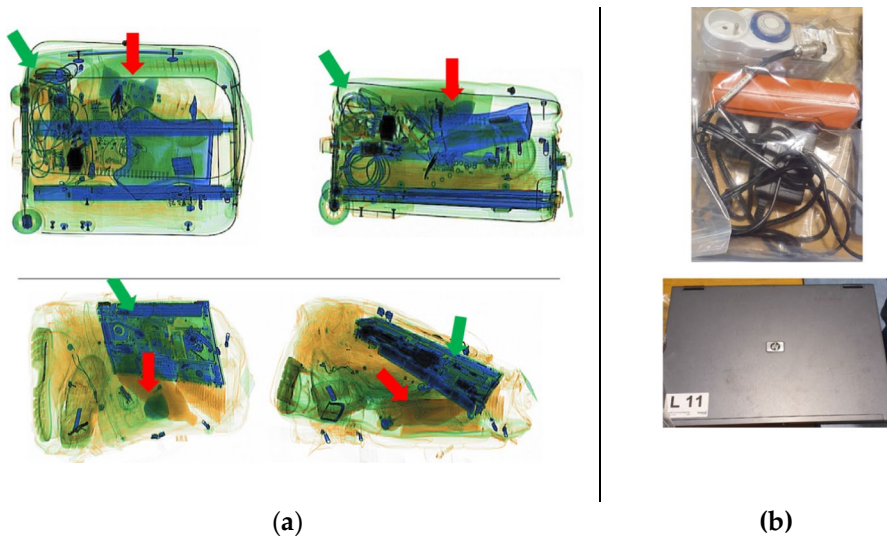


Fig. 8 Two examples of X-ray images from the dataset created and used in this study: **a** two views of X-ray images with the red arrow pointing to the inorganic powder and the green arrow to the electronic clutter; **b** corresponding RGB images of the electronic clutter (b)

The final number of bag-images was 225, each X-ray scan containing 2 views, out of which 65 scans (130 images) containing no threat, and 160 scans with threat (320 images).

In-house labelling software was developed for the data annotation process. Observers could see both views, the amount and shape of the threat present, and the total number of threats was shown on the screen. Then, ROIs are drawn in one chosen view, and afterwards the other view is shown with the vertical lines showing the x-range where the object is positioned. Observers had unlimited time to spot the powder in the images, and could consult RGB images (photographs) of open bags with the threat in it made by smartphone immediately prior to scanning.

The conditions for human observers were deliberately made easier than in airport checkpoints in order to get a more accurate ground truth. Nevertheless, it was sometimes difficult for observers to spot the powders, for example, for small amounts of 50 g, especially if the powder was thinly spread and fully overlapped with the electronic clutter. The amount of time needed for observers to spot the powders was long, and even longer when small amounts of powder were observed; *Observer 1*, on average, spent 63 s per bag, and while labelling only bags with 50 g powder, *Observer 1* spent 105 s per bag. *Observer 2* spent on average 86 s per bag, and while labelling only bags with 50 g powder, *Observer 2* spent on average 129 s per bag.

Experimental set-up

The algorithm was trained with the ground truth labelled by *Observer 1* only. *Observer 2* labels were used to assess the inter-observer variability of detection. From the set of images where the simulated threat was present, those where *Observer 1* labelled each image were selected. This set – referred to as Set A – contained 264 images.

The test set was created in the following way. From Set A, 25 bags (50 images) were randomly selected. Added to this set were 25 randomly selected bags (50 images), from the set of images with no threat present. In total, the test set contained 50 bags (100 images).

A small tuning set was created in the following way. From the remaining images in set A, 10 bags (20 images) were randomly selected. Added to this set were 10 bags (20 images) selected randomly from the remainder of the image set with no threat present. The tuning set contained 20 bags (40 images) in total. The final test set for powder detection was created by merging the test set and tuning set, containing in total 140 images of 70 bags (35 with powder and 35 without powder).

The remaining images of set A, 194 images, were used for training. Both positive and negative samples were extracted from the *Initial ROI* of each image belonging to this set as described in Sect. 3. From each train image Initial ROI, 100 positive and 200 negative samples were randomly selected.

A tuning set was used in order to choose the best classifier and its parameters, and the optimal threshold applied on the pixel-based classification result. Four classifiers were used initially: SVM (Vapnik 1999) with linear kernel, SVM with RBF kernel, AdaBoost (Freund and Schapire 1997; Friedman et al. 2000; Schapire 2013), and

decision trees (Rokach and Maimon 2005). The parameters of each classifier were determined empirically on a few examples from the tuning set using the *scikit-learn* Python library. The parameters of the linear SVM classifiers were $C = 100$, L2 norm for penalty, and the number of maximum iterations were set to 4,000. For the SVM classifier with RBF kernel, $C = 100$, and $1/\text{gamma}$ that is equal to number of training samples was used. AdaBoost classifier was used with decision tree classifier as a base classifier, a number of estimators equal to 300, and the learning rate equal to 1. The maximum depth of the decision trees classifier was equal to 10. Each of the classifiers was trained on the training set, and tested on the tuning set, using the ground truth labelled by *Observer 1*. The test set and final test set were used to test the performance of the chosen classifier against the ground truth labelled by the *Observer 1*.

The features used were rotation invariant and the uniform LBP feature, $LBP_{r,p}^{riu}$ with (r, p) pairs values equal to:

$$(r, p) = [(1, 24), (2, 24), (5, 24), (8, 24)] \quad (10)$$

where (r, p) pairs are defined in Fig. 4, and the sliding window size used for each pixel classification was set to 80×80 pixels.

Evaluation measures

For pixel-based classification, the classes are highly imbalanced: there are many more negative examples than positive ones in each image. In the testing phase, each of the pixels belonging to the *Initial ROI* was classified as belonging to the powder or not, and the number of non-powder pixels, according to the GT, was, on average, more than 10 times higher than the number of powder GT pixels. The result of such an imbalance is typically that the error rate for the majority class is much smaller than for the minority class (Monard and Batista 2002), hence, the minority class has a much higher misclassification cost. F-measure is a harmonic mean between recall and precision, and tends to be closer to the smaller of the two. Therefore, a high F-measure value ensures that both recall and precision are reasonably high (Sun et al. 2009). F-measure is defined as:

$$F = \frac{2}{\frac{1}{R} + \frac{1}{P}} \quad (11)$$

where R represents recall, and P precision.

The performance of the presented algorithm is evaluated per powder and per bag. When using standard evaluation measures for detecting a powder (not an alarmed bag) defining true negatives (TN) is not possible. Hence, in the evaluation of the algorithm's detection performance per powder, TNs are not included, while for detection per bag they are.

With detection per powder, a true positive (TP) detected object is one whose bounding box overlaps with the ground truth (GT) object bounding box marked by *Observer 1*, and a false positive (FP) is one that does not. If there is a ROI marked

by *Observer 1*, and no automated detection in this location, this is counted as a false negative (FN).

The evaluation per bag is performed following the algorithm described in Fig. 9. Namely, if there is an object placed in a bag, detected by the observer from images, $GT_bag = 1$, otherwise, $GT_bag = 0$. If $GT_bag = 1$, and if there is at least one correctly detected object of interest in either of the two views, TP_bag is equal one. If $GT_bag = 1$, and TP and FP in both views are equal to 0, then the false negative of this bag is set to 1. If the *Observer 1* did not see the threat, GT_bag is set to 0, and if false positives are detected by the algorithm, FP_bag is set to 1, otherwise, TN_bag is set to 1. Inter-observer variability is assessed by performing the same detection analyses, per powder and per bag, by comparing *Observer 2* labels with the ground truth (*Observer 1* labels).

The following evaluation metrics were used:

$$Precision = TP / (TP + FP) \quad (12)$$

$$Recall = TPR = TP / (TP + FN) \quad (13)$$

$$FNR = \frac{FN}{P} = \frac{FN}{FN + TP} \quad (14)$$

Because TN is possible to calculate for evaluation per bag, the ROC curve (TPR vs FPR) using FPR is also presented. FPR is calculated as:

$$FPR = \frac{FP}{N} = \frac{FP}{FP + TN} \quad (15)$$

Results

The F-measure calculated on the tuning set for all four classifiers is presented in Fig. 10.

```

IF GT object exists in either view1 or view2 of a bag
THEN:
     $GT\_bag = 1$ 
    IF (TP_view1 >0 OR TP_view2>0)
    THEN:  $TP\_bag = 1$ 
    ELSE:  $FN\_bag = 1$ 
ELSE:
     $GT\_bag = 0$ 
    IF FP_view1>0 OR FP_view2 >0:
    THEN:  $FP\_bag = 1$ 
    ELSE:  $TN\_bag = 1$ 

```

Fig. 9 Pseudocode describing the algorithm for determining if a bag is a hit ($TP_bag = 1$), correctly cleared ($TN = 1$), false alarm ($FP_bag = 1$) or a miss ($FN_bag = 1$). The algorithm input is the detection success per powder in two views (i.e., TP_view1 , TP_view2 , FP_view1 , FP_view2)

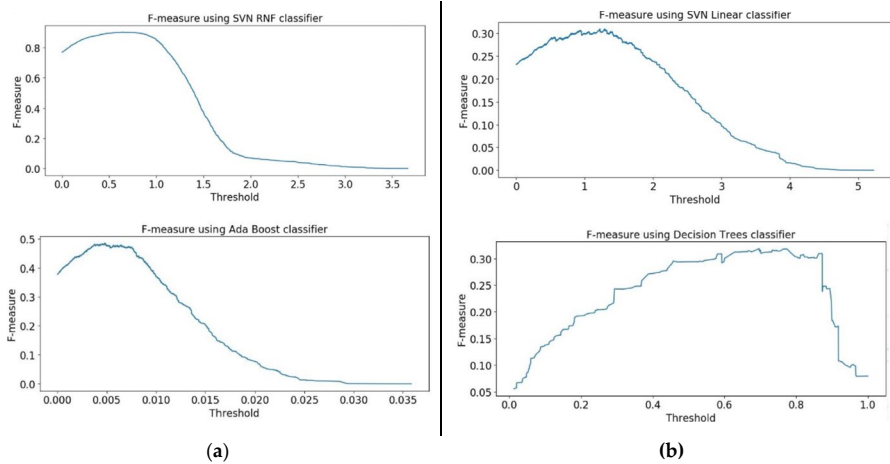


Fig. 10 F-measure vs. threshold applied on the pixels of the confidence image. F-measure is plotted for four different classifiers: SVN with RBF kernel (upper left), SVM with linear kernel (upper right), Ada Boost (bottom left), and Decision Trees (bottom right)

The following performance curves for all four classifiers used on the tuning set are shown: Precision-Recall curves in Fig. 11, TPR curves in Fig. 12, and FNR curves in Fig. 13. Each of the curves is created by varying classification thresholds, and the results of the detection per bag and per powder are presented in each figure.

ROC curves for detection per bag, for all four classifiers, tested on the tuning set are presented in Fig. 14.

As indicated from the F-measure for the pixel-based classification (see Fig. 10) where the best performing classifier was SVM-RBF (F-measure = 0.90), the same classifier had the best performance for powder detection. After selecting the optimal threshold from P-R and ROC curves for the SVM-RBF classifier

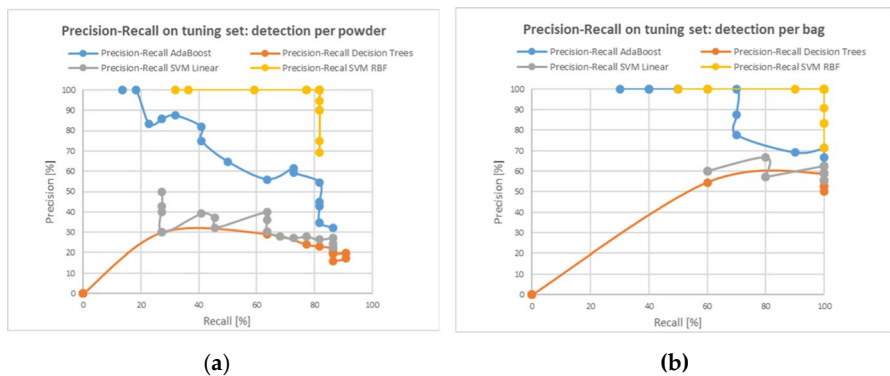


Fig. 11 Precision-recall curves for all four classifier applied on the tuning set and calculated for detection: **a** per powder and **b** per bag

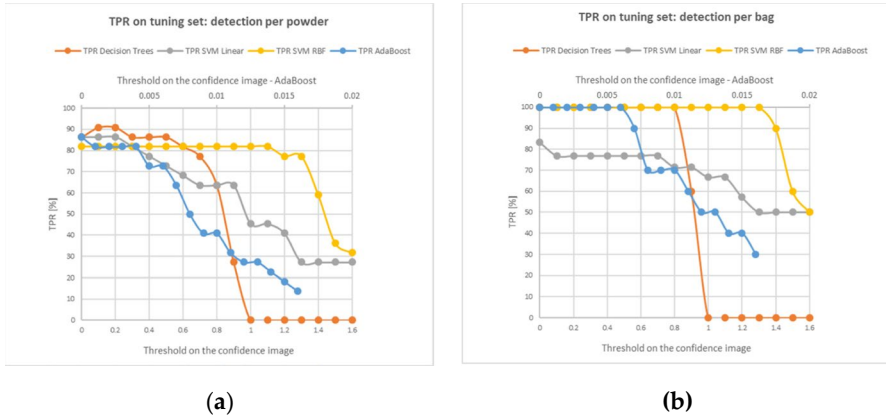


Fig. 12 True Positive Rate (TPR) curves for all classifiers applied on the tuning set and calculated for detection: **a** per powder, and **b** per bag

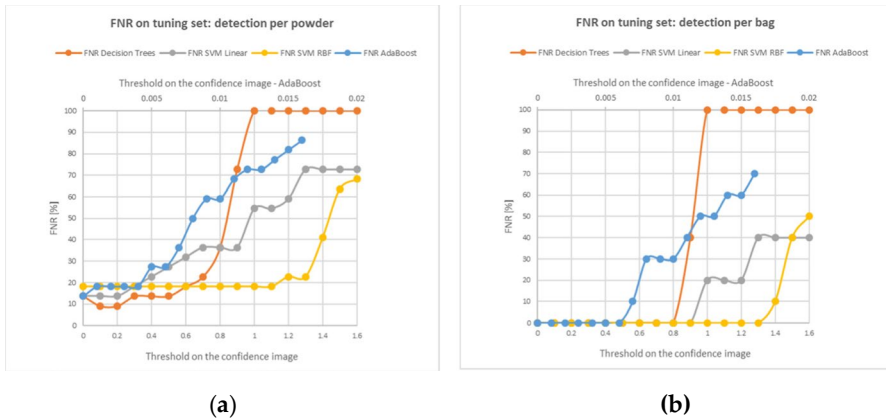


Fig. 13 False negative rate (FNR) curves for all classifiers applied on the tuning set and calculated for detection: **a** per powder, and **b** per bag

($th_{opt} = 1.1$), the method was tested on the test set and the final test set (test set + tuning set), and the results are presented in Table 1 and Table 2 respectively. These results are also compared with the inter-observer variability, i.e. applying the same metrics for evaluating the performance of *Observer 2* versus *Observer 1* annotations. The inter-observer variability is calculated only on the samples with powder present since only on these images the observers annotated powder ROIs. TNs in the calculation of FPR between two observers are the bags that *Observer 1* did not annotate, irrespective of whether the powder was present in the bag or not.

Additionally, the precision-recall curves for positive and negative class of the final test set are shown in Fig. 15, where mAP over both classes is equal to 88%.

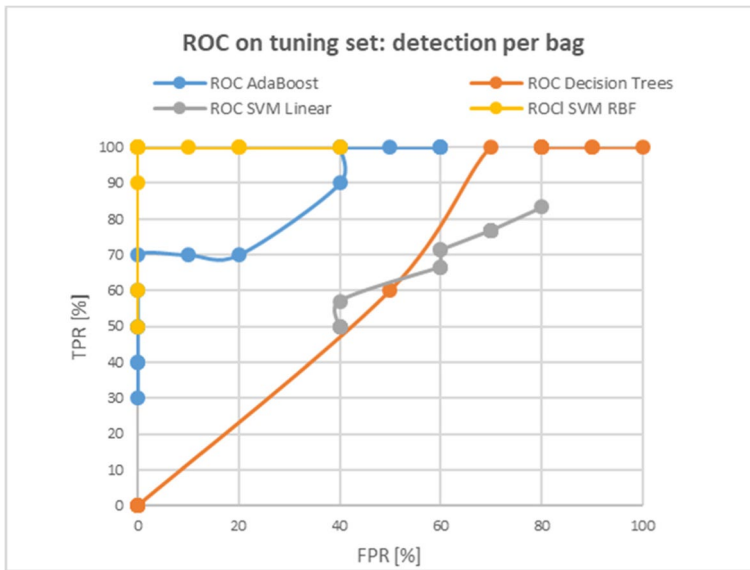


Fig. 14 ROC curves for all four classifiers applied on the tuning set and calculated for detection per bag

Table 1 Test set: evaluation of the results. Automated detection (AM) vs ground truth (GT) as annotated by *Observer 1*, and *Observer 2* detection vs *Observer 1*. *Observer 1* is marked as O1, *Observer 2* as O2, number of true positives as TPs, number of false positives as FPs, precision as P, recall as R, false positive rate as FPR, and accuracy as ACC

	Detection type	No. of powders (O1)	TPs	FPs	FNs	P	R			
Per powder	AM vs. O1	44	33	-	5	11	87%	75%		
	O2 vs. O1		39	-	14	5	74%	89%		
		No. of bags with powder (O1)	TPs	TNs	FPs	FNs	P	R	FPR	ACC
Per bag	AM vs. O1	24	21	25	1	3	95%	87%	4%	92%
	O2 vs O1		22	25	1	2	96%	92%	4%	94%

Some examples of correct and incorrect positive classifications are shown in Fig. 16.

The algorithm presented in this study was implemented in the Python programming language using standard Python libraries. Machine learning was implemented using the *scikit-learn* Python library. The algorithm was developed using a laptop of modest computational power: 16 GB of RAM and i7-855U CPU with two cores. On this laptop, testing one image took around 8 min. When the same algorithm was executed on the JRC Big Data Platform (Soille et al. 2018), utilizing parallel processing

Table 2 Final test set (test set+tuning set): evaluation of the results. Automated detection (AM) vs ground truth (GT) as annotated by *Observer 1*, and *Observer 2* detection vs *Observer 1*. *Observer 1* is marked as O1, *Observer 2* as O2, number of true positives as TPs, number of false positives as FPs, number of false negatives as FNs, precision as P , recall as R , false positive rate as FPR, and accuracy as ACC

	Detection type	No. of powders (O1)	TPs	FPs	FNs	P	R			
Per powder	AM vs. O1	66	51	-	5	15	91%	77%		
	O2 vs. O1		60	-	17	6	78%	91%		
		No. of bags with powder (O1)	TPs	TNs	FPs	FNs	P	R	FPR	ACC
Per bag	AM vs. O1	34	31	35	1	3	97%	91%	3%	94%
	O2 vs O1		32	35	1	2	97%	94%	3%	96%

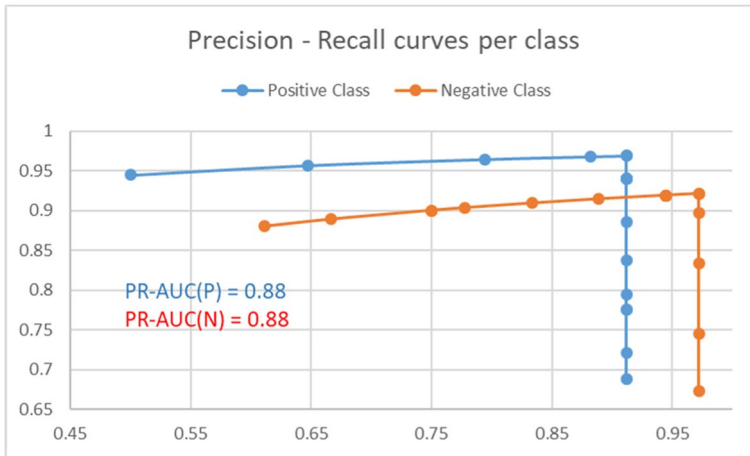


Fig. 15 PR curves for SVM-RBF based classification per positive (blue) and negative (orange) class. The area under curve (AUC) for both classes is 88%

with 100 cores and in a Linux-based environment, the processing time was reduced to around 15 s per bag on average.

Discussion

Classifiers comparison

From F-measure calculated for pixel-based classification, and for Precision-Recall curve and TPR and FNR curves calculated on the tuning set, it is clear that the

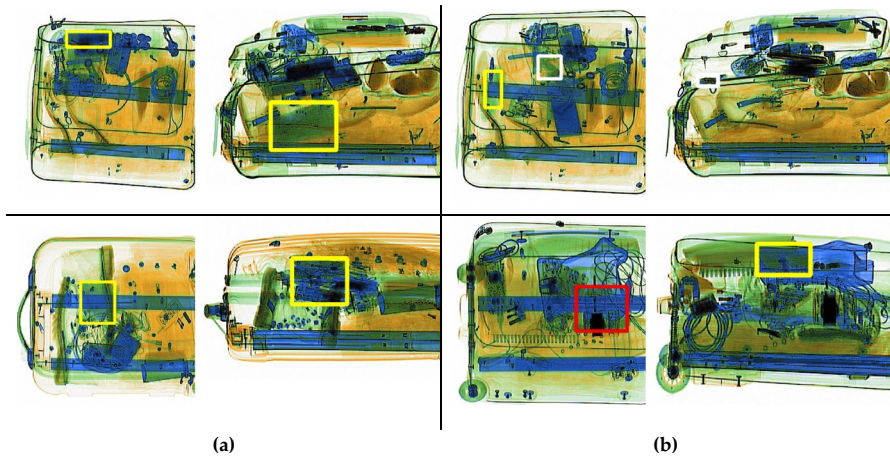


Fig. 16 Examples of correct and incorrect automated detection presented on both views of X-ray images of bags. TPs are marked with a yellow rectangle, FNs with white, and FP with red

Support Vector Machine with RBF kernel performs best. That is mostly reflected in the small number of false positives that SVM-RBF classifier generates. To better illustrate classifier performances, typical examples of a bag containing a powder (Fig. 17) and a bag with no powder and a laptop inside (Fig. 18) are shown, together with confidence images resulting from implementation of different classifiers.

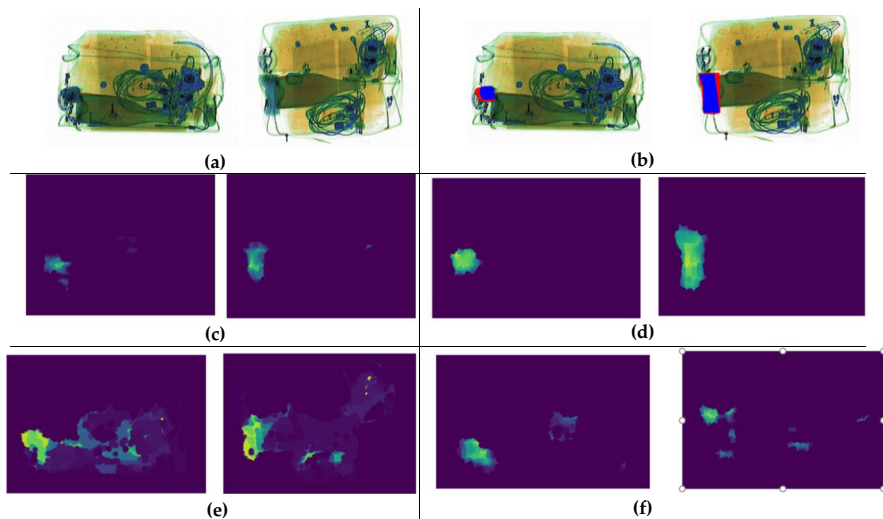


Fig. 17 Qualitative evaluation of detection results using different classifiers: (a) RGB images of a bag containing powder, (b) manual annotations of observers with overlapping area between two observers ROIs marked with blue, and confidence images after pixel classification using: (c) AdaBoost classifier, (d) SVM-RBF, (e) Decision Trees classifier, and (f) SVM linear classifier

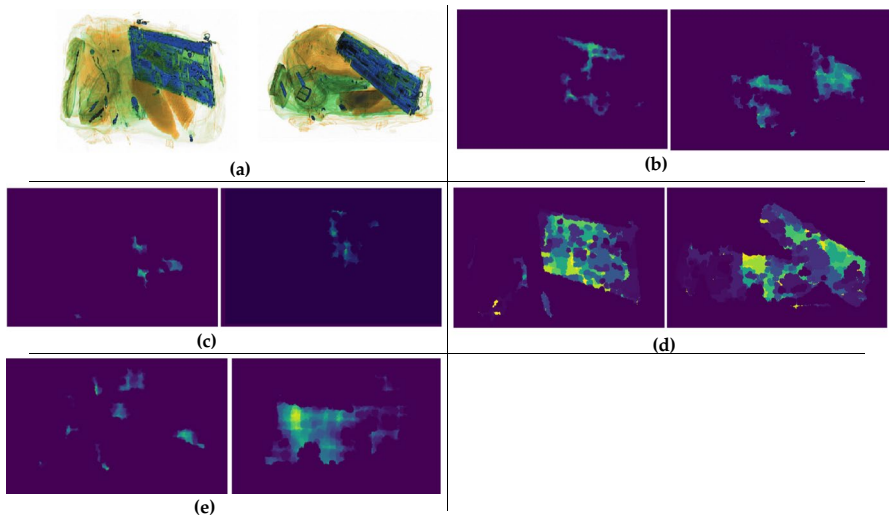


Fig. 18 Qualitative evaluation of detection results using different classifiers: (a) RGB images of a bag not containing powder, and confidence images after pixel classification using: (b) Ada-Boost classifier, (c) SVM-RBF, (d) Decision Trees classifier, and (e) SVM linear classifier

Compared with other classification methods, the SVM-RBF classifier shows a high pixel classification accuracy and a low number of pixels classified as false positives. In the example with no powder present (Fig. 18), a number of pixels are classified as positive, but the classification confidence is low and they are not well grouped, hence they would be discarded in the post processing step. This superior performance of the SVM with RBF kernel can be attributed to the fact that the problem addressed is not linear and the samples are not linearly separable in the complex feature space of 728 features. AdaBoost, and other boosting algorithms, although consisting of simple base classifiers, usually perform well on complex problems with classes that are not linearly separable (Huang et al. 2011; Friedman et al. 2000). Indeed, as shown in Fig. 12, the TPR of AdaBoost is equal to 1, as is the TPR of SVM-RBF classifier. This is also reflected in the qualitative evaluation of classifiers performance on a bag containing powder (Fig. 17). However, a high number of false positives reduces its performance which is reflected in a low precision on the Precision-Recall curve (Fig. 11) and also illustrated in Fig. 18 showing classifiers performance on a bag without powder.

The AdaBoost algorithm uses a number of “weak” classifiers and boosts their performance by an iterative procedure where in each stage classifiers are focusing more on mis-classified examples from the previous stage, in order to improve the global classification accuracy (Freund and Schapire 1997). The samples’ weights and the classifiers’ weights are updated for all samples, positive and negative, following the same set of rules, and the sole criteria is the error on the training set. If two “weak” classifiers have the same error, the same weights are going to be assigned to them, without taking into account their differences in probabilities of classifying positive and negative samples. An adjustment of the algorithm to take into account

false positive rate during an iterative weight adjusting procedure (Niyomugabo et al. 2016) or improving class imbalance problem by adding higher misclassification cost to minority class (Sun et al. 2007; Wang and Sun 2021; Zhou et al. 2017) could improve the results. Additionally, other modifications of boosting algorithms, could be more successful, such as GentleBoost (Friedman et al. 2000) that performs particularly well on problems with a significant class overlap.

Detection with SVM – RBF classifier and inter-observer variability

Inter-observer variability was comparable or even higher than the error of the automated detection on the test sets (Table 1 and Table 2). This was especially true for the precision of detection per powder (87% vs 74% and 91% vs 78%). As the number of TPs of *Observer 2* was usually higher than the TPs of the automated method (39 vs 33 and 60 vs 51), this lower precision indicates that the number of FPs of *Observer 2* was higher (14 vs 5 and 17 vs 5). *Observer 1* had a tendency to annotate only when fairly certain about the location of the powder, while *Observer 2* annotated more powders. This could explain why there is such a large number of FPs that *Observer 2* created. This could also explain the higher recall of *Observer 2* compared to that of the automated method (89% vs 75% and 91% vs 77%). *Observer 2* rarely missed a powder marked by the *Observer 1*, hence had a low number of FNs (5 vs 11 and 6 vs 15).

When the analysis was done per bag, the number of FPs created by the *Observer 2* and the automated method were equal: in both cases, and for both test set and final test set, equal to 1. The precision of *Observer 2* was equal or better than that of the automated method (96% vs 95% and 97% for both on the final test set). The Recall of the second observer remained higher, but the difference was reduced (92% vs 87% and 94% vs 91%). This is a logical consequence of the detection on one view of the bag being enough to achieve a TP – see the detection per bag algorithm (Fig. 9). Similarly, a FN for a powder on one view would not cause the whole bag to be a FN if there is a TP in the second view. Both detection of *Observer 2* and of the automated method improve with the detection per bag, where both views are taken into account. FPR was also calculated for the evaluation per bag, and in both sets, test and final test set, they are equal for *Observer 2* and the automated method (3% and 4%). Finally, *mAP* calculated over both classes was 88% (equal for each class).

The results overall are better in the final test set than on the test set. This could be explained by the fact that the tuning set on which the best classification threshold was chosen was included in the final test set. Regarding difficult types of bags, the two test sets were almost equal: bags with a laptop were almost the same percentage of both sets (12% and 13%), of thinly spread powders were 14% and 11%, and 50 g amounts constituted 10% and 10% of test set and final test respectively.

It is important to underline that the ground truth in this study was the annotations of *Observer 1*. It is therefore possible that according to the ground truth, there is no powder in the bag, hence, potentially a correct automated detection would be marked as a FP—see an example with a red rectangle in Fig. 16. Although it was

known for each bag whether the powder was present or not, it was necessary to be sure that the algorithm detected the powder and not some other object in that bag.

To the best of our knowledge, no other study focused on the automated detection of inorganic powders in X-ray images with additional electronic clutter added to make the bags more realistic. In (Kayalvizhi et al. 2022), explosive simulants and substances with similar Z_{eff} are automatically classified using a CNN-based algorithm and achieving a FPR of 1% and an *AUC* of 0.98% compared to a FPR of 3% and an *AUC* of 0.88 of the algorithm presented here. However, the bags used to evaluate the algorithm in Kayalvizhi et al. (2022) are of low complexity, with non-overlapping objects and containing only objects that the algorithm was trained on (explosive simulants and substances with similar Z_{eff}). Another two studies present methods for the automated detection of explosives (Morris et al. 2018; Chouai et al. 2020). In Morris et al. (2018), several CNN-based methods are tested and the best performance reported was an *AUC* of 0.95 and a FPR of 6%. In Chouai et al. (2020), an SVM using features chosen by an autoencoder achieved an Accuracy of 96% which is comparable with 94% achieved by the algorithm presented here (see Table 2). In both Morris et al. (2018) and Chouai et al. (2020), the datasets are not disclosed, nor were the explosives to be detected described due to security issues, but data used in Morris et al. (2018) does include high-complexity bags. Regarding processing time, the algorithm proposed by Morris et al. (2018) takes 350 ms to 450 ms, depending on the image size, which is significantly faster than the presented algorithm (8 s), while Kayalvizhi et al. (2022) reported 6 s processing time for pseudo-coloured images which is comparable to the results in this work.

Regarding published literature on the detection of other prohibited objects, perhaps the most similar to this study is the work of (Baştan et al. 2011). They detected handguns in 4 view X-ray images, using low-energy, high-energy and colour image and BoVW where feature descriptors used were SIFT, DoG and Harris. The best average precision achieved on 764 images was 57% and a TPR equal to 70%. In the improved version of the method (Baştan 2015; Baştan et al. 2013) where the geometry of the 4 scanner views was known and features were extracted from all 4 views for each bag, the performance improved to *mAP* equal to 66% for gun detection, 87% for laptop detection and 64% for bottle detection. Mery et al. (2016) achieved above 95% accuracy (different values depending on object occlusion) for detection of handguns, shuriken, razor blades, and clips on 100 images per class. This is better than the 94% accuracy of this work, however, the method was tested on patches of images, not images of cluttered bags. The same authors published a method for the detection of clips, springs and blades using 3D based k-NN classifier approach utilizing all views of the multi-view images (Mery et al. 2017) and achieving 100% precision and 93% recall for 15 blades and 96% precision and 85% recall for springs.

Regarding deep-learning based methods, mostly utilizing augmented data, including threat image projection (TIP), and transfer learning, (Bhowmik et al. 2019) achieved *mAP*=91% for firearm and firearm parts detection, (Kim et al. 2020) achieved *mAP*=91% for handgun (*AP*=91%), shuriken (*AP*=92%) and razor (*AP*=91%) detection. In all these studies, the objects detected were easier to see than the powders in this study. Additionally, better *mAP* is achieved using deep learning

approach with many more datasets used. Perhaps the explanation for the success of this presented method is that LBP texture descriptors were used for the detection of objects that are indeed recognizable by its texture, while the other methods deal with the detection of objects with prominent shape characteristics that would not have a high response to texture detectors, even when similar LBPs versions are used.

Conclusions and future work

In this paper, an automated method is presented for the detection of inorganic powders from dual-energy X-ray images of airport luggage. A novel dataset was created, with variable shapes and sizes of powders and with inserted additional electronic clutter in order to have more challenging and realistic dataset. The method has a comparable error rate to the inter-observer variability.

For the future, it would be interesting to explore possibilities of using raw low and high energy sensor data to make the algorithm more robust for usage with data from different scanners. Attempts to run this algorithm on data obtained from a different model of scanner with a different colour code resulted in poor results. Additionally, knowing the geometry of the scanner might improve the performance of the automated method. Furthermore, exhaustive parameter tuning could lead to better results and more insight on why certain algorithms perform better than the others. The speed of the algorithm could probably be increased by performing feature selection. Finally, data augmentation could be utilized to expand the existing dataset and enable the use deep-learning algorithms.

Authors' contributions D.V., M.R.O. and D.A. produced the images used in this work. D.V. performed the computer vision and machine learning. D.V. prepared the first draft of the manuscript. M.R.O. and D.A. reviewed the manuscript. D.A. was the project manager.

Funding Funded by the European Union's Horizon Europe research and innovation programme under JRC Direct Actions.

Data availability Not applicable.

Declarations

Ethical approval Not applicable.

Competing interests Not applicable.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Akçay S, Breckon TP (2017) An evaluation of region based object detection strategies within X-ray baggage security imagery. 2017 IEEE International Conference on Image Processing (ICIP), 1337–1341. <https://doi.org/10.1109/ICIP.2017.8296499>
- Akçay S, Kundegorski ME, Devereux M, Breckon TP (2016) Transfer learning using convolutional neural networks for object classification within X-ray baggage security imagery. 2016 IEEE International Conference on Image Processing (ICIP), 1057–1061. <https://doi.org/10.1109/ICIP.2016.7532519>
- 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 Trans Inf Forensics Secur* 13(9):2203–2215. <https://doi.org/10.1109/TIFS.2018.2812196>
- Baştan M (2015) Multi-view object detection in dual-energy X-ray images. *Mach vis Appl* 26(7–8):1045–1060. <https://doi.org/10.1007/s00138-015-0706-x>
- Baştan M, Byeon W, Breuel T (2013) Object Recognition in Multi-View Dual Energy X-ray Images. *Proc Br Mach Vision Conf 2013*:130.1–130.11. <https://doi.org/10.5244/C.27.130>
- Baştan M, Yousefi MR, Breuel TM (2011) Visual Words on Baggage X-Ray Images. In Real P, Diaz-Pernil D, Molina-Abril H, Berciano A, Kropatsch W (Eds.), *Computer Analysis of Images and Patterns* (Vol. 6854, pp. 360–368). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-23672-3_44
- Belongie S, Malik J, Puzicha J (2002) Shape matching and object recognition using shape context. *IEEE Trans Pattern Anal Mach Intell* 24(4):509–522. <https://doi.org/10.1109/34.993558>
- Benedykciuk E, Denkowski M, Dmitruk K (2021) Material Classification in X-Ray Images Based on Multi-Scale CNN. *SIVIP* 2021(15):1285–1293. <https://doi.org/10.1007/s11760-021-01859-9>
- Benedykciuk E, Denkowski M, Dmitruk K (2020) Learning-based Material Classification in X-ray Security Images: Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, 284–291. <https://doi.org/10.5220/0008951702840291>
- Bhowmik N, Wang Q, Gaus YFA, Szarek M (2019) The Good, the Bad and the Ugly: Evaluating Convolutional Neural Networks for Prohibited Item Detection Using Real and Synthetically Compositated X-ray Imagery. *British Machine Vision Conference, Workshop on Object Detection and Recognition for Security Screening*, 13. <https://doi.org/10.48550/arXiv.1909.11508>
- Bloice MD, Roth PM, Holzinger A (2019) Biomedical Image Augmentation Using Augmentor. *Bioinformatics* 2019(35):4522–4524. <https://doi.org/10.1093/bioinformatics/btz259>
- Chollet F (2017) Xception: Deep Learning with Depthwise Separable Convolutions. In *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE: Honolulu, HI, July 2017, pp. 1800–1807. <https://doi.org/10.48550/arXiv.1610.02357>
- Chouai M, Merah M, Sancho-Gómez J-L, Mimi M (2020) Supervised Feature Learning by Adversarial Autoencoder Approach for Object Classification in Dual X-Ray Image of Luggage. *J Intell Manuf* 31:1101–1112. <https://doi.org/10.1007/s10845-019-01498-5>
- Dai J, Li Y, He K, Sun J (2016) R-FCN: Object Detection via Region-based Fully Convolutional Networks. *Advances in Neural Information Processing Systems (NIPS)*, 379–387. Retrieved from <https://dl.acm.org/doi/10.5555/3157096.3157139>. Accessed Mar 2023
- Dalal N, Triggs B (2005) Histograms of Oriented Gradients for Human Detection. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 1, 886–893. <https://doi.org/10.1109/CVPR.2005.177>
- de Souza GB, da Silva Santos DF, Pires RG, Marana AN, Papa JP (2017) Deep Texture Features for Robust Face Spoofing Detection. *IEEE Trans Circuits Syst II Express Briefs* 64(12):1397–1401. <https://doi.org/10.1109/TCSII.2017.2764460>
- European Commission. (2021). Information for air travellers. https://transport.ec.europa.eu/transport-modes/air/aviation-security/information-air-travellers_en
- Fernández de la Ossa MÁ, Amigo JM, García-Ruiz C (2014) Detection of Residues from Explosive Manipulation by near Infrared Hyperspectral Imaging: A Promising Forensic Tool. *Forensic Sci Int* 242:228–235. <https://doi.org/10.1016/j.forsciint.2014.06.023>
- Franzel T, Schmidt U, Roth S (2012) Object Detection in Multi-view X-Ray Images. In Pinz A, Pock T, Bischof H, Leberl F (Eds.), *Pattern Recognition* (Vol. 7476, pp. 144–154). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-32717-9_15

- Freund Y, Schapire RE (1997) A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *J Comput Syst Sci* 55(1):119–139. <https://doi.org/10.1006/jcss.1997.1504>
- Friedman J, Hastie T, Tibshirani R (2000) Additive logistic regression: A statistical view of boosting (With discussion and a rejoinder by the authors). *Ann Stat* 28(2):337–407. <https://doi.org/10.1214/aos/1016218223>
- Ghalati MK, Nunes A, Ferreira H, Serranho P, Bernardes R (2021) Texture Analysis and its Applications in Biomedical Imaging: A Survey. *IEEE Rev Biomed Eng*, 1–1. <https://doi.org/10.1109/RBME.2021.3115703>
- Gittinger JM, Suknot AN, Jimenez ES, Spaulding TW, Wenrich SA (2018) Passenger Baggage Object Database (PBOD). Provo, Utah, USA, 2018, p. 230021. <https://doi.org/10.1063/1.5031668>
- Hancock PA, Hart SG (2002) Defeating Terrorism: What Can Human Factors/Ergonomics Offer? *Ergon Des* 10:6–16. <https://doi.org/10.1177/106480460201000103>
- Harding G, Fleckenstein H, Kosciuszka D, Olesinski S, Strecker H, Theedt T, Zienert G (2012) X-Ray Diffraction Imaging with the Multiple Inverse Fan Beam Topology: Principles, Performance and Potential for Security Screening. *Appl Radiat Isot* 2012(70):1228–1237. <https://doi.org/10.1016/j.apradiso.2011.12.015>
- Hassan T, Khan SH, Akçay S, Bennamoun M, Werghi N (2019) Deep CMST Framework for the Autonomous Recognition of Heavily Occluded and Cluttered Baggage Items from Multivendor Security Radiographs. *Comput Sci*, 18. <https://doi.org/10.48550/arXiv.1912.04251>
- Hättenschwiler N, Sterchi Y, Mendes M, Schwaninger A (2018) Automation in Airport Security X-Ray Screening of Cabin Baggage: Examining Benefits and Possible Implementations of Automated Explosives Detection. *Appl Ergon* 2018(72):58–68. <https://doi.org/10.1016/j.apergo.2018.05.003>
- He K, Zhang X, Ren S, Sun J (2016) Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE: Las Vegas, NV, USA, June 2016, pp. 770–778. <https://doi.org/10.48550/arXiv.1512.03385>
- Huang D, Shan C, Ardabilian M, Wang Y, Chen L (2011) Local Binary Patterns and Its Application to Facial Image Analysis: A Survey. *IEEE Trans Syst Man Cyberne, Part C* 41(6):765–781. <https://doi.org/10.1109/TSMCC.2011.2118750>
- Huegli D, Merks S, Schwaninger A (2020) Automation Reliability, Human-Machine System Performance, and Operator Compliance: A Study with Airport Security Screeners Supported by Automated Explosives Detection Systems for Cabin Baggage Screening. *Appl Ergon* 2020 86:103094. <https://doi.org/10.1016/j.apergo.2020.103094>
- Kayalvizhi R, Malarvizhi S, Choudhury SD, Topkar A (2022) Automated Detection of Threat Materials in X-Ray Baggage Inspection Systems (XBISs). *IEEE Trans Nucl Sci* 2022(69):1923–1930. <https://doi.org/10.1109/TNS.2022.3182771>
- Khotanzad A, Hong YH (1990) Invariant image recognition by Zernike moments. *IEEE Trans Pattern Anal Mach Intell* 12(5):489–497. <https://doi.org/10.1109/34.55109>
- Kim J, Kim J, Ri J (2020) Generative adversarial networks and faster-region convolutional neural networks based object detection in X-ray baggage security imagery. *OSA Continuum* 3(12):3604. <https://doi.org/10.1364/OSAC.412523>
- Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet classification with deep convolutional neural networks. *Proc Adv Neural Inf Process Syst*, 25(6): 1090–1098. Retrieved from <https://proceedings.neurips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf>. Accessed 1 Mar 2023
- Liu L, Fieguth P, Guo Y, Wang X, Pietikäinen M (2017) Local binary features for texture classification: Taxonomy and experimental study. *Pattern Recogn* 62:135–160. <https://doi.org/10.1016/j.patcog.2016.08.032>
- Makhzani A, Shlens J, Jaitly N, Goodfellow I, Frey B (2016) Adversarial Autoencoders. [arXiv:1511.05644](https://arxiv.org/abs/1511.05644) [cs] 2016. <https://doi.org/10.48550/arXiv.1511.05644>
- Mery D, Riffo V, Zscherpel U, Mondragón G, Lillo I, Zuccar I, Lobel H, Carrasco M (2015) GDXray: The Database of X-Ray Images for Nondestructive Testing. *J Nondestruct Eval* 2015(34):42. <https://doi.org/10.1007/s10921-015-0315-7>
- Mery D, Svec E, Arias M (2016) Object Recognition in X-ray Testing Using Adaptive Sparse Representations. *J Nondestruct Eval* 35(3):45. <https://doi.org/10.1007/s10921-016-0362-8>
- Mery D, Riffo V, Zuccar I, Pieringer C (2017) Object recognition in X-ray testing using an efficient search algorithm in multiple views. *Insight - Nondestruct Test Cond Monit* 59(2):85–92. <https://doi.org/10.1784/insi.2017.59.2.85>

- Miao C, Xie L, Wan F, Su C, Liu H, Jiao J, Ye Q (2019) SIXray: A Large-Scale Security Inspection X-Ray Benchmark for Prohibited Item Discovery in Overlapping Images. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE: Long Beach, CA, USA, June 2019, pp. 2114–2123. <https://doi.org/10.48550/arXiv.1901.00303>
- Monard MC, Batista GEAPA (2002) Learning with Skewed Class Distributions. *Adv Log Artif Intell Robot* 85:173–180
- Morris T, Chien T, Goodman E (2018) Convolutional Neural Networks for Automatic Threat Detection in Security X-Ray Images. In Proceedings of the 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), IEEE: Orlando, FL, December 2018, pp. 285–292. <https://doi.org/10.1109/ICMLA.2018.00049>
- Niyomugabo C, Choi H, Kim TY (2016) A Modified AdaBoost Algorithm to Reduce False Positives in Face Detection. *Math Probl Eng* 2016:1–6. <https://doi.org/10.1155/2016/5289413>
- Ojala T, Pietikäinen M, Harwood D (1996) A Comparative Study of Texture Measures with Classification Based on Feature Distributions. *Pattern Recogn* 29(1):51–59. [https://doi.org/10.1016/0031-3203\(95\)00067-4](https://doi.org/10.1016/0031-3203(95)00067-4)
- Ojala T, Pietikäinen M, Maenpää T (2002) Multiresolution grey-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans Pattern Anal Mach Intell* 24(7):971–987. <https://doi.org/10.1109/TPAMI.2002.1017623>
- Otsu N (1979) A Threshold Selection Method from Grey-Level Histograms. *IEEE Trans Syst Man Cybern* 9(1):62–66. <https://doi.org/10.1109/TSMC.1979.4310076>
- Pietikäinen M, Heikkilä J (2011) Tutorial: Image and Video Description with Local Binary Pattern Variants. Conference on Computer Vision and Pattern Recognition, CVPR. Retrieved from <https://www.scribd.com/document/373690056/Image-and-Video-Description-with-Local-Binary-Pattern-Variants-CVPR-Tutorial-Final>. Accessed 1 Mar 2023
- Radford A, Metz L, Chintala S (2016) Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *arXiv:1511.06434* [cs] 2016. <https://doi.org/10.48550/arXiv.1511.06434>
- Redmon J, Farhadi A (2018) YOLOv3: An Incremental Improvement. *arXiv:1804.02767* [cs] 2018. <https://doi.org/10.48550/arXiv.1804.02767>
- Ren S, He K, Girshick R, Sun J (2017) Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans Pattern Anal Mach Intell* 39(6):1137–1149. <https://doi.org/10.1109/TPAMI.2016.2577031>
- Rokach L, Maimon O (2005) Decision Trees. In O. Maimon & L. Rokach (Eds.), *Data Mining and Knowledge Discovery Handbook* (pp. 165–192). Springer US. https://doi.org/10.1007/0-387-25465-X_9
- Roomi MM (2012) Detection of Concealed Weapons in X-Ray Images Using Fuzzy K-NN. *IJCSEIT* 2:187–196. <https://doi.org/10.5121/ijcseit.2012.2216>
- Rosenfeld A, Pfaltz JL (1966) Sequential Operations in Digital Picture Processing. *J ACM* 13(4):471–494. <https://doi.org/10.1145/321356.321357>
- Schapire RE (2013) Explaining AdaBoost. In Schölkopf B, Luo Z, Vovk V (Eds.), *Empirical Inference* (pp. 37–52). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-41136-6_5
- Schmidt-Hackenberg L, Yousefi MR, Breuel TM (2012) Visual cortex inspired features for object detection in X-ray images. In Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), Tsukuba, Japan, 2573–2576, Retrieved from <https://ieeexplore.ieee.org/document/6460693>. Accessed 1 Mar 2023
- Sharifi O, Mokhtarzade M, Asghari Beirami B (2020) A Deep Convolutional Neural Network based on Local Binary Patterns of Gabor Features for Classification of Hyperspectral Images. 2020 International Conference on Machine Vision and Image Processing (MVIP), 1–5. <https://doi.org/10.1109/MVIP49855.2020.9187486>
- Simonyan K, Zisserman A (2015) Very Deep Convolutional Networks for Large-Scale Image Recognition. Conference Track Proceedings: San Diego, CA, USA, May 2015. <https://doi.org/10.48550/arXiv.1409.1556>
- Soille P, Burger A, De Marchi D, Kempeneers P, Rodriguez D, Syrris V, Vasilev V (2018) A versatile data-intensive computing platform for information retrieval from big geospatial data. *Futur Gener Comput Syst* 81:30–40. <https://doi.org/10.1016/j.future.2017.11.007>
- Sun Y, Kamel MS, Wong AKC, Wang Y (2007) Cost-sensitive boosting for classification of imbalanced data. *Pattern Recogn* 40(12):3358–3378. <https://doi.org/10.1016/j.patcog.2007.04.009>

- Sun Y, Wong AKC, Kamel MS (2009) Classification of Imbalanced Data: A Review. *Int J Pattern Recognit Artif Intell* 23(04):687–719. <https://doi.org/10.1142/S0218001409007326>
- Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z (2016) Rethinking the Inception Architecture for Computer Vision. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE: Las Vegas, NV, USA, June 2016, pp. 2818–2826. <https://doi.org/10.48550/arXiv.1512.00567>
- Transport Security Administration. (2021). What Can I Bring? Retrieved from <https://www.tsa.gov/travel/security-screening/whatcanibring/all>. Accessed 1 Nov 2022
- Turcsany D, Mouton A, Breckon TP (2013) Improving feature-based object recognition for X-ray baggage security screening using primed visual words. *2013 IEEE International Conference on Industrial Technology (ICIT)*, 1140–1145. <https://doi.org/10.1109/ICIT.2013.6505833>
- Vapnik VN (1999) *The Nature of Statistical Learning Theory*. Springer science & business media New York. <https://doi.org/10.1007/978-1-4757-2440-0>
- Vukadinovic D, Anderson D (2022) X-ray Baggage Screening and Artificial Intelligence (AI), EUR 31123 EN, Publications Office of the European Union, Luxembourg, JRC129088. <https://doi.org/10.2760/46363>
- Wang W, Sun D (2021) The improved AdaBoost algorithms for imbalanced data classification. *Inf Sci* 563:358–374. <https://doi.org/10.1016/j.ins.2021.03.042>
- Xi M, Chen L, Polajnar D, Tong W (2016) Local binary pattern network: A deep learning approach for face recognition. *2016 IEEE International Conference on Image Processing (ICIP)*, 3224–3228. <https://doi.org/10.1109/ICIP.2016.7532955>
- Zhou B, Wang T, Luo M, Pan S (2017) An online tracking method via improved cost-sensitive adaboost. *2017 Eighth International Conference on Intelligent Control and Information Processing (ICICIP)*, 49–54. <https://doi.org/10.1109/ICICIP.2017.8113916>

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