A Logarithmic X-ray Imaging Model for Baggage Inspection: Simulation and Object Detection

Domingo Mery
Department of Computer Science
Pontificia Universidad Católica de Chile
Av. Vicuña Mackenna 4860(143) – Santiago de Chile
domingo.mery@uc.cl

Aggelos K. Katsaggelos
Department of Electrical Engineering and Computer Science
Northwestern University
Evanston, IL 60208, USA
aggk@eecs.northwestern.edu

Abstract

In the last years, many computer vision algorithms have been developed for X-ray testing tasks. Some of them deal with baggage inspection, in which the aim is to detect automatically target objects. The progress in automated baggage inspection, however, is modest and very limited compared to what is needed because X-ray screening systems are still being manipulated by human inspectors. In this work, we present an X-ray imaging model that can separate foreground from background in baggage screening. The model can be used in two main tasks: i) Simulation of new X-ray images, where simulated images can be used in training programs for human inspectors, or can be used to enhance datasets for computer vision algorithms. ii) Detection of (threat) objects, where new algorithms can be employed to perform automated baggage inspection or to aid an user in the inspection task showing potential threats. In our model, rather than a multiplication of foreground and background, that is typically used in X-ray imaging, we propose the addition of logarithmic images. This allows the use of linear strategies to superimpose images of threat objects onto X-ray images and the use of sparse representations in order to segment target objects. In our experiments, we simulate new X-ray images of handguns, shuriken and razor blades, in which it is impossible to distinguish simulated and real X-ray images. In addition, we show in our experiments the effective detection of shuriken, razor blades and handguns using the proposed algorithm outperforming some alternative state-of- the-art techniques.

1. Introduction

Baggage inspection using X-ray screening is a priority task that reduces the risk of crime, terrorist attacks and propagation of pests and diseases [29]. Security and safety screening with X-ray scanners has become an important process in public spaces and at border checkpoints [18]. However, as shown in Fig. 1, inspection is a complex task because threat items are very difficult to detect when placed in closely packed bags, occluded by other objects, or rotated, thus presenting an unrecognizable view [2]. Manual detection of threat items by human inspectors is extremely demanding [24]. It is tedious because very few bags actually contain threat items, and it is stressful because the work of identifying a wide range of objects, shapes and substances (metals, organic and inorganic substances) takes a great deal of concentration. In addition, human inspectors receive only minimal technological support. Furthermore, during rush hours, they have only a few seconds to decide whether or not a bag contains a threat item [1]. Since each operator must screen many bags, the likelihood of human error becomes considerable over a long period of time even with intensive training. The literature suggests that detection performance is only about 80–90% [15]. In baggage inspection, automated X-ray testing remains an open

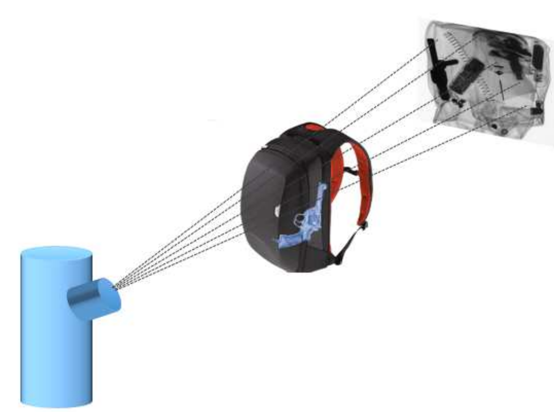

Figure 1. Setup of an X-ray imaging system, the X-ray source irradiates the object (a bag containing a handgun) and produces an X-ray image.

question due to: *i) loss of generality*, which means that approaches developed for one task may not transfer well to another; *ii) deficient detection accuracy*, which means that there is a fundamental tradeoff between false alarms and missed detections; *iii) limited robustness* given that requirements for the use of a method are often met for simple structures only; and *iv) low adaptiveness* in that it may be very difficult to accommodate an automated system to design modifications or different specimens. There are some contributions in computer vision for X-ray testing such as applications on inspection of castings, welds, food, cargos and baggage screening [10] and very interesting advances in automated recognition of objects [14] that evaluates ten different modern computer vision algorithms. Nevertheless, as we can see in the literature review [21, 27, 30, 11, 20, 6, 17, 4, 9, 5, 16, 13], the progress in automated baggage inspection is far from being perfected given that the appearance of the object of interest can become extremely difficult due to problems of (self)occlusion, noise, acquisition, clutter, etc. We believe, however, that computer vision algorithms can be improved if we could model the X-ray images as a contribution of background and foreground. Thus, the detection can be achieved by analyzing the foreground image only.

In this work, we present an imaging model that can separate foreground from background of an X-ray image. The model can be used in the detection of (threat) objects, where new algorithms can be employed to perform automated baggage inspection or to aid an user in the inspection task showing potential threats. In our model, rather than a multiplication of foreground and background, that is typically used in X-ray imaging, we propose the addition of logarithmic images. This allows the use of linear strategies such as sparse representations in order to segment target objects.

In addition, the model can be used in the simulation of new X-ray images, where simulated images can be used in training programs for human inspectors, or can be used to enhance datasets for computer vision algorithms. Similarly, this allows the use of linear strategies to superimpose images of threat objects onto X-ray images. The simulated X-ray images should be as similar as possible to real X-ray images. In the literature, the simulation attempts to model the physics of the X-ray formation (generation, interaction and detection) and handle complex 3D objects efficiently [25]. State–of–the–art of computer modeling of X-ray testing methods are able to simulate different X-ray spectrum and X-ray source size, varied photon–matter interactions, and several X-ray detector responses [10]. Special attention has been given to general purpose Monte Carlo methods that are able to calculate higher order scattering events [19, 22, 28]. A computer simulator for X-ray testing should include the following modules [23]: source model, ray-tracing engine, material data base, straight line attenuation model and detector model. Rather than a complex simulation system, that simulates the whole imaging process, we propose a simple strategy that takes advantage of real X-ray images. Thus, in our approach, using the proposed model, we start with an original X-ray image, *e.g.* acquiered from a cluttered bag, and we superimpose onto the original image a real X-ray image of a threat object. A similar approach based on superimposition was suggested for the simulation of defects in X-ray images of castings and weldings [3], in which a synthetic CAD model of a defect was superimposed onto a real X-ray image.

In our experiments, we simulate new X-ray images of handguns, shuriken and razor blades, in which it is impossible to distinguish simulated and real X-ray images. In addition, we show in our experiments the effective detection of shuriken, razor blades and handguns using the proposed algorithm.

The rest of the paper is organized as follows: Section 2 describes the proposed model and the simulation approach, Section 3 outlines the proposed method for threat object detection, Section 4 shows the experimental results, and finally Section 5 gives concluding remarks.

2. X-ray Imaging Model

Typically, X-ray imaging can be modeled using the absorption law which characterizes the intensity distribution of X-rays through matter [8]:

$$\varphi(d) = \varphi_0 e^{-\mu d} \tag{1}$$

with μ absorption coefficient, d thickness of the irradiated matter, φ_0 incident energy flux density, and φ energy flux density after passage through matter with the thickness of d. In Fig. 2, we can see an example with $n = 3$ materials that can be modeled by:

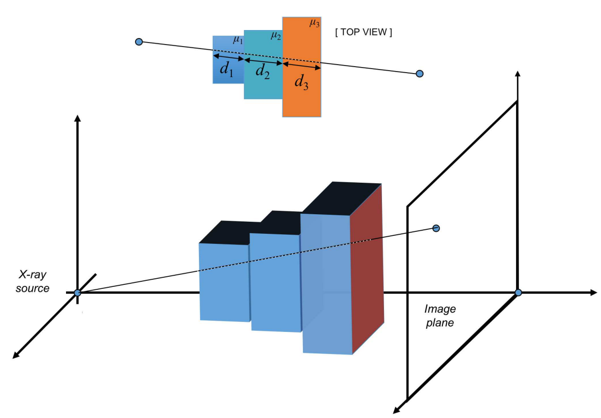

Figure 2. The X-ray beam passes through materials with linear absorption coefficients μ_1, μ_2 and μ_3.

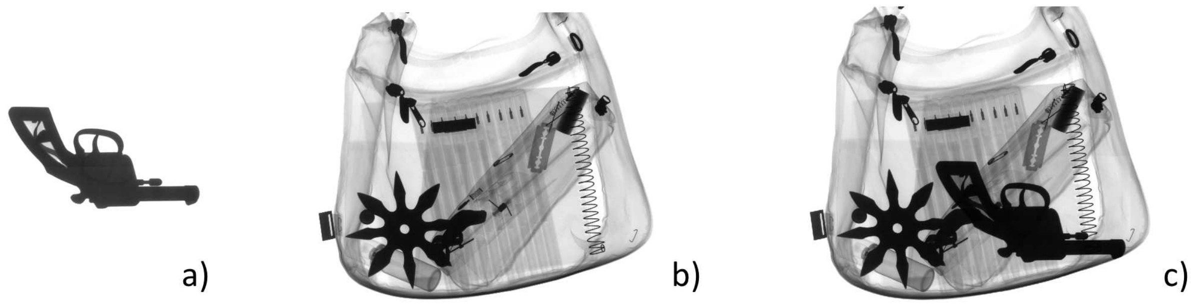

Figure 3. Superimposition of an X-ray image of a handgun onto an X-ray image of a cluttered bag: a) I_f: Foreground (threat object). b) I_b: Background (cluttered bag). c) I_t: Total (bag with threat object).

$$\varphi = \varphi_0 \exp\left(-\sum_{i=1}^{n} \mu_i d_i\right). \tag{2}$$

In an X-ray digital image, the grayvalue of a pixel can be linearly modeled as [3]:

$$I = A \cdot \varphi + B. \tag{3}$$

where A and B are constant parameters of the model.

Following the models (2) for the energy flux density and (3) for the digital image, it is possible to model the X-ray image of the foreground (I_f), *e.g.* a handgun, and the background (I_b), *e.g.* a cluttered bag, as illustrated in Figures 1 and 3. Thus,

$$I_f = A \cdot \varphi_f + B \qquad I_b = A \cdot \varphi_b + B \tag{4}$$

where

$$\varphi_f = \varphi_0 e^{-\mu_f d_f} \qquad \varphi_b = \varphi_0 e^{-\mu_b d_b} \tag{5}$$

in this case μ_f and μ_b are the absorption coefficients of the foreground and background respectively. It is worth mentioning, that $\mu_b d_b$ represents $\sum_j \mu_j d_j$ considering all cluttered objects j that lie on the X-ray beam shown in Fig. 2. The total X-ray image, called I_t, can be modeled as

$$\varphi_t = \varphi_0 e^{-\mu_f d_f} e^{-\mu_b d_b} \tag{6}$$

$$I_t = A \cdot \varphi_t + B = C e^{-\mu_f d_f} e^{-\mu_b d_b} + B \tag{7}$$

where $C = A \cdot \varphi_0$. It is clear, that (7) can be used to simulate new X-ray images from I_f and I_b: replacing (5) in (4), we obtain

$$e^{-\mu_f d_f} = \frac{I_f - B}{C} \qquad e^{-\mu_b d_b} = \frac{I_b - B}{C}. \tag{8}$$

From (8) and (7), it yields

$$\frac{I_t - B}{C} = \frac{I_f - B}{C} \cdot \frac{I_b - B}{C}. \tag{9}$$

We can normalize the X-ray images by subtracting B and dividing by C, *e.g.*, $J_t = (I_t - B)/C$. Thus, using the normalized images for total, foreground and background images, we obtain:

$$J_t = J_f \cdot J_b. \tag{10}$$

Easily, we can compute the total image by

$$I_t = C \cdot J_f \cdot J_b + B. \tag{11}$$

Indeed, image I_t in Fig. 3c was simulated from I_f and I_b in Fig. 3a and 3b respectively using (11).

3. Application: Object detection

In this section, we explain how to detect threat objects using a sparse representation based on the model of Section 2. The keyidea of our strategy is to use a dictionary for the foreground and another dictionary for the background. In the foreground images, only the threat objects to be detected are present, whereas in the background, the rest of the objects are present. In this classic strategy, a testing image can be modeled by the sum of the contributions of the foreground and the background, so the detection of the threat objects can be easily achieved by analyzing the contribution of foreground only. However, in our model (10) the testing image is obtained by a multiplication –not by an addition– of two images. We can solve this problem, if we take the logarithm of both sides of the equation (see Fig. 4). Thus,

$$Z_t = Z_f + Z_b. \tag{12}$$

where the three variables are $\log(J_t)$, $\log(J_f)$, $\log(J_b)$ respectively. We call these images the *logarithmic images*. In general, the logarithmic image Z of an X-ray image I is computed by

$$Z(J) = \log\left(\frac{I - B}{C}\right). \tag{13}$$

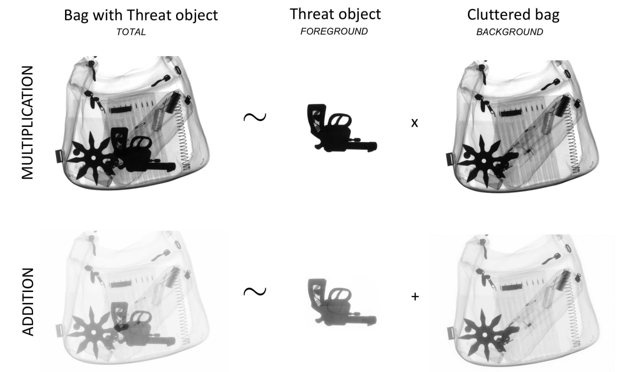

Figure 4. The total image I_t can be modeled by multiplication or addition, in first case the original images are used, whereas in the second case we considere the logarithmic images.

We believe that algorithms based on sparse representations can be used for this general task because in recognition applications, under the assumption that natural images can be represented using sparse decomposition, state-of-the-art results have been significantly improved [26]. In the sparse representation approach, a dictionary is built from the training X-ray images, and matching is done by reconstructing the test image using a sparse linear combination of the dictionary. Usually, the test image is assigned to the class with the minimal reconstruction error.

In our approach, we build two dictionaries: $\mathbf{D}_f$ for the foreground and $\mathbf{D}_b$ for the background. The first one is obtained from patches of the foreground (see Fig. 5), the second one from patches of the background (see Fig. 6). In both cases, representative X-ray images are used. In our approach, we extract a patch $\mathbf{z}$ of size $w \times w$ pixels of the logarithmic images subsampled by a pixels in both directions. For the dictionaries, we use vector $\mathbf{y}$ of $(w/a)^2$ elements that corresponds to the intensity values of $\mathbf{z}$ given by stacking its columns normalized to unit length in order to deal with different contrast conditions. This operation is represented by function f:

$$\mathbf{y} = f(\mathbf{z}). \tag{14}$$

Dictionaries $\mathbf{D}_f$ and $\mathbf{D}_b$ have n_f and n_b columns respectively. Both dictionaries are concatenated as shown in Fig. 7 yielding a new dictionary of $n = n_f + n_b$ columns and $(w/a)^2$ rows:

$$\mathbf{D} = [\mathbf{D}_f \quad \mathbf{D}_e]. \tag{15}$$

In testing stage, we use the well-known methodology of sliding windows, *i.e.*, a small window $\mathbf{z}$ of $w \times w$ pixels (the same size of the patches used in the dictionaries) is sledded over the testing image in both horizontal and vertical directions, and for each localization of $\mathbf{z}$, first we compute $\mathbf{y}$ using transformation (14), and second we calculate $\mathbf{x}$ the

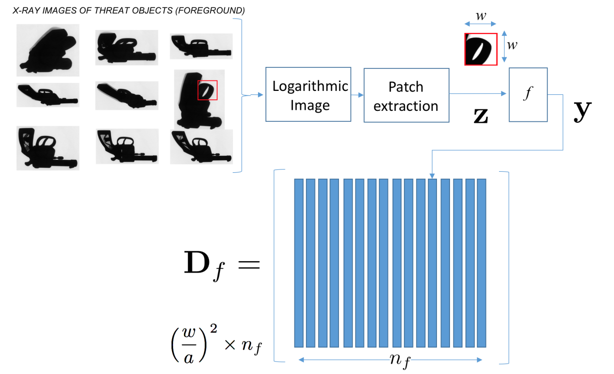

Figure 5. Dictionary for foreground: patches of X-ray images of the detection object are extracted and processed.

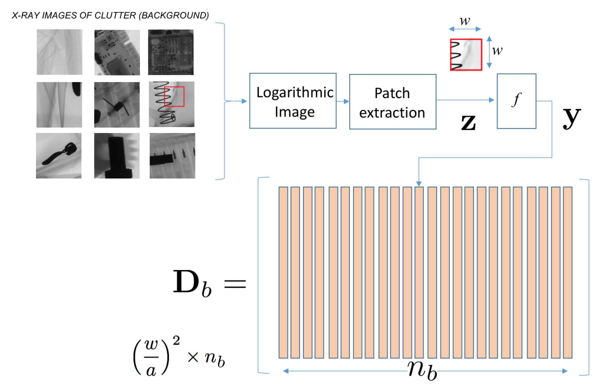

Figure 6. Dictionary for background: patches of X-ray images of cluttered bags with no detection object are extracted and processed.

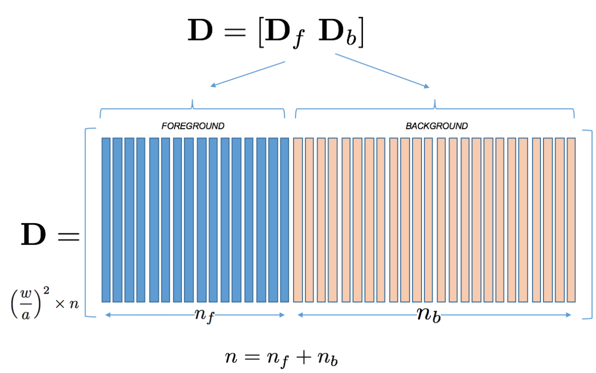

Figure 7. The dictionary is built by concatenating both foreground and background dictionaries.

sparse representation of $\mathbf{y}$ using dictionary $\mathbf{D}$:

$$||\mathbf{y} - \mathbf{D}\mathbf{x}|| \rightarrow \min \quad \text{subject to } ||\mathbf{x}||_0 \leq T, \tag{16}$$

where $||\mathbf{x}||_0$ means the number of non-zero elements of $\mathbf{x}$. Thus, in the sparse representation, no more than T elements

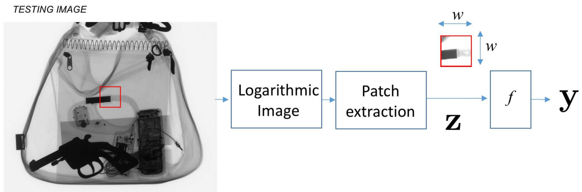

Figure 8. A testing patch is processed.

of $\mathbf{x}$ are allowed to be non-zero. According to (15) and (16), sparse vector $\mathbf{x}$ has $n = n_f + n_b$ elements: the first n_f and the last n_b elements corresponds of the contribution to the foreground and background respectively. Thus, vector $\mathbf{x}$ can be decomposed as:

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_f \\ \mathbf{x}_b \end{bmatrix}, \tag{17}$$

the first one with n_f elements, and the last one with n_b elements. We can investigate, how representations $\mathbf{x}_f$ and $\mathbf{x}_b$ can reconstruct the original patch $\mathbf{y}$ using dictionaries $\mathbf{D}_f$ and $\mathbf{D}_b$ respectively.

$$\mathbf{y}_f = \mathbf{D}_f \mathbf{x}_f, \quad \mathbf{y}_b = \mathbf{D}_b \mathbf{x}_b. \tag{18}$$

The idea here, is the following: for example if the reconstruction $\mathbf{y}_f$ is good enough, that means that the reconstruction error $||\mathbf{y} - \mathbf{y}_f||$ is minimal, then the patch can be classified as foreground, otherwise as background. The reconstruction error for both cases are defined as:

$$e_f = ||\mathbf{y} - \mathbf{y}_f||, \quad e_b = ||\mathbf{y} - \mathbf{y}_b||. \tag{19}$$

The proposed algorithm is simple and it is shown in Algorithm 1. It follows the sliding-windows strategy. Thus, in each location of the sliding window, we extract a patch $\mathbf{z}$ that is transformed into a vector $\mathbf{y}$ which is represented as a sparse vector $\mathbf{x}$. The idea is to investigate the contribution of $\mathbf{x}$ using both foreground and background dictionaries ($\mathbf{D}_f$ and $\mathbf{D}_e$ respectively). If the reconstruction error of the foreground information is low enough and less than the reconstruction error of the background information, then our algorithm concludes that a part of the threat object is detected. Finally, we select those regions that are large enough and have enough parts that were detected.

4. Experimental Results

In this work, the main contribution is a new model that can be used in X-ray imaging. In this section, we show two experiments: the first one corresponds to the new methodology that can be used to simulate X-ray images following equation (11), the second one refers to the new approach to detect threat objects in X-ray images using Algorithm 1.

Algorithm 1 Detection of Threat Objects

Input: Testing X-ray image $\mathbf{I}$, dictionary $\mathbf{D}$ and threshold θ
Initialize Matrices $\mathbf{Z}_f$, $\mathbf{E}_f$, $\mathbf{E}_b$ with zero (same size as $\mathbf{I}$)
$\mathbf{Z} \leftarrow$ logarithmic image of $\mathbf{I}$ using (13)
Sliding-windows:
    $(i, j) \leftarrow$ center position of the patch
    $\mathbf{z} \leftarrow$ extracted patch of $\mathbf{Z}$ centered in (i, j)
    $\mathbf{y} \leftarrow$ transformation of $\mathbf{z}$ using (14)
    $\mathbf{x} \leftarrow$ sparse representation of $\mathbf{y}$ using (16)
    $\mathbf{y}_f, \mathbf{y}_b \leftarrow$ reconstruction using (18)
    $e_f, e_b \leftarrow$ reconstruction error using (19)
    add e_f to $\mathbf{E}_f$ in all locations of the patch centered in (i, j)
    add e_b to $\mathbf{E}_b$ in all locations of the patch centered in (i, j)
    if $e_f < e_b$
        $\mathbf{z}_f \leftarrow$ transformation of $\mathbf{y}_f$ into a squared patch
            (same size as $\mathbf{z}$ using interpolation)
        add $\mathbf{z}_f$ to $\mathbf{Z}_f$ in location centered in (i, j)
    end
end
Detection: $\mathbf{R}$ = AND$(\mathbf{E}_f < \theta, \mathbf{Z}_f > 0)$
Output: $\mathbf{S} \leftarrow$ large regions of $\mathbf{R}$.

Figure 9. Simulation of a handgun superimposed onto an X-ray image of a bag. The original X-ray image (I_b) has only one shuriken, whereas the simulated X-ray image (I_t) has an additional handgun in different poses (see the handgun in the middle of the image). Bottom image shows a zoom in of a simulated image.

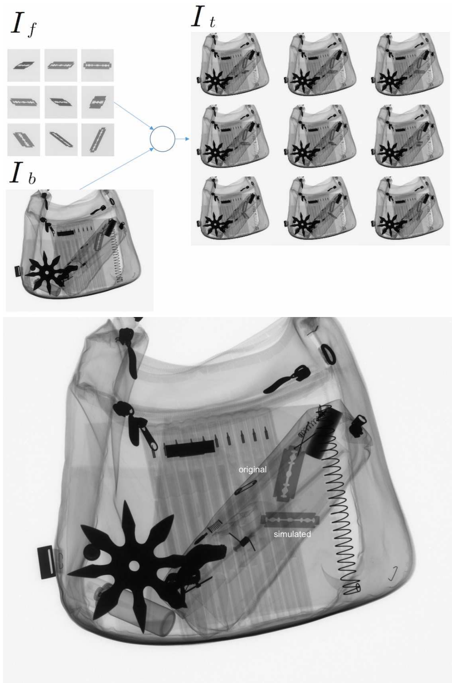

Figure 10. Simulation of a razor blade superimposed onto an X-ray image of a bag. The original X-ray image (I_b) has only one shuriken, whereas the simulated X-ray image (I_t) has an additional razor blade in different poses (see both razor blades together in the image). Bottom image shows a zoom in of a simulated image.

Figure 11. Simulation of a shuriken superimposed onto an X-ray image of a bag. The original X-ray image (I_b) has only one shuriken, whereas the simulated X-ray image (I_t) has an additional shuriken in different poses (see the shuriken partial occluded by the gun). Bottom image shows a zoom in of a simulated image.

In the following two sections, we show some experiments that validate the proposed model and the methods. All X-ray images used in our experiments belong to the $\mathbb{GDX}$ray[1] database [12].

4.1. Simulation

In this Section we show how to simulate X-ray images. Simulated images can be used in training programs for human inspectors, or can be used to enhance datasets for computer vision algorithms. The idea is simple, we have to acquire X-ray images of objects that are completely isolated and then we can superimposed them onto X-ray images of cluttered bags. In order to acquire isolated X-ray images, the threat object can be located inside a sphere of expanded polystyrene (EPS)[2] as suggested in [21]. In $\mathbb{GDX}$ray we have those kind of images, where a threat object is irradiated from different points of views. Thus, the threat object can be superimposed in many different poses.

In order to illustrate the similarity between original and simulated X-ray images, we show experiments where the original X-ray image has only one threat object and the simulated image has the original threat object and the superimposed threat object, so in the same image we can compare both of them. We tested with the following threat objects: handguns, razor blades and shuriken (ninja stars) in nine different poses. The results are given in Figures 9, 10 and 11 respectively. In our results, the reader can see both threat objects –simulated and original–, and can conclude that both objects are so similar that it is imposible to say which one is the simulated and which one is the original

4.2. Object detection

In this Section, we show some experiments using the method explained in Section 3 to detect threat objects in baggage screening. In our experiments, we detect handguns, razor blades and shuriken using sparse representations. In order to illustrate step by step, in Fig. 12, steps of Algorithm 1 are shown. We can see the effectiveness of the proposed method using the sliding-window approach and the classification basen on sparse representations and reconstruction error. In addition, we show the output of of Algorithm 1 for a shuriken and a handgun in Figure 13 and 14 respectively. Again, the effectiveness of the proposed method is validated.

[1] $\mathbb{GDX}$ray is a public database for X-ray testing with more than 20.000 images. The X-ray images included in $\mathbb{GDX}$ray can be used free of charge, for research and educational purposes only. Available at `http://dmery.ing.puc.cl/index.php/material/gdxray/`.

[2] EPS is used due to its low X-ray absorption coefficient.

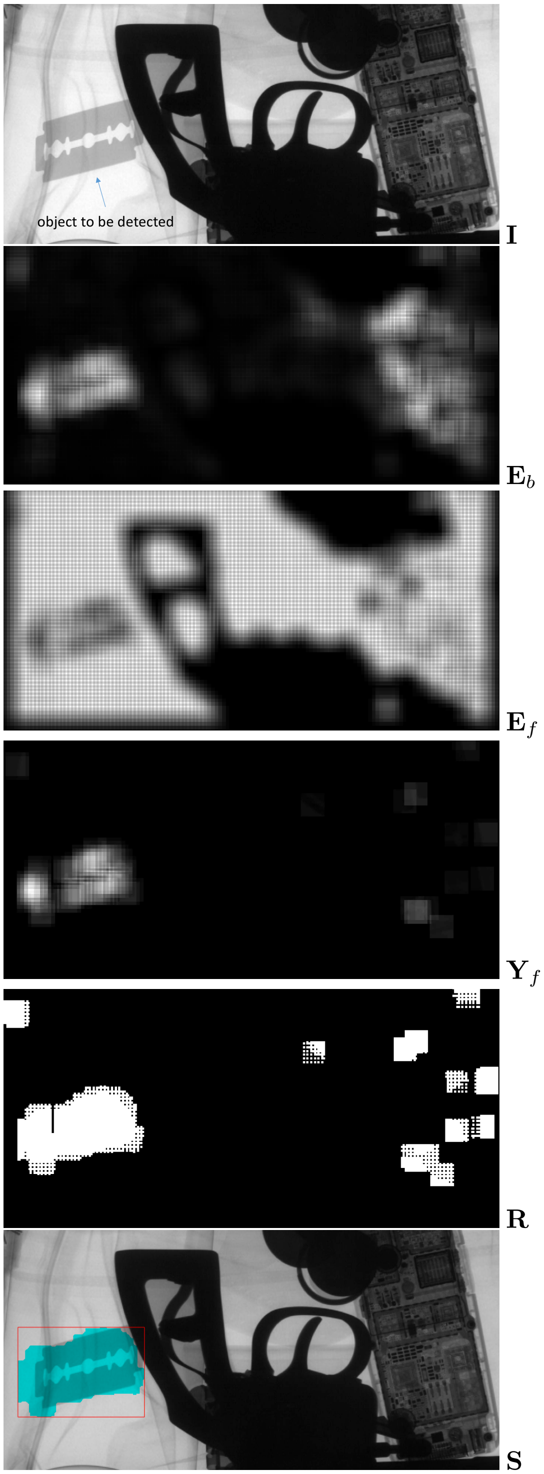

Figure 12. Steps of the detection of a razor blade using proposed algorithm 1. In this example $\mathbf{E}_f$ and $\mathbf{E}_b$ are the reconstruction error of foreground and background respectively. We observe that for the object to be detected the error is low for the foreground and high for the background.

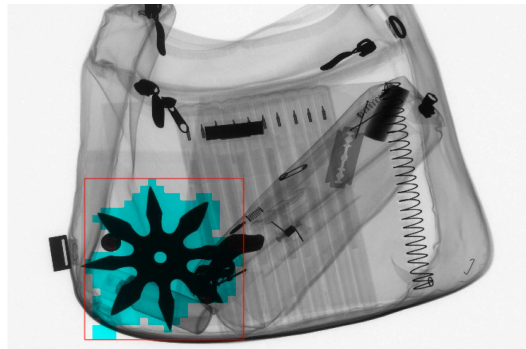

Figure 13. Setection of a shuriken using proposed algorithm 1.

Figure 14. Setection of a razor blade using proposed algorithm 1.

In order to compare our method with other methods that can be used for this task, we followed the evaluation protocol proposed in [14] for the detection of handguns in cropped X-ray images. In this experiment, there are two classes: target and no-target, *i.e.*, handguns and no-handguns. In no-handguns class there are razor blades, shuriken and other objects like pens, clips, etc. The number of target/no-target X-ray images of the sets of training (used to design the detector), validation (used to tune the detectors' parameters) and testing (used to measure the performance) are: 200/700, 50/300, 100/600 respectively (see more details in [14]). The obtained results (precision, recall and accuracy) are summarized in Table 1. The first 10 rows of Table 1 show the reported results by [14] in this experiment. The last row shows the results obtained by our method (see 'ours'). As a result, the proposed approach reaches a very good recognition performance, outperforming some alternative state-of- the-art techniques. In order to see the positive effect of the use of the logarithmic images in our model, we repeated this experiment using our method with no-logarithmic images. The results are shown in the row called 'no-log'. We observe that the use of logarithmic images increases the recall from 0.65 to 0.99.

Table 1. Detection of guns

Method	Precision	Recall	Accuracy [%]
BoW	0.65	0.84	91.3
Sparse KNN	0.99	0.97	99.4
Sparse KNN*	0.92	1.00	98.8
AISM	0.97	0.97	99.1
XASR+	0.92	0.88	97.2
GoogleNet	0.83	1.00	97.1
AlexNet	0.85	0.99	97.4
SVM	0.90	0.99	98.3
AdaBoost	1.00	0.87	98.1
SRC	0.75	1.00	95.2
no-log	0.93	0.65	94.3
ours	0.93	0.99	98.7

4.3. Implementation

The experiments were carried out on a iMac OS X 10.12.4 with a 3.7 GHz Quad-Core Intel Xeon E5 processor and 12GB memory (12GB RAM 1866 MHz DDR3). The X-ray images were obtained from $\mathbb{GDX}$ray [12]. The algorithms were implemented in MATLAB[3]. For the sparse representation we used SPAMS library from INRIA [7]. The code of the MATLAB implementation is available on our webpage[4]. For the simulation: Parameters B and C of model (7) were obtained using a calibration approach [10] in which two images of a material with different thickness are acquired. In our case, $B = 0$ and $C = 230$. The simulation of an X-ray image of 6M pixels is performed in 0.09s. For the detection, we give here details of the last experiment (detection of handguns). For the implementation of the dictionaries, we used patches of 180×180 pixels subsampled to 18×18, *i.e.* $w = 180$ and $a = 10$. That means that vector $\mathbf{y}$ has 100 elements. Around 650.000 patches were extracted for the dictionaries. The sparsity (parameter T) was set to 10. The training was performed in 7.45 min, and the detection of threat objects in 65 sec per image (for images of 1M pixel).

5. Conclusions

The main contribution of our paper is a new X-ray imaging model that can separate foreground from background. The model can be used for example in: *i)* simulation of new X-ray images, and *ii)* recognition of objects when foreground and background are appropriately defined. On the one hand, simulated images can be used in training programs for human inspectors, or can be used to enhance datasets for computer vision algorithms. On the other hand, detection algorithms can be employed in automated baggage inspection, because it can be used to aid an user in an inspection task.

Rather than a multiplication of foreground and background, that is typically used in X-ray imaging, we propose the addition of logarithmic images. This allows the use of linear strategies such as sparse representations in order to segment target objects.

In our experiments, we simulate new X-ray images of handguns, shuriken and razor blades, in which it is impossible to distinguish simulated and real X-ray images. On the other hand, we developed an algorithm based on the minimization of the reconstruction error in sparse representations that can be used effectively in the recognition of threat objects. We show in our experiments the detection of shuriken, razor blades and handguns. As a result, the proposed approach reaches a very good recognition performance, outperforming some alternative state-of- the-art techniques. In our experiments, the increase of the performance by including logarithmic images was considerable.

The preliminary results are promising. Nevertheless, since the effectiveness of the proposed model has been verified on a few X-ray images, it is necessary an evaluation on a broader dataset.

[3]Release 2016a from http://ww.mathworks.com

[4]See http://dmery.ing.puc.cl/index.php/material/.

Acknowledgments

This work was supported by Fondecyt Grant No. 1161314 from CONICYT, Chile.

References

[1] G. Blalock, V. Kadiyali, and D. H. Simon. The Impact of Post-9/11 Airport Security Measures on the Demand for Air Travel. *The Journal of Law and Economics*, 50(4):731–755, Nov. 2007.

[2] A. Bolfing, T. Halbherr, and A. Schwaninger. How image based factors and human factors contribute to threat detection performance in X-ray aviation security screening. In A. Holzinger, editor, *HCI and Usability for Education and Work*, volume 5298 of *Lecture Notes in Computer Science*, pages 419–438. Springer Berlin Heidelberg, 2008.

[3] D.Mery, D. Hahn, and N. Hitschfeld. Simulation of defects in aluminium castings using cad models of flaws and real x-ray images. *Insight-Non-Destructive Testing and Condition Monitoring*, 47(10):618–624, 2005. Ron Halmshaw Award 2005.

[4] G. Flitton, T. P. Breckon, and N. Megherbi. A comparison of 3D interest point descriptors with application to airport baggage object detection in complex CT imagery. *Pattern Recognition*, 46(9):2420–2436, Sept. 2013.

[5] G. Flitton, A. Mouton, and T. P. Breckon. Object classification in 3D baggage security computed tomography imagery using visual codebooks. *Pattern Recognition*, 48(8):2489–2499, Aug. 2015.

[6] T. Franzel, U. Schmidt, and S. Roth. Object Detection in Multi-view X-Ray Images. *Pattern Recognition*, 2012.

[7] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online learning for matrix factorization and sparse coding. *J. Mach. Learn. Res.*, 11:19–60, Mar. 2010.

[8] H. Martz, C. Logan, D. Schneberk, and P. Shull. *X-ray Imaging: fundamentals, industrial techniques and applications*. CRC Press, 2017.

[9] N. Megherbi, J. Han, T. P. Breckon, and G. T. Flitton. A comparison of classification approaches for threat detection in CT based baggage screening. In *Image Processing (ICIP), 2012 19th IEEE International Conference on*, pages 3109–3112. IEEE, 2012.

[10] D. Mery. *Computer Vision for X-Ray Testing*. Springer, 2015.

[11] D. Mery. Inspection of Complex Objects Using Multiple-X-Ray Views. *IEEE/ASME Transactions on Mechatronics*, 20(1):338–347, 2015.

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

[13] D. Mery, V. Riffo, I. Zuccar, and C. Pieringer. Automated X-Ray Object Recognition Using an Efficient Search Algorithm in Multiple Views. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE Conference on*, pages 368–374. IEEE Computer Society, 2013.

[14] D. Mery, E. Svec, M. Arias, V. Riffo, J. Saavedra, and S. Banerjee. Modern computer vision techniques for x-ray testing in baggage inspection. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2016. Accepted in October 2016.

[15] S. Michel, S. Koller, J. de Ruiter, R. Moerland, M. Hogervorst, and A. Schwaninger. Computer-based training increases efficiency in X-ray image interpretation by aviation security screeners. In *Security Technology, 2007 41st Annual IEEE International Carnahan Conference on*, pages 201–206, Oct 2007.

[16] A. Mouton and T. P. Breckon. Materials-based 3D segmentation of unknown objects from dual-energy computed tomography imagery in baggage security screening. *Pattern Recognition*, 48(6):1961–1978, June 2015.

[17] A. Mouton, G. T. Flitton, and S. Bizot. An evaluation of image denoising techniques applied to CT baggage screening imagery. In *IEEE International Conference on Industrial Technology (ICIT 2013)*. IEEE, 2013.

[18] E. Parliament. Aviation security with a special focus on security scanners. *European Parliament Resolution (2010/2154(INI))*, pages 1–10, Oct. 2012.

[19] V. Rebuffel, J. Tabary, M. Tartare, A. Brambilla, and L. Verger. SINDBAD: a simulation software tool for multi-energy X-ray imaging. *Proc. 11th European Conference on Non Destructive Testing, Prague*, 2014.

[20] V. Riffo and D. Mery. Active X-ray testing of complex objects. *Insight-Non-Destructive Testing and Condition Monitoring*, 54(1):28–35, 2012.

[21] V. Riffo and D. Mery. Automated detection of threat objects using Adapted Implicit Shape Model. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 46(4):472–482, 2016.

[22] F. Salvat, J. M. Fernández-Varea, and J. Sempau Roma. Penelope-2008: A code system for monte carlo simulation of electron and photon transport. In *Workshop Proceedings, Barcelona, 30 June–3 July 2, 2008*. OECD, 2009.

[23] A. Schumm, P. Duvauchelle, V. Kaftandjian, R. Jaenisch, C. Bellon, J. Tabary, F. Mathy, and S. Legoupil. Modelling of radiographic inspections. In *Nondestructive Testing of Materials and Structures*, pages 697–702. Springer, 2013.

[24] A. Schwaninger, A. Bolfing, T. Halbherr, S. Helman, A. Belyavin, and L. Hay. The impact of image based factors and training on threat detection performance in X-ray screening. In *Proceedings of the 3rd International Conference on Research in Air Transportation, ICRAT 2008*, pages 317–324, 2008.

[25] J. Tabary, P. Hugonnard, and F. Mathy. SINDBAD: a realistic multi-purpose and scalable X-ray simulation tool for NDT applications. In *Int. Symp. on DIR and CT, Lyon*, volume 1, pages 1–10, 2007.

[26] I. Tosic and P. Frossard. Dictionary learning. *Signal Processing Magazine, IEEE*, 28(2):27–38, 2011.

[27] D. Turcsany, A. Mouton, and T. P. Breckon. Improving feature-based object recognition for X-ray baggage security screening using primed visualwords. In *IEEE International Conference on Industrial Technology (ICIT 2013)*, pages 1140–1145, 2013.

[28] M. Yao, P. Duvauchelle, V. Kaftandjian, A. Peterzol-Parmentier, and A. Schumm. X-ray imaging plate performance investigation based on a monte carlo simulation tool. *Spectrochimica Acta Part B: Atomic Spectroscopy*, 103:84–91, 2015.

[29] G. Zentai. X-ray imaging for homeland security. *IEEE International Workshop on Imaging Systems and Techniques (IST 2008)*, pages 1–6, Sept. 2008.

[30] N. Zhang and J. Zhu. A study of X-ray machine image local semantic features extraction model based on bag-of-words for airport security. *International Journal on Smart Sensing and Intelligent Systems*, 8(1):45–64, 2015.