Towards Real-Time Anomaly Detection within X-ray Security Imagery: Self-Supervised Adversarial Training Approach

AKCAY, SAMET

Towards Real-Time Anomaly Detection within X-ray Security Imagery: Self-Supervised Adversarial Training Approach

Samet Akçay

A Thesis presented for the degree of
Doctor of Philosophy

Department of Computer Science
Durham University
United Kingdom
October 2019

Abstract

Automatic threat detection is an increasingly important area in X-ray security imaging since it is critical to aid screening operators to identify concealed threats. Due to the cluttered and occluded nature of X-ray baggage imagery and limited dataset availability, few studies in the literature have systematically evaluated the automated X-ray security screening. This thesis provides an exhaustive evaluation of the use of deep Convolutional Neural Networks (CNN) for the image classification and detection problems posed within the field. The use of transfer learning overcomes the limited availability of the object of interest data examples. A thorough evaluation reveals the superiority of the CNN features over conventional hand-crafted features. Further experimentation also demonstrates the capability of the supervised deep object detection techniques as object localization strategies within cluttered X-ray security imagery. By addressing the limitations of the current X-ray datasets such as annotation and class-imbalance, the thesis subsequently transitions the scope towards deep unsupervised techniques for the detection of anomalies based on the training on normal (benign) X-ray samples only. The proposed anomaly detection models within the thesis employ a conditional encoder-decoder generative adversarial network that jointly learns the generation of high-dimensional image space and the inference of latent space — minimizing the distance between these images and the latent vectors during training aids in learning the data distribution for the normal samples. As a result, a larger distance metric from this learned data distribution at inference time is indicative of an outlier from that distribution — *an anomaly*. Experimentation over several benchmark datasets, from varying domains, shows the model efficacy and superiority over previous state-of-the-art approaches. Based on the current approaches and open problems in deep learning, the thesis finally provides discussion and future directions for X-ray security imagery.

Declaration

The work in this thesis is based on research carried out at the Department of Computer Science, Durham University, United Kingdom. No part of this thesis has been submitted elsewhere for any other degree or qualification and it is all my own work unless referenced to the contrary in the text.

Acknowledgements

Conducting this PhD has been a rather challenging experience, which would not be possible for me to finish without the support I received from numerous people.

First of all, I would like to express my sincere gratitude to Professor Toby Breckon for being not only an advisor but also a great mentor. Thanks to his friendly approach, I was able to get his support whenever I needed. Within a great team, together with his knowledge and motivational support, I have had a rather productive PhD, leading to publications with high quantity and quality. I could not have imagined having a better advisor and mentor for my PhD study.

Besides, I would also like to thank the thesis committee Dr George Alex Koulieris and Professor Lewis Griffin, for taking their valuable time to examine this PhD work.

I am also very grateful to my fellow labmates for the endless discussions and for the great time when we were pushing the deadlines. More specifically, I would like to thank Amir Atapour-Abarghouei for the insightful discussions for the algorithms we design. I am also grateful to Gregoire Payen de La Garanderie for helping me resolve technical issues.

I would also like to say thank you to my sisters, Mum and Dad, for always believing in me and showing their support even from $2,833$ miles away. I very much appreciate my Dad's help, who has been a great mentor for me throughout my entire life.

Another thank you goes to my lovely cat, Obur, for accompanying me during my sleepless nights to conduct the experiments and to meet paper deadlines. He was always there when needed.

Last but definitely not least, I would like to thank my wife, Cangül, for her great support within the every second of this journey. Despite conducting her own PhD, raising a 2-year old son as well as a newborn, she tried to motivate me every second. This PhD study would not be complete without her support. I would also like to thank my 2-year old son, Kerem Bera and the newborn, Erdem Uraz for being such lovely boys during this challenging research and writeup period.

Contents

List of Figures

List of Tables

Dedication

to Kerem Bera & Erdem Uraz

CHAPTER 1

Introduction

X-ray security screening is widely used to maintain aviation and transport security. It poses a significant image-based screening task for human operators reviewing compact, cluttered and highly varying baggage contents within limited time-scales. The increased passenger throughput, in the global travel network, and the increased focus on broader aspects of extended border security (e.g., freight, shipping, postal) results in a challenging automated image classification task.

A considerable amount of literature has been published on X-ray security systems [17, 18]. With a great deal of the previous research within the field has focused on screening systems [19–23]. It was, however, not until the early 2010s that computer-aided X-ray security screening, in terms of prohibited item and threat object detection, attracted sufficient scholarly attention [17, 18]. Despite the study of automated X-ray security imaging gaining momentum recently, there are still significant research challenges remaining for both the detection of threat objects and also anomalous occurrences within such cluttered and complex X-ray security imagery.

This thesis aims to contribute to this growing area of research by exploring both supervised and unsupervised learning algorithms to design state-of-the-art object detection systems that can operate in real-time.

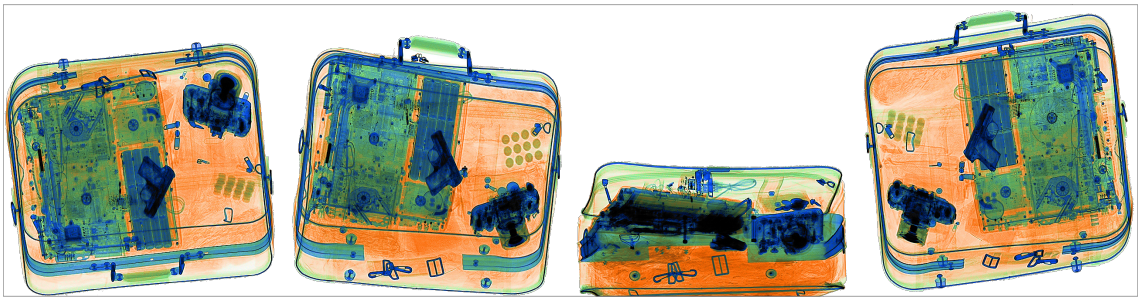


Figure 1.1: Exemplary X-ray baggage image from multiple-views.

The first part of the thesis focuses on deep supervised learning techniques to classify and localize the threat objects from X-ray images (Chapter 3). The rest of the thesis, on the other hand, concentrates more on unsupervised learning approaches, due to highly imbalanced X-ray datasets (Chapter 4 and 5).

1.1 Motivation

X-ray security screening is one of the most widely used security measures to ensure the airport and transport security. Human operators play a vital role to screen thousands of bags each day, a non-trivial task in terms of assured threat detection accuracy. In manual X-ray security imagery by human operators, experience and knowledge are critical to effectively overcoming difficulties during testing [24]. Even though experienced and more knowledgeable screeners are more confident than novice screeners to assess X-ray security imagery bags, they both need diagnostic aid for challenging cases [25]. Besides, although training and experience could improve screener knowledge and skills, their actual job performance highly depends on external factors such as emotional exhaustion and job satisfaction [26–28]. Besides, measures used to evaluate the visual inspection performance of the screeners do not always reflect the actual performance [29]. Computer-aided systems such as automated machine learning algorithms showing the exact location of the prohibited items boost operator detection performance and response time as well as higher operator trust [30].

The complex and cluttered nature of X-ray security imagery makes threat detection a challenging task, and adversely impact human operator decision time and

detection performance [31–33]. For instance, the threat detection performance of human screeners significantly reduces when laptops are inside the bags since the compact structure of laptops conceals potential threats [34, 35]. All of these issues readily invite the potential for the use of automated object detection algorithms within X-ray security screening.

Recently, there has been a surge of interest in X-ray screening systems [19–23]. However, automated computer-aided X-ray security image screening is understudied, particularly due to the lack of data resources, and the need for advanced learning algorithms to solve the cluttered nature of the task. State-of-the-art studies within the literature have focused on image enhancement [36–38], segmentation [**?**, 39, 40], classification [41–44], detection [42, 45–47] and unsupervised anomaly detection [48–51] tasks in order to further investigate the real-time applicability of these systems to automate the X-ray security imagery.

Prior surveys such as those conducted by Rogers *et al.* [18] and Mouton & Breckon [17] categorize the existing literature within two main categories (i) image processing and (ii) image understanding. Pioneering work within the field focuses more on image processing approaches such as image manipulation, image enhancement, threat image projection (TIP), material discrimination and segmentation. Recent work, on the other hand, has a particular interest in image understanding focusing more on automated threat detection and automated content verification by using machine learning algorithms.

Traditionally, a machine learning algorithm pipeline contains pre-processing, enhancement, segmentation, feature extraction, and classification stages. Pre-processing and enhancement stages clean the input data and improve the overall quality of the images. The segmentation step crops the regions of interests from the full cluttered image. Feature extraction stage extracts the hand-crafted features of the object, such as edges and shapes. The final stage classifies the images based on the features derived from the preceding step.

The main drawback of these machine learning approaches is their dependency on hand-crafted features requiring manual engineering. Deep neural networks overcome this issue by learning the features that are specific to the problem domain, which

overall yields a significant improvement over the previous approaches [52]. A neural network contains a single or multiple layers, each of which comprises a set of neurons activations and non-linear transformation. The earlier layers learn high-level features such as edges and shapes, while higher layers learn lower level features that are more specific to the image fed into the network [53]. LeCun *et al.* [54] is considered to be one of the first successful implementations of a neural network, where the model classifies the hand-written digits. After AlexNet, proposed by Krizhevsky *et al.* [7] that won the 2012 ImageNet object classification challenge [55] by a large margin, deep neural networks have become the golden standard approach within automatic scene understanding.

Within the X-ray security imaging, on the other hand, the transition from the classical machine learning to modern deep learning approaches has not been instant. This is due to extensive data-driven requirements of deep learning approaches, which limits its use within the field, where the availability of such large datasets is significantly limited due to the specialist and security-related nature of the topic. After transfer learning paradigm is introduced into the field [56], which enables to train deep models on small datasets by transferring the learned model weights from larger datasets, the use of deep learning approaches has become feasible to use in X-ray security imaging [9, 57, 58]. Despite the growing interest and various proposed approaches, X-ray security imaging is still understudied compared to more general object recognition literature or applications such as perception for autonomous road vehicles [59]. The work presented in this thesis goes some way to address these issues representing both advances in supervised and unsupervised deep machine learning for the X-ray security image understanding context.

1.2 Thesis Contributions

The main contributions of the thesis are as follows:

- An exhaustive evaluation of conventional hand-crafted features and contemporary end-to-end and feature-space deep learning training for the classification and detection tasks. (Chapter 3)

- An unsupervised anomaly detection algorithm based on a novel adversarial autoencoder within an encoder-decoder-encoder pipeline, capturing the training data distribution within both image and latent vector space, yielding both statistically and computationally superior results to contemporary GAN-based [13, 14] and traditional autoencoder-based approaches [60] (Chapter 4).

- A variant unsupervised anomaly detection model, over a skip-connected encoder-decoder convolutional network architecture and a multi-task discriminator network, which addresses the high-reconstruction error and redundant network parametrization issues of the previous work, yielding superior reconstruction within the image and latent vector spaces (Chapter 5).

- An extensive overview of classical machine learning approaches and contemporary deep learning algorithms within X-ray security imaging.

1.3 Publications

The work contained within this thesis has been previously published in the following peer-review publications by the author, and is used in the chapters as indicated below:

- **Transfer Learning using Convolutional Neural Networks for Object Classification within X-Ray Baggage Security Imagery**, S. Akçay, M. E. Kundegorski, M. Devereux, T.P. Breckon, In Proceedings of the International Conference on Image Processing, IEEE, 2016, pp. 1057-1061. (Contributing to Chapter 3)

- **An Evaluation of Region-Based Object Detection Strategies within X-ray Baggage Security Imagery** ,S. Akçay, TP. Breckon, In Proceedings of the International Conference on Image Processing, IEEE, 2017, pp. 1337-1341. (Contributing to Chapter 3)

- **Using Deep Convolutional Neural Networks for Automated Object Detection and Classification within X-ray Baggage Security Imagery**, S. Akçay, ME. Kundegorski, CG.Willcocks, TP. Breckon, Transactions

on Information Forensics and Security, IEEE, 2018, pp. 2203-2215. (Contributing to Chapter 3)

- **GANomaly: Semi-Supervised Anomaly Detection via Adversarial Training**, S. Akçay, A.Atapour-Abarghouei, T.P. Breckon, In Proceedings of the Asian Conference on Computer Vision ACCV, Lecture Notes in Computer Science, Springer, 2018, pp. 622-637. (Contributing to Chapter 4)

- **Skip-GANomaly: Skip Connected, and Adversarially Trained Encoder-Decoder Anomaly Detection**, S. Akçay, A. Atapour-Abarghouei, T.P. Breckon, In Proceedings of the International Joint Conference on Neural Networks IJCNN, IEEE, 2019, pp. 1-8. (Contributing to Chapter 5)

1.4 Thesis Scope and Structure

This thesis presents a number of topics spanning both supervised and unsupervised deep machine learning. Chapter 2 thoroughly reviews the X-ray security imaging literature by categorizing the field based on the machine and deep learning-related algorithms. In this chapter, machine learning algorithms are divided into image enhancement, threat image projection, object segmentation, feature extraction, and object classification algorithms. Likewise, deep learning-based algorithms are reviewed based on object classification, detection, and segmentation-based algorithms.

Chapter 3 explores the use of deep Convolutional Neural Networks (CNN) by comparing the conventional state-of-the-art Bag of Visual Words (BoVW) approach. Employing a transfer learning paradigm such that a pre-trained CNN, primarily trained for generalized image classification tasks where sufficient training data exists, can be optimized explicitly as a later secondary process towards this application domain demonstrates the superiority of the CNN against BoVW. The chapter further explores the use of object detection algorithms within the X-ray security context and show promising results within the field.

Chapter 4 addresses the class imbalance issue within X-ray security imaging and proposes an unsupervised anomaly detection algorithm that utilizes encoder-

decoder-encoder networks with adversarial training, and that outperforms the previous state-of-the-art anomaly detection algorithms.

Chapter 5 also explores the unsupervised anomaly detection problem within X-ray security imaging, and proposes a more straightforward pipeline that utilizes more advanced network architectures, which improves the anomaly detection performance even further.

Chapter 5 further investigates the high-reconstruction error and parameter redundancy issues of the work presented in Chapter 4, and introduces an unsupervised anomaly detection algorithm by employing an adversarial training with skip-connected generator and multi-tasked discriminator networks, which overall outperforms the previous work.

Chapter 6, the final chapter, draws together the key findings presented within the previous chapters and provides a discussion based on the strengths and the weaknesses of the proposed algorithms together with future directions within the field.

CHAPTER 2

Literature Review

2.1 Introduction

This chapter reviews the current X-ray security screening literature by taxonomising the field into conventional image analysis, traditional machine learning and contemporary deep learning approaches. Conventional image analysis section reviews the early attempts of image enhancement and threat image projection techniques studied in the field. Similarly, traditional machine learning section reviews the literature based on the techniques proposed for classification, detection and segmentation tasks. Finally, for the deep learning-based approaches, the chapter reviews the available X-ray datasets, classification, detection, anomaly detection and segmentation-based models. The chapter is concluded by the discussion of the current and future challenges within the field.

Despite the surge of interest in X-ray screening [19–23], automated computer-aided screening is understudied, particularly due to the lack of data, and the need for advanced learning algorithms. State-of-the-art studies within the literature have focused on image enhancement [36–38], classification [41–44], detection [42, 45–47], segmentation [**?**, 39, 40], and unsupervised anomaly detection [48–51] for automated

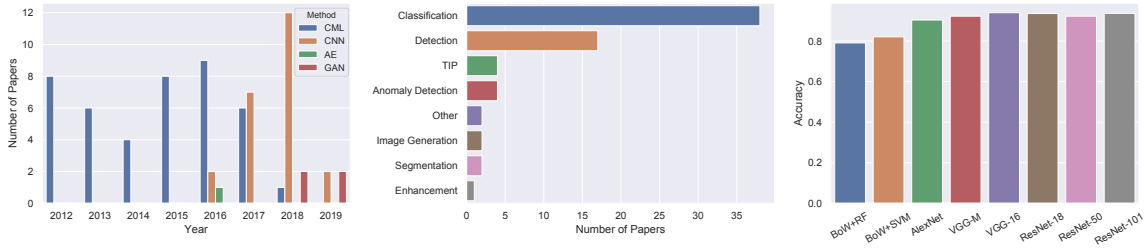


Figure 2.1: Statistics for the recent papers published in X-ray security imaging. Conventional Machine Learning (CML) approaches were dominant in the field before 2016, while deep learning approaches have recently become the standard approach.

security screening. Notable surveys within the field [17, 18] categorize the existing literature within two main categories: (i) image processing [36] and (ii) image understanding [40, 57, 61]. Pioneering work within the field focuses more on image processing approaches such as image enhancement [36], threat image projection (TIP) [62], material discrimination and segmentation [38]. Recent work, on the other hand, has a particular interest in image understanding focusing more on automated threat detection and automated content verification via machine/deep learning algorithms [**?**, 43, 46, 50, 51].

In a traditional setting, a machine learning algorithm pipeline contains pre-processing, enhancement, segmentation, feature extraction, and classification stages [17, 18]. Pre-processing and enhancement stages reduce the noise from the input data and improve the overall quality. The segmentation step crops the regions of interests from the full cluttered image. Feature extraction stage extracts the hand-crafted features of the object, such as edges and shapes. The final stage classifies the objects based on the features derived from the preceding step.

The main drawback of these machine learning approaches is their dependency on hand-crafted features requiring manual engineering. Deep convolutional neural networks overcome this issue by learning the task-specific features, which overall yields a significant improvement. A convolutional neural network contains a single or multiple layers, each of which comprises a set of neuron activations and non-linear transformation. The earlier layers learn high-level features such as edges and shapes, while higher layers learn lower level features that are more specific to the image fed into the network. Despite being initially proposed more than decades ago [54], the use of convolutional neural networks within the field of computer vision has become

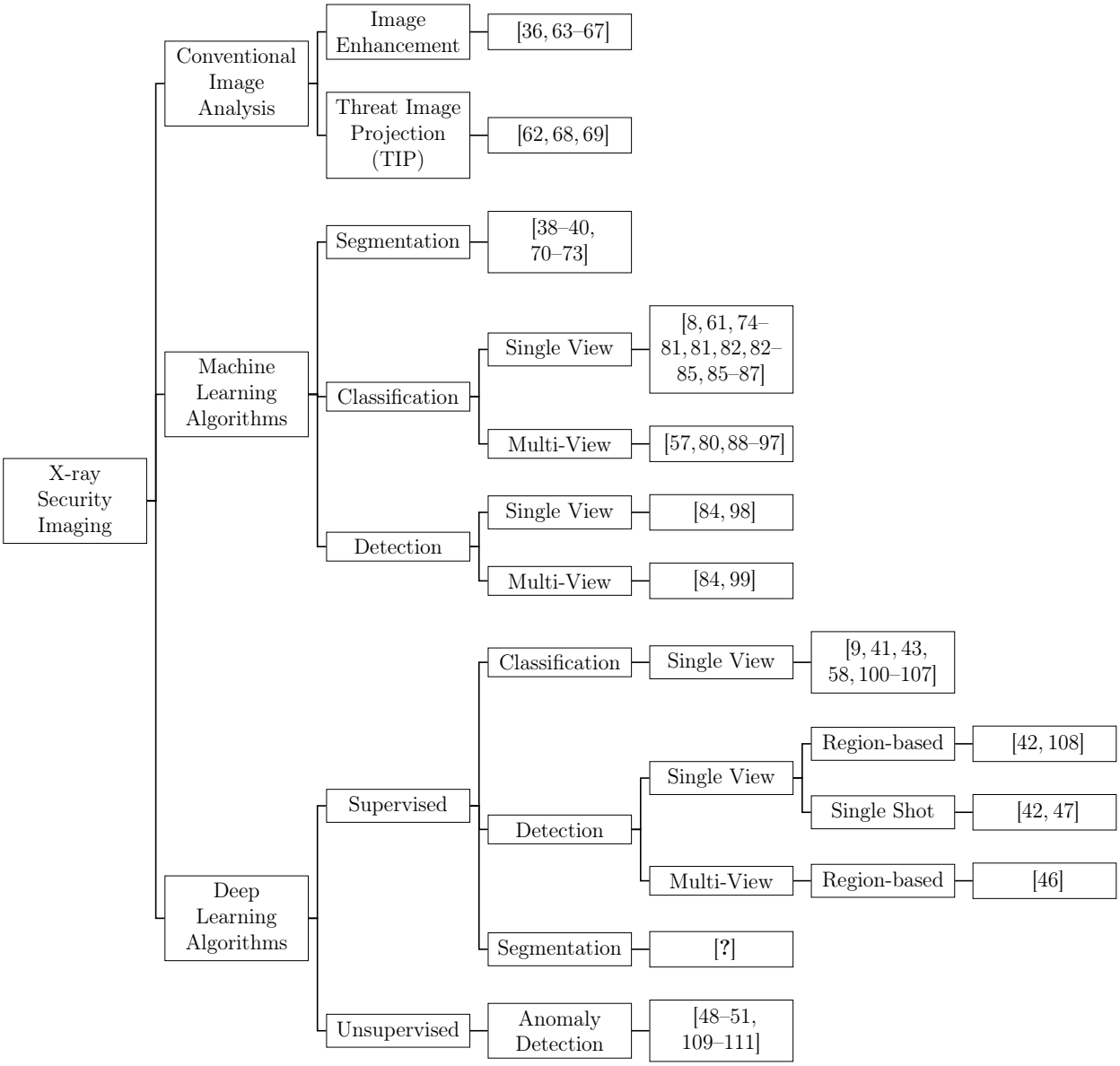


Figure 2.2: A Taxonomy of the X-ray security imaging papers.

prevalent especially after achieving state-of-the-art performance [7] on ImageNet object classification challenge [55] by a large margin.

Within the X-ray security imaging, on the other hand, the transition from the classical machine learning to modern deep learning approaches was not instant. This is due to data-hunger nature of deep learning approaches, which initially limited its use within the field, where the availability of such large datasets is somewhat limited. With the utilisation of transfer learning paradigm [56] and synthetic data generation [62], the use of deep learning approaches has become the general approach within the field [9, 57, 58].

This literature survey reviews the published work within various computer vision tasks (Figure 2.1b) in X-ray security screening, with a particular focus on the deep learning applications. The main contributions of this thesis are as follows:

- *taxonomy* — an extensive overview of classical machine learning and contemporary deep learning within X-ray security imaging.
- *datasets* — an overview of the large datasets used to train deep learning approaches within the field.
- *open problems* — discussion of the open problems, current challenges, and future directions based on the current trends within computer vision.

The rest of the chapter is as follows: Sections 2.2 and 2.3 explore conventional image analysis and machine learning algorithms with a specific focus on image enhancement, threat image projection, image segmentation, object classification, and object detection. Section 2.4 reviews the applications of the deep learning algorithms within X-ray security imaging. Section 2.5 finally concludes the chapter.

2.2 Conventional Image Analysis

A conventional image understanding consists of the following stages: (i) pre-processing stage that enhances the quality of the input image, (ii) segmentation stage to crop the region of interests (RoI) from the full image, (iii) feature extraction stage that computes fundamental attributes of the object such as edges, texture and shape, (iv) classification stage to predict the corresponding class label based on the extracted features. This section explores the conventional image analysis techniques that perform image enhancement and threat image projection.

2.2.1 Image Enhancement

Pre-processing the input data plays a substantial role to yield higher-quality images that increase the readability by both screener and computer.

An image enhancement algorithm presented in [36] comprises three stages: A wavelet transformation to fuses high and low energy X-ray images, (ii) background subtraction via histograms to reduce the fusion noise and (iii) histogram-based image enhancement. Qualitative experimentation depicts superior image output than that of the original X-ray images.

An adaptive enhancement algorithm [64] utilizes a multi-layer perceptron to predict the most optimum technique for the input image. The model takes an input of varying viewability measures, a measurement for the images before and after the enhancement operation, and outputs the most suitable enhancement algorithm. Adaptively predicting the optimum enhancement is shown to outperforms the use of fixed enhancement methods.

Another image enhancement algorithm [63] declutters and clusters RoI from complex X-ray images by an optimum threshold selection, achieved by Radon Transform. In addition to being efficient, the algorithm yields up to 170% general threat detection improvement compared to the original raw X-ray images and improves low-density threat detection of human operators by 58%.

The seminal works of [65] and [36] investigate the use of pseudocolouring techniques within X-ray baggage imagery. Application of pseudocolouring to grey-scale X-ray images improves the detection performance of human operators from 40% to 97% as well as their alertness level. Another set of experimentation reveals that HSI-based colour mapping techniques are more suited for human perception and alertness. A similar work [112] enhances threat detection performance within X-ray imagery by proposing a new colour coding scheme, calibrating the estimation of effective atomic number (Z_{eff}) and density information (ρ).

Wobble effect is a severe issue for the X-ray images produced by mobile scanners. To address this issue, Rogers *et al.* [66] first quantify the wobble error via the use of root mean square (RMS) deviations of mobile and static images. The second step estimates the position of the X-ray beam based on the ground truth, finally fused with the calculated deviations. A follow-up work [67] proposes a wobble estimator via Random Regression Forest (RRF) [113], which estimates/calibrates the sensor activities and corrects the images with wobble artefacts. Experiments report 87%

improvement on image error that stems from the wobble effect.

2.2.2 Threat Image Projection (TIP)

The detection performance of human screeners is heavily dependent on experience and knowledge acquired with computer-based training [26, 114, 115]. Due to the limited availability of X-ray scans with prohibited items, the training is achieved with the images onto which threat images are synthetically projected (Threat Image Projection (TIP) [62]).

More recently TIP has also been used for synthetic data generation to address the data requirements of machine learning models. By projecting a large number of threat objects onto benign X-ray images, it is possible to gather large datasets that could train/evaluate machine learning algorithms [68, 69].

One of the recent TIP implementations [68] first removes the background from a threat patch, yielding a binary threat mask. Projection of the binary threat mask onto the input X-ray image via multiplication finally yields the output X-ray image with the threat item. To provide robust training to machine learning algorithms with diverse and realistic image samples, the algorithm utilizes affine transformations during the projection.

Another TIP study [69] employs logarithmic transformations to separate foreground objects from the background. Subsequently, the threat objects are projected via multiplication operation since it is empirically shown to achieve superior projection than that of addition. Another use of the algorithm is the task of object detection, where a sparse representation algorithm extracts the dictionaries of both foreground (threat) and background (benign) objects and performs classification, which yields 93.0%, 99.0%, 98.7% precision, recall, and accuracy, respectively.

2.3 Machine Learning Approaches in X-ray Security Imaging

This section explores the applications of conventional machine learning approaches in X-ray security imaging. The literature is reviewed based on three tasks: (i) classification, (ii) detection, and (iii) segmentation. For an alternative perspective for this section, the reader could refer to the related reviews of Mery [116] and Rogers *et al.* [18].

2.3.1 Object Classification

Prior to the dominance of the deep learning within the field, the bag of visual words (BoVW) approach was prevalent. In of the initial attempts utilising BoVW, Başţan *et al.* [77] perform classification of X-ray objects on a relatively limited dataset. Scale Invariant Feature Transform (SIFT) [117], Speeded Up Robust Features (SURF) [118] and Binary Robust Independent Elementary Features (BRiEF) feature descriptors are computed around the points detected using standard Difference of Gaussians (DoG), Hessian Laplace, Harris, Features from accelerated segment test (FAST) and STAR feature detectors. k-means [119] clusters the visual vocabulary, which is trained with an SVM [120]. DoG detector and SIFT descriptor are shown to perform the best among the descriptors (mAP: 0.65 on 200 X-ray images.

Inspired by [77], Turcsany *et al.* [78] presents a unique BoWV approach for X-ray firearm classification via class-specific feature extraction. With the use of SURF [118] feature detector and descriptor with a BoVW approach trained on an SVM [120] classifier achieves 99.07% true positive rate and 4.31% false-positive rate.

A multi-staged approach [82] performs car detection from X-ray images of freight containers. The method first creates *cars* vs *non-cars* image patches from stream-of-commerce X-ray images. The next step extracts features via image intensity, log intensity together with basic and oriented images features [121]. The final stage utilizes Random Forest (RF) [113], achieving 100% detection rate with 1.23% false alarm rate. A follow-up work [85] detects loads in cargo containers by an RF classifier

trained with local image moments and oriented basic image features (oBIF) [121], yielding 99.3 % detection accuracy and 0.7% false positives.

BoVW approach is further employed in [61]. A dictionary is formed for each class that consists of SIFT [117] feature descriptors of randomly cropped image patches. Fitting a sparse representation classification to the feature descriptors of the random test patches yields 95% accuracy for each class and 85% in case of occlusion. In another BoVW approach in [86, 87], an SVM is trained with local latent low-level image features extracted from a dataset with 15 different classes, each of which comprises 100 images (AUC: 80.1%).

Inspired by the various research outcome drawn for the BoVW, Kundegorski *et al.* [8] exhaustively evaluate various feature point descriptors within a BoVW-based image classification task. The combination of FAST-SURF trained with an SVM classifier [120] is the best performing feature detector and descriptor combination for a firearm detection task on a large dataset, yielding a statistical accuracy of 0.94 (true positive: 83% and false positive: 3.3%).

Despite the BoVW dominance, other computer vision/machine learning techniques have also been studied for X-ray object classification task—a study [80] aims at detecting threat items in vehicles using X-ray cargo imagery. The proposed multi-staged approach (i) initially improves the image quality via normalisation, denoising, and enhancement, (ii) subsequently performs multi-view alignment and pseudocolouring (iii) finally classifies the threats via correlating the similarities between temporally aligned images. Another study by Zhang *et al.* [81] investigate the use of joint shape and texture features extracted from superpixel regions of the input. Training the extracted feature-map with SVM [120] yields 89% classification accuracy.

Mery *et al.* [90] utilize structure estimation and segmentation together with a general tracking algorithm to detect X-ray objects. Another classification pipeline by Mery *et al.* [90] (i) extracts features with SIFT [117], (ii) removes redundancy via RANSAC [122], (iii) sort features based on the difference between two consecutive frames and (iv) use Mahalanobis distance classifier to predict class labels (P: 70%, R: 86% 64 X-ray images).

Similar works [57,95,96,100] exhaustively evaluate various computer vision techniques, with a specific focus on k-nn based sparse representation. A k-means algorithm [119] clusters the features, segmented from input via an adaptive k-means [123] and extracted via SIFT [117]. During the test, the score for a patch is calculated based on the closest distance to a neighbour clustered via k-nn classifier [124], achieving comparable accuracy to deep models on GDXray (94.7% vs. 96.3%).

2.3.2 Object Detection

This section reviews the conventional X-ray object detection models presented in the literature. Being a challenging task, where the bounding box coordinates and class labels are to be predicted simultaneously, conventional object detection algorithms in the literature is relatively limited in the field.

Schmidt-Hackenberg *et al.* [98] compare the use of visual cortex inspired features such as SLF-HMAX and v1-like to the standard features such as SIFT [117]. Compared to SIFT, HMAX features are shown to provide superior feature encoding for BoVW approach trained with SVM [120].

Evaluation works of [79, 84] exhaustively investigate the use of BoVW for X-ray object detection. Evaluating various feature descriptors within a single and multiple-view imagery for the detection via branch and bound algorithm with structural SVM classifier [120] shows that (i) combination of SIFT and SPIN achieves the best detection performance (mAP: 46.1%), and (ii) utilising multi-view improves the detection (mAP: 66.5%).

Multi-view X-ray imaging improves the performance when rotation and superimposition hinder the viewability of the objects from one view [125]. Despite its computational complexity, multi-view imaging help human operators and machines to improve the detection performance [84, 126].

A general multi-staged approach proposed in the works of [89, 91, 93, 127] (i) initially performs feature extraction via feature descriptors and k-NN classifier [124], (ii) matches the key-points for the consecutive images from different views and (iii) analyse the multiple-views where the key-points of the two successive images are matched, and their 3D points are formed with structure from motion. After being

clustered, 3D points are re-projected back to 2D key-points, which are classified by the k-NN classifier [124]. The best performing approach achieves 95.7% precision, 92.5% recall for 120 X-ray images.

Franzel *et al.* [99] propose a sliding window detection approach with the use of a linear SVM classifier [120] and histogram of oriented gradients (HOG) [128]. As HOG is not fully rotationally invariant, they supplement their approach by detection of varying orientations. Multi-view integration step fuses the multiple viewpoints to find the intersection of the true detections, which achieves superior performance compared to single-view (mAP: 0.645).

2.3.3 Object Segmentation

One of the crucial steps for accurate object classification in conventional image understanding is the precise object segmentation. The rest of the section explores various segmentation techniques presented in the literature.

Early work in the field [70, 71] investigates simplistic pixel-based segmentation with a fixed absolute threshold and region grouping. Sing *et al.* [39] optimize segmentation parameters to accurately separate cluttered baggage objects. The model extracts features via complexity estimate, average edge gradient and colour purity to fine-tune the parameters of a Gaussian Mixture Model (GMM) [129] for segmentation. Mapping the features and GMM with a three hidden layer neural network achieves 88.2% accuracy.

The segmentation algorithm by Ding *et al.* [72] segments X-ray objects within three steps: (i) pre-segmentation stage groups nearest neighbour pixels based on colour texture match, (ii) attributed regional graphs (ARG) represents the objects in the image, (iii) the fuzzy similarity distance between ARG images yields the number of layers within the object, segmenting the overlapping regions.

Lu and Conners' [38] three-stage segmentation approach (i) segments objects by removing the noise and by determining the ROI, (ii) removes the overlapping background to accurately compute grey-level and (iii) computes R and L values that yield the information to detect threats.

Instead of using shape information, Heitz and Chechik [40] estimate chemical

(attenuation) properties of the objects for the segmentation. The utilisation of multiple viewpoints improves the estimation accuracy further. Empirical evaluation demonstrates the method's superiority to standard image segmentation approaches (RMS error: 1.15 on 23 X-ray images).

On the assumption that threat items are mostly metallic objects with high atomic numbers (Z_{eff}) values, Kechagias-Stamatis *et al.* [73] utilizes Z_{eff} to detect threat materials. The authors subsequently make use of morphological operations to disconnect the regions of interests and noise filtering for image enhancement. Next, soft clustering reduces the artefacts of the segmented regions and hard clustering to cluster overlapping objects. Finally, the use of Nearest Neighbor Distance Ratio [130] matches SURF [117] key-point detector and descriptors, and RANSAC [122] refines the paired features.

2.4 Deep Learning in X-ray Security Imaging

This section reviews the X-ray security applications utilising deep learning algorithms. By initially introducing the well-established datasets in the field, we explore the applications for various computer vision tasks such as object classification, detection, segmentation and unsupervised anomaly detection.

2.4.1 Datasets

This section explores X-ray security imaging datasets that are widely used in the literature.

Durham Baggage Patch/Full Image Dataset

This dataset comprises 15449 X-ray samples with associated false color materials mapping from dual-energy. Originally, samples have the following class distributions: 494 *camera*, 1596 *ceramic knife*, 3,208 *knife*, 3192 *firearms*, 1203 *firearm parts*, 2390 *laptop* and 3366 *benign* images. Several variants of this dataset is constructed for classification (DBP2 and DBP6) [8,9,42] and detection (DBF2 and DBF6) [42,108].

GDXray

Grima X-ray Dataset (GDXRAY) [131] comprises 19407 X-ray samples from five various subsets including castings (2727), welds (88), baggage (8150), natural images (8290), and settings (152).

The baggage subset is mainly used for security applications and comprises images from multiple-views. The limitation of this dataset is its non-complex content, which is non-ideal to train for real-time deployment.

UCL TIP

This dataset comprises $120,000$ benign images, each of which is 16-bit grayscale with sizes varying between 1920×850 and 2570×850. The train and test split of the dataset is $110000 : 10000$, where the training images are 256×256 patches randomly sub-sampled from 110000 images and the test set comprises 5000 benign and 5000 threat images. The threat images are synthetically generated via the TIP algorithm proposed in [68], where, depending on the application, small metallic threats (SMT) or car images are projected into the benign samples. With several variants, this dataset is used in several studies such as [58, 101–104, 110, 111].

SIXray

Collected and released by [43], SIXray dataset comprises $1,059,231$ X-ray images, 8929 of which are manually annotated for 6 different classes: gun, knife, wrench, pliers, scissors, hammer, and background. The dataset consists of objects with a wide variety in scale, viewpoint and especially overlapping, and is first studied in [43] for classification and localization problems.

Durham Baggage Anomaly Dataset –DBA

This in-house dataset comprises $230,275$ dual energy X-ray security image patches extracted via a 64×64 overlapping sliding window approach. The dataset contains 3 abnormal sub-classes —*knife* $(63,496)$, *gun* $(45,855)$ and *gun component* $(13,452)$. Normal class comprises $107,472$ benign X-ray patches, split via $80 : 20$ train-test

ratio. DBA dataset is used in [49] and [50] for unsupervised anomaly detection.

Full firearm vs Operational Benign –FFOB

As presented in [42, 49, 50], this dataset contains samples from the UK government evaluation dataset [15], comprising both expertly concealed firearm (threat) items and operational benign (non-threat) imagery from commercial X-ray security screening operations (baggage/parcels). Denoted as FFOB, this dataset comprises $4,680$ firearm full-weapons as full abnormal and $67,672$ operational benign as full normal images, respectively.

2.4.2 Evaluation Criteria

Before listing the performance results of the reviewed papers, it is important to introduce the various performance metrics used in the field.

Accuracy (ACC) Accuracy is defined as the number of correctly predicted samples over the total number of predictions, which is mathematically shown as $ACC = (TP + TN)/(TP + TN + FP + FN)$.

Mean Average Precision (mAP) mAP is defined as the mean of the average precision, a metric evaluated by the area under the precision and recall curve, where precision is $TP/(TP + FP)$, and recall is $TP/(FN + TP)$.

Dataset	Domain	Task	# Samples	Classes	Performance	Reference
DBP2	Baggage	Classification	19,938	firearm, background	ACC: 0.994	[9, 42]
DBP6	Baggage	Classification	10,137	firearm, firearm parts, camera, knife, ceramic knife, laptop	ACC: 0.937	[9, 42]
UCL TIP	Cargo	Classification Detection Anomaly Detection	120,000	small metallic threat (SMT), car	ACC: 0.970	[41, 102–104, 110, 111]
GDXRay	Baggage	Classification Detection	19,407	gun, shuriken, razor blade	ACC: 0.963	[57, 107, 132, 133]
DBF2	Baggage	Detection	15,449	firearm, background	mAP: 0.974	[42, 108]
DBF6	Baggage	Detection	15,449	firearm, firearm parts, camera, knife, ceramic knife, laptop	mAP: 0.885	[42, 108]
PBOD	Baggage	Classification	9,520	Explosives	AUC: 0.950	[134]
MV-Xray	Baggage	Detection	16,724	Glass Bottle, TIP Weapon, Real Weapon	mAP: 0.956	[46]
SASC	Baggage	Detection	3,250	Scissors, Aerosols	mAP: 0.945	[47]
Zhao *et al.*	Baggage	Classification	1,600	wrench, pliers, blade, lighter, knife, screwdriver, hammer	ACC: 0.992	[106]
Smiths-Duke	Baggage	Detection	16,312	gun, pocket knife, mixed sharp	mAP: 0.938	[45]
SIXray	Baggage	Detection	1,059,231	gun, knife, wrench, pliers, scissors, hammer, background	mAP: 0.439	[43]
UBA	Baggage	Anomaly Detection	230,275	gun, gun part, knife	AUC: 0.940	[49, 50]
FFOB	Baggage	Anomaly Detection	72,352	full-weapon, benign	ACC: 0.998	[49, 50]
Yang *et al.*	Baggage	Classification	2,000	wrench, fork, handgun, power bank, lighter, pliers, knife, liquid, umbrella, screwdriver	ACC: 0.991	[44]

Table 2.1: Datasets used in deep learning applications within X-ray security imaging

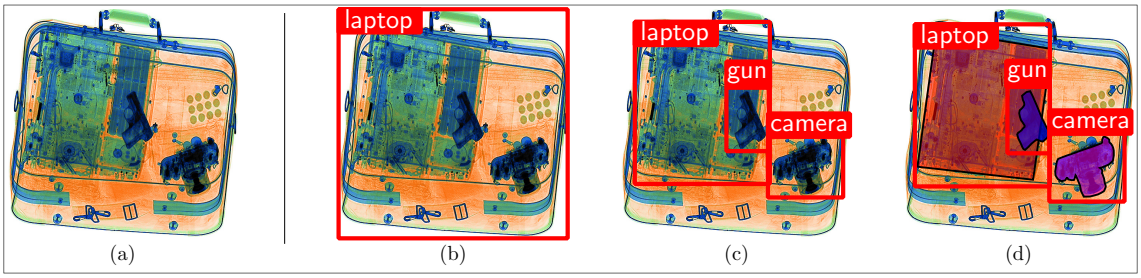


Figure 2.3: An input X-ray image, and the outputs depending on the deep learning task, (a) classification via ResNet-50 [1], (b) detection with YOLOv3 [2] and segmentation via Mask RCNN [3]

AUC AUC is the area under the curve (AUC) of the receiver operating characteristics (ROC), plotted by the true positive rates and false positives rates.

2.4.3 Classification

The study of [9] is one of the first research applying CNN to X-ray security imagery. The authors examine the use of CNN via transfer learning to evaluate to what extent transfer learning helps classify X-ray objects within the problem domain, where the availability of the datasets is somewhat limited. Freezing AlexNet weights layer by layer on a two-class (*gun vs no-gun*) X-ray classification problem shows that CNN significantly outperforms the BoVW approach (SIFT+SURF), trained with SVM or RF, even when the layers of the network are all frozen. Another set of experimentation analyses the use of CNN within a challenging 6-class classification problem, whose results show a great promise of the use of CNN in the field.

A similar work [58] compares the use of deep learning against conventional machine learning to classify non-empty cargo containers with cars or SMT. A multi-stage approach first classifies cargo containers as empty vs non-empty. The second stage is the classification of cars from the containers classified as non-empty, achieved via oBIF + RF. By using UCL TIP dataset, the authors evaluate the of 9 and 19 layers networks [101] that are similar to [7] and [5] and show that even the worst-performing CNN outperforms conventional machine learning approach (oBIF + RF).

A follow-up work [102] further investigates the detection of cars from X-ray cargo images. A sliding window splits UCL TIP images into patches. Authors then explore

various features including intensity, oBIF [121], Pyramid of Histogram of Visual Words (PHOW) [135] and CNN features. Training these features on SVM [120], RF [113], and soft-max (CNN) shows that an RF classifier trained on the VGG-18 [5] features extracted from log-transform images achieves the highest performance (FPR: 0.22%).

Additional work by Jaccard *et al.* [103] evaluate the impact of input types on CNN performance by training single-channel raw image and dual-channel data that contains the raw image and its log-transformed image on VGG [5] variants. The quantitive analysis demonstrates that VGG-19 model trained from scratch by using dual-channel raw and log-transformed images outperforms the other variants (AUC: 97%, FPR: 6%).

Rogers *et al.* [104] explore the use of dual-energy X-ray images for automated threat detection. Authors investigate varying transformations applied to high-energy (H) and low-energy(L) X-ray images captured via the dual-energy X-ray machine. Using UCL TIP dataset, 640,000 image patches are generated via a 256×256 sliding-window. Training this dataset with a fixed VGG-19 network [5] with varying input channels, including single-channel (H), dual-channel($\{H, -\log H\}$, $\{-\log H, -\log L\}$) and four-channels ($\{-\log L, L, H, -\log H\}$) shows that dual and four-channels always achieves superior detection performance compared to their single-channel variants (ACC: 95%–dual vs 90%–single).

Inspired by the limited availability of X-ray datasets, a three-stage algorithm by Zhao *et al.* [106] first classifies and labels the input X-ray dataset via KNN Matting [136] that uses the angle information of the foreground objects extracted from the input image. The second stage generates new X-ray objects via an adversarial network similar to [137]. Additional use of [138] improves the quality of the generated images. Finally, a small classification network confirms whether the generated image belongs to the correct class. In a follow-up study, Yang *et al.* [44] further investigate the ways to improve the GAN training to produce better X-ray images. Experiments and evaluation based on Frechet Inception Distance (fiD) score [139] show that the proposed GAN approach in the paper generates visually superior prohibited items.

Miao *et al.* [43] introduce a model (CHR) to classify/localize X-ray images from

SIXray. The model copes with class imbalance and clutter issue by extracting image features from three consecutive layers, where subsequent layers are upsampled and concatenated with the previous layers. A refinement function $g()$ removes the redundant information from the concatenated feature map. The objective of the work is to minimize the loss of the weighted sum of the classification of the refined mid-level features from the three consecutive layers ($\{h(\tilde{x}_n^{(l-1)}), h(\tilde{x}_n^{(l)}), h(\tilde{x}_n^{(l+1)})\}$). Training the model with the proposed loss yields 2.13% mAP improvement when used with ResNet-101 on SIXray (36.01 vs 38.14).

An evaluation work [134] investigates the use of CNN for the task of explosive detection. An initial stage process the input data by fixing the image size, cropping the irrelevant background object where $Z_{eff} = 0$ and applying data augmentation transformations. Evaluation of random initialization vs pre-training on VGG19 [5], Xception [140], and InceptionV3 [141] networks shows that randomly initialized models achieves superior accuracy for binary classification task. To study the impact of intensity and Z-eff values on the performance, the authors train three VGG-19 networks on both intensity and Z-effective, the intensity only and Z-effective only. Training the model with only Z-eff is shown to yield the highest accuracy. The final set of experiments investigates localization via heatmaps and shows that pre-trained networks achieves superior performance since randomly initialized networks tend to overfit on small datasets.

Caldwell *et al.* [41] study the generalization capability of models trained with different datasets. To investigate this problem, the authors first train a network with a cargo dataset and evaluate its performance with a test set that also contains some parcel dataset samples. Quantitative analysis reveals that the performance of the CNN model is weak when it is tested with the combined dataset. The second stage combines these two datasets within the training stage, yielding considerable improvement in the performance of the model. Based on this experimentation, authors conclude that transferring information between different modalities is challenging since CNN cannot sufficiently generalize to the unseen target dataset.

2.4.4 Detection

After the success of CNN for classification, the work of [108] train sliding-window based CNN, Faster RCNN [10] and R-FCN [142] models on DBF2/6 datasets for firearm and multi-class detection problems. Experiments demonstrate that Faster RCNN [10] with VGG16 [5] yield 88.3% mAP on 6-class DBF6 dataset, while R-FCN with ResNet-101 achieves the highest performance (96.3 mAP) on 2-class (*gun* vs *no-gun*) on DBF2 dataset.

Similar to [108], another evaluation work [45] explores the performance of F-RCNN, R-FCN [142] and SSD [143] within single/multi-view X-ray imagery. Utilizing *OR-gate* detection by merging object detection outputs from individual views shows that multi-view outperforms that of single-view (0.938 vs 0.798 when trained with R-FCN and ResNet-101).

Another work [46] utilises multi-view by modifying Faster-RCNN. A multi-view pooling layer constructs 3D feature 2D extracted from the convolutional layers. 3D region proposal network generates the RoI. Classification and bounding box prediction is performed after 3D RoI pooling layer. Experiments show that multi-view yields an improvement compared to single-view imagery (95.56% vs 91.23%).

Liu *et al.* [47] also performs object detection via YOLOv2 [2] to detect scissors and aeorosols on SASC dataset. Training YOLO v2 for 6000 iterations yield 94.5% average precision and 92.6% recall rates with 68 FPS run-time speed.

Motivated by the lack of annotated X-ray datasets, Xu *et al.* [107], make use of attention mechanisms for the localization of threat materials. The first stage forward-passes an input and finds the corresponding class probability. The back-propagation stage finds which neurons within the network decides the output class. Using the neurons from the first convolutional layer on top of the input image localizes the threat. The final stage refines the activation map by normalizing the layers with the activations of the previous layer. Comparison against the traditional deconvolution method (mAP: 34.3%) shows that the proposed method achieves superior detection (56.6%) without needing for bounding box information.

Similar to [41], generalisation capability of CNN is studied by Gaus *et al.* [144] by training/validating CNN on different datasets (DBF3 (0.88 mAP) $\rightarrow$ SIXray (0.85

mAP)).

2.4.5 Anomaly Detection

Human operators tend to perform better detection when focusing on benign objects rather than threat items. In addition, the knowledge of every-day benign objects leads to much better detection performance [145]. The same concept is applied in anomaly detection, where the model is only trained with normal samples, and tested on normal/abnormal examples.

An anomaly detection approach [48] employs sparse feed-forward autoencoders in an unsupervised manner to learn the feature encoding of normal and abnormal data. An SVM [120] then classifies the images either anomalous or benign. Validation on MNIST [54] and freight container dataset (*empty* vs *non-empty*) shows that hidden layer representation extracted from the autoencoder, in fact, is rather significant for the detection of abnormalities in the images. When fused with the raw-input and residual error, features encoding from the hidden layers yield even better detection performance.

A follow-up work utilizes intensity, log-intensity and VGG-19 [5] features extracted from patches from UCL TIP dataset and train normal images via a forest of random split trees anomaly detector [146]. Testing the model on normal + abnormal data yields 64% AUC.

A similar study [49], in which image and latent vector spaces are optimized for anomaly detection, utilizes an adversarial network such that the generator comprises encoder-decoder-encoder sub-networks. The objective of the model is to minimize the distance between both real/generated images and their latent representations jointly, which overall outperforms the previous state-of-the-art both statistically and computationally (UBA: 0.643, FFOB: 0.882 – AUC). A follow-up work [50] improves the performance of [49] further by (i) utilizing skip-connections in the generator network to cope with higher resolution images, and (ii) learning the latent representations within the discriminator network (UBA: 0.940, FFOB: 0.903 – AUC).

Another anomaly detection algorithm [51] (i) first extract the feature of the

normal images from Inception v3 [147] alike network, (ii) subsequently trains a multivariate Gaussian model to capture the normal distribution of CAST dataset. Anomaly score of a test sample is based on its likelihood that is relative to the model, which overall yields 92.5% AUC.

2.4.6 Segmentation

Due to the scarcity of datasets with pixel-level annotation, the task of segmentation is understudied within the field. One of the published work [144] addresses segmentation and anomaly detection tasks together, whereby a dual-CNN pipeline initially segments RoI via Mask RCNN [3] and classifies the regions as benign/abnormal via ResNet-18 [1], achieving 97.6% segmentation mAP and 66.0% anomaly detection accuracy. Another work proposes three-stage approach, whereby (i) object-level segmentation is achieved by the use of Mask RCNN [3], (ii) sub-component regions are segmented via super-pixel segmentation and (iii) final object classification is performed via fine-grained CNN classification, which overall yields 97.91% anomaly detection accuracy on $7,878$ electronic items.

Reference	Domain	Problem	Method
Akçay *et al.* [9]	Baggage	Object Classification	CNN with transfer learning
Svec [100]	Baggage	Object Classification	CNN with transfer learning
Andrews *et al.* [111]	Cargo	Anomaly Detection	Train CNN features with Random Split Trees
Jaccard *et al.* [58]	Cargo	Object Classification	oBIF+RF for non-empty cargo detection, followed by CNN for car detection
Jaccard *et al.* [101]	Cargo	Object Classification	CNN from scratch outperforms RF
Rogers *et al.* [104]	Cargo	Object Classification	Evaluation of high and low energy x-ray imagery
Caldwell *et al.* [41]	Cargo, Baggage	Object Classification	Transferability between domains
Yuan and Gui [105]	Tera Hertz	Object Classification	Two-stage. Classify from RGB, then Tera-Hertz images.
Zhao *et al.* [106]	Baggage	Image Generation, Object Classification	Generate X-ray objects via GAN, and classify with CNN
Yang *et al.* [44]	Baggage	Image Generation Object Classification	Generate X-ray objects via GAN, and classify with CNN
Miao *et al.* [43]	Baggage	Object Classification	with class-balanced hierarchical refinement
Morris *et al.* [134]	Baggage	Object Classification	Region-based detection with Z-effective
Akçay and Breckon [108]	Baggage	Object Detection	Object Detection, Faster-RCNN is the best.
Liang *et al.* [45]	Baggage	Object Detection	RFCN is the best. Multi-view outperforms single view.
Steitz *et al.* [46]	Baggage	Object Detection	F-RCNN with multi view pooling is superior to single view only.
Liu *et al.* [47]	Baggage	Object Detection	YOLOv2 achieves real time performance.
Xu *et al.* [107]	Baggage	Object Detection	Localizes the threat material from the X-ray images via attention mechanisms
Islam *et al.* [148]	Baggage	Object Detection	track passengers and their belongings in airports while passing X-ray security checkpoints
Andrews *et al.* [48]	Cargo	Anomaly Detection	Fusion of the raw-input and residual error with feature encoding from the hidden layers.
Akçay *et al.* [49]	Baggage	Anomaly Detection	encoder- decoder-encoder sub-networks. Minimize latent vector and image space.
Akçay *et al.* [50]	Baggage	Anomaly Detection	Use of skip connections. Minimize latent vector in the discriminator network.
Griffin *et al.* [51]	Baggage	Anomaly Detection	Feature Extraction with CNN, then train with Gaussian model.

Table 2.2: Overview of deep learning approaches applied within X-ray security imaging.

2.5 Conclusion

This chapter taxonomises conventional machine and modern deep algorithms utilised within X-ray security imaging — the taxonomy sub-categorises image analysis and machine learning approach into the traditional algorithms. Subsequently, a thorough investigation of advanced algorithms reviews the current deep learning techniques within the classification, detection, anomaly detection and segmentation tasks. The discussion finally outlines the strengths and weaknesses of the existing techniques and envision future challenges and directions of the field.

Motivated by the promising performance of the modern deep learning approaches, the next chapter evaluates their use within the classification and detection tasks. The evaluation provides a thorough comparison against the conventional machine learning algorithms that achieved state-of-the-art results in pre-deep learning era (Chapter 2). Furthermore, we consider the use of deep learning approaches beyond this initial remit of classification and detection into the related challenge of generalised anomaly detection (Chapters 4 and 5).

CHAPTER 3

On Using Deep Convolutional Neural Network Architectures for Object Classification and Detection within X-ray Baggage Security Imagery

3.1 Introduction

This chapter explores the use of deep Convolutional Neural Networks (CNN) with transfer learning for the image classification and detection problems posed within the context of X-ray baggage security imagery. The use of the CNN approach requires large amounts of data to facilitate a complicated end-to-end feature extraction and classification process, where the model predicts the corresponding class label for the given input. Within the context of X-ray security screening, limited availability of object of interest data examples can thus pose a problem. To overcome this issue, this chapter employs a transfer learning paradigm such that a pre-trained CNN, primarily trained for generalised image classification tasks where sufficient training data exists, can be optimised explicitly as a later secondary process towards this application domain. In addition to the classification task, the chapter also explores the applicability of multiple CNN driven detection paradigms, where the models not only output the class label but also localise the object by predicting its bounding box coordinates. The work presented here is one of the first exploiting CNN within X-ray security imaging.

For the classification task, the chapter contains an extensive set of experiments to evaluate the strength of CNN features, and traditional hand-crafted features (SIFT, SURF, FAST, KAZE [8]) explained in Chapter 2. As with [9], we perform layer freezing by fixing parameters from the source domain without any further optimisation to observe how fixing the layer parameters at different points in the network influences the overall performance of the transfer learning-based tuning of the end-to-end CNN. Furthermore, in contrast to [9, 57] comparing end-to-end CNN classification with traditional feature-driven pipelines, we additionally present results whereby we extract the output of the last layer of a given CNN (fc_7 of Krizhevsky2012 [7]) as a feature map itself. We subsequently train an SVM classifier, generally used as the final classification stage of feature-driven approaches [8, 77–79, 84], to provide a consistent feature-space comparison between both learned (CNN) and traditional feature representations.

In addition to the proposed classification scheme, we explore the task of object detection within this problem domain to both classify and localise objects of interests

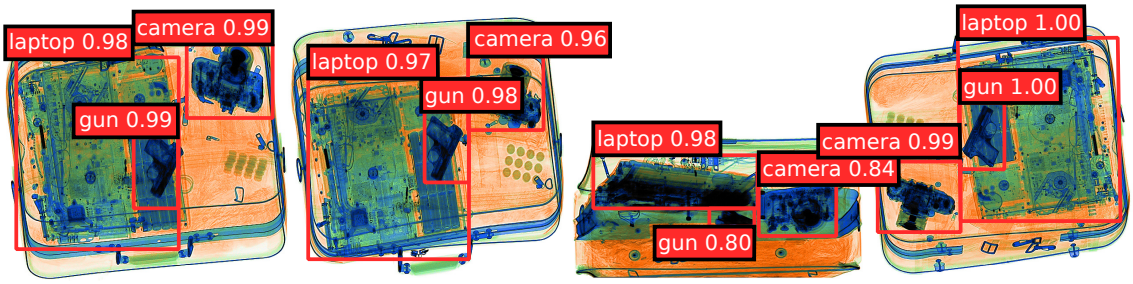


Figure 3.1: Exemplar X-ray baggage imagery multiple objects from Figure 1.1, and the detection results using ResNet-50 [1]. Values next to the object labels indicate the predicted probability that the object belongs to the corresponding class.

from the image by predicting class label and bounding box coordinates. We therefore investigate the use of a sliding window paradigm (akin to [84, 99]) and evaluate contemporary approaches to learn efficient object localization via R-CNN [10], R-FCN [11], and YOLOv2 [12] approaches. As shown in previous work [9, 57] the challenging and cluttered nature of object detection in X-ray security imagery often poses additional challenges for established contemporary classification and detection approaches, such as RCNN/R-FCN [10, 11].

Overall, the main contributions of this chapter are as follows:

- the exhaustive evaluation of classification architectures of [1, 5, 7, 149] against prior work in the field from [8, 61, 77, 78, 127]
- the feature-space comparison of the end-to-end CNN classification results of [9,57] against the final stage SVM classification on the extracted CNN features,
- the comparison of the region based object detection/localization strategies of [10, 11] against the prior strategies proposed in [99, 150].

Contrasting performance results are obtained against the prior published studies of [8, 9] over a comprehensive dataset of 11, 627 examples making this one of the largest combined X-ray object detection and classification study in the literature at the time of publication. Moreover, the evaluation is strengthened further by using UK government evaluation dataset [15]. Overall, we identify classification approaches and detection strategies that outperform the prior work of [9, 84, 99]

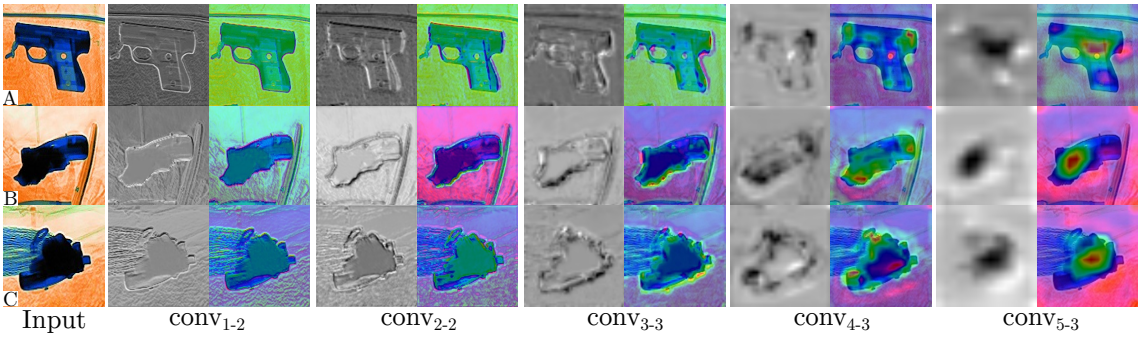


Figure 3.2: Gradient-based class activation map (Grad-CAM [4]) of VGG16 [5] trained on X-ray data. The first column of each convolution box demonstrates grayscale Grad-CAM, while the second column is Grad-CAM heatmap on an input image.

and establish the use of CNN architectures for detection and classification in x-ray security imagery via the paradigm of transfer learning.

3.2 Classification

Automated threat screening task in X-ray baggage imagery can be considered as a classical image classification problem. Here we address this task using convolutional neural networks and transfer learning approaches based on the prior work of [1, 5, 7, 56, 151, 152], and expanding the earlier preliminary studies of [9, 57]. To these ends, we initially outline a brief generalized background for convolutional neural networks and transfer learning, and explain our approach to applying these techniques to object classification within X-ray baggage security imagery.

3.2.1 Transfer Learning

Modern CNN architectures such as [1, 5, 7, 152] are trained on huge datasets such as ImageNet [55] which contains approximately a million of data samples and 1000 distinct class labels. However, the limited applicability of such training and parameter optimization techniques to problems where such large datasets are not available gives rise to the concept of transfer learning [151]. The work of [56] illustrated that each hidden layer in a CNN has distinct feature representation related characteristics among of which the lower layers provide general feature extraction capabilities (akin to Gabor filters and alike), while higher layers carry information that is increasingly

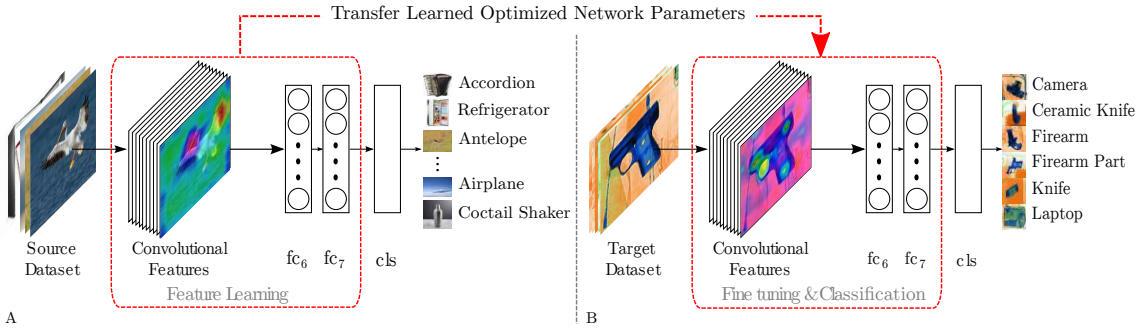


Figure 3.3: Transfer learning pipeline. (A) shows classification pipeline for a source task, while (B) is a target task, initialized by the parameters learned in the source task.

more specific to the original classification task.

Figure 3.2, for instance, demonstrates Gradient-based class activation map (Grad-CAM [4]) of VGG16 [5] for an example X-ray classification object. Lower layers - *i.e.* $conv_{1-2}$ and $conv_{2-2}$, behave as edge detectors, while higher layers like $conv_{4-3}$ and $conv_{5-3}$ provides more specific representations belonging to the input image. This finding facilitates the verbatim re-use of the generalized feature extraction and representation of the lower layers in a CNN, while higher layers are fine-tuned towards secondary problem domains with related characteristics to the original.

Using this paradigm, as demonstrated in Figure 3.3, we can leverage the *a priori* CNN parametrization of an existing fully trained network on a generic 1000+ object class problem [55] (Figure 3.3A), as a starting point for optimization towards to the specific problem domain of limited object class detection within X-ray images (Figure 3.3B). Instead of designing a new CNN with random weight initialization, we instead adopt a pre-trained CNN, pre-optimized for generalized object recognition, and fine-tune its weights towards our specific classification domain

3.2.2 Classification within X-ray Security Imagery

To investigate the applicability of convolutional neural networks in object classification in X-ray baggage imagery, we address two specific target problems:- a) binary classification problem that performs firearm detection (i.e., gun vs no-gun) akin to that of the prior work of [8] to compare CNN features to conventional handcrafted attributes; b) a multi-class X-ray object classification problem (6 classes: firearm, firearm-components, knives, ceramic knives, camera and laptop), which further in-

vestigates the performance of CNN for the classification of multiple X-ray objects. The following subsection describes the datasets we use in our experiments.

Datasets

To perform classification tasks, we use four types of datasets described below:

- *Dbp_2:* Our dataset ($11,627$ X-ray images) are constructed using single conventional X-ray imagery with associated false-colour materials mapping from dual-energy [17]. To generate a dataset for firearm detection, we manually crop baggage objects, and label each accordingly (e.g., Figure 3.4) - on the assumption, an in-service detection solution would perform scanning window search through the whole baggage X-ray image. In addition to manual cropping, we also generate a set of negative images by randomly selecting 256×256 fixed-sized overlapping image patches from a large corpus of baggage X-ray images that do not contain any target objects. Following these approaches, our evaluation datasets consist of $19,398$ X-ray sample patches for a classical two-class firearms detection problem (positive class: $3,179$ firearm images / $1,176$ images of firearm components; negative class: 476 images of cameras, $2,750$ knives, $1,561$ ceramic knives, 995 laptops and $9,261$ cropped images of background clutter)

- *Dbp_6:* For the multiple class problem, we separate firearms and firearm subcomponents into two distinct classes to make the problem even more challenging. Likewise, regular and ceramic knives are considered as two different class objects, which overall we have a 6-class problem for the multi-class task (i.e., each patch being either one of the six object labels).

  In addition to these datasets, we also use the UK government evaluation dataset [15], which is available upon request from the UK Home Office Centre for Applied Science and Technology (CAST). This dataset comprises of both expertly concealed firearm (threat) items and operational benign (non-threat) imagery from commercial X-ray security screening operations on the the UK

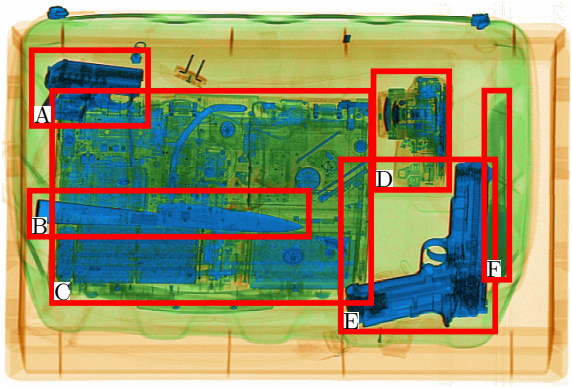


Figure 3.4: Exemplar X-ray baggage image with extracted data set regions including background samples. Type of baggage objects in the dataset is as follows: (A) Firearm Component, (B) Ceramic Knife, (C) Laptop, (D) Camera , (E) Firearm , (F) Knife

(baggage/parcels). From this dataset, we define two evaluation problems based on the provided annotation for the presence of firearms threat items.

- *Full Firearm vs Operational Benign - (FFOB):* comprising $4,680$ firearm threat and 5,000 non-threat images, and is denoted as FFOB.

- *Firearm Parts vs Operational Benign - (FPOB):* contains $8,770$ firearm and parts threat and $5,000$ non-threat images (denoted FPOB, comprising of annotations as any of *{bolt carrier assembly, Pump action, Set, Shotgun, Sub-Machine-Gun}*).

We split the datasets into training (60%), validation (20%) and test sets (20%) such that each split has similar class distribution, but unseen test set contains somewhat challenging samples never trained before. We also perform random flipping, random cropping, and rotation to each sample to augment the datasets. Moreover, when computing the loss, we weight the data such that the classes with fewer samples have more weight. This weighting approach eliminates the classification bias stemming from the class imbalance.

Classification

Using transfer learning paradigm explained in Section 3.2.1, this work leverages the *a priori* CNN parametrization of an existing fully trained network, on a generic

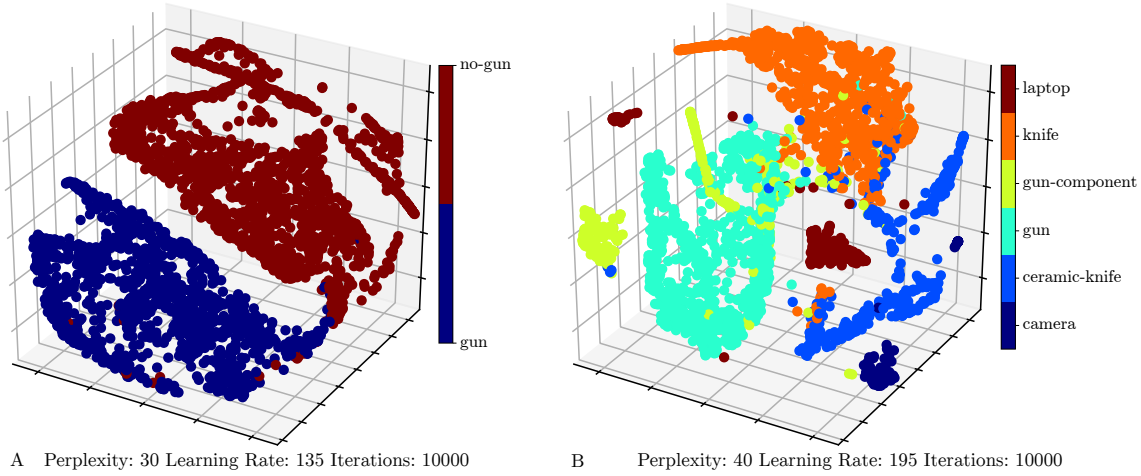


Figure 3.5: t-SNE [6] visualization of feature maps extracted from the last fc layer of VGG_{16} [5] fine-tuned for binary (A) and multi-class (B) problems.

1000 object class problem [55], as a starting point for optimization towards another problem domain of limited object class detection within X-ray images.

For the binary classification problem, we specifically make use of the CNN configuration designed by Krizhevsky *et al.* [7], having 5 convolutional layers (*conv*), 3 fully-connected layers (fc), and trained on the ImageNet dataset on a 1000 class image classification problem, denoted as AlexNet [7].

The first step is to fine-tune all of the *conv* and fc layers of the network via transfer learning on the training set of the target classification problem. In addition to this, we also perform layer freezing, meaning that instead of updating layer parameters for our task, we use the original unmodified weights from the initial trained CNN parametrization of [7]. This allows us to observe how fine-tuning each layer impacts the overall performance.

Also, having fine-tuned the parameters via this transfer learning approach, we extract the features of the last fully connected layer (fc_7) to train on an SVM classifier. This allows us to additionally compare the internal feature space representation of the CNN model to alternative more traditional (handcrafted) BoVW features as used in prior work [8].

Evaluation of our proposed approach is performed against the prior SVM-driven work of Kundegorski *et. al.* [8] within a BoVW framework. SVM are trained using Radial Basis Function (RBF) kernel $\{SVM_{RBF}\}$ with a grid search over kernel pa-

rameter, $\gamma = 2^x : x \in \{-15, 3\}$, and model fitting cost, $c = 2^x : x \in \{5, 15\}$, using k-fold cross validation ($k = 5$) with F-score optimization (being more representative then accuracy for unbalanced datasets). The results for the best performing parameter set are reported for each feature configuration.

The second set of experiments is the classification of multiple baggage objects, a more complex six class object problem. Here the lesser performing SVM with handcrafted features are not considered (Table 3.1), in favour of the CNN approach. Instead, we fine-tune AlexNet [7], VGG [5] and ResNet [1], each of which are top-performing entries of ImageNet [55] object recognition competition.. By doing so, we aim to evaluate the feasibility of CNN for this problem domain further.

To update the parameters of all the networks during training, we use cross-entropy for the loss function and utilize Adam [4] optimizer with a learning rate of 10^{-3}, and a weight decay of 0.005. Our stopping criterion is to terminate optimization where validation starts to reduce, while training accuracy continues to improve. This fork between training and validation performance usually takes 30 epochs for this task.

3.2.3 Evaluation

The performance is evaluated by the comparison of True Positive Rate (TP) (%), False Positive Rate (FP) (%) together with Precision (P), Accuracy (A) and F-score (F) (harmonic mean of precision and true positive rate).

Results for the two class problem is given in Table 3.1, which is divided into four sections: - first section lists the performance of the CNN approach, notated as $AlexNet_{a_b}$, meaning that the network is fine-tuned from layer a to layer b, while the rest of the layers are frozen (Table 3.1, top). This means, for instance, $AlexNet_{4-8}$ is trained by fine-tuning the layers $\{4, 5, 6, 7, 8\}$ and freezing the layers $\{1, 2, 3\}$ (i.e. remain unchanged from the pre-trained model of [7]). The second section has the results of an SVM classifier trained on the output of the last layer of CNN (Table 3.1, middle upper). Similar to the first section, we again perform layer freezing here for a consistent comparison of CNN features and BoVW features. The third section shows fine tuning results based on contemporary end to end CNN architectures

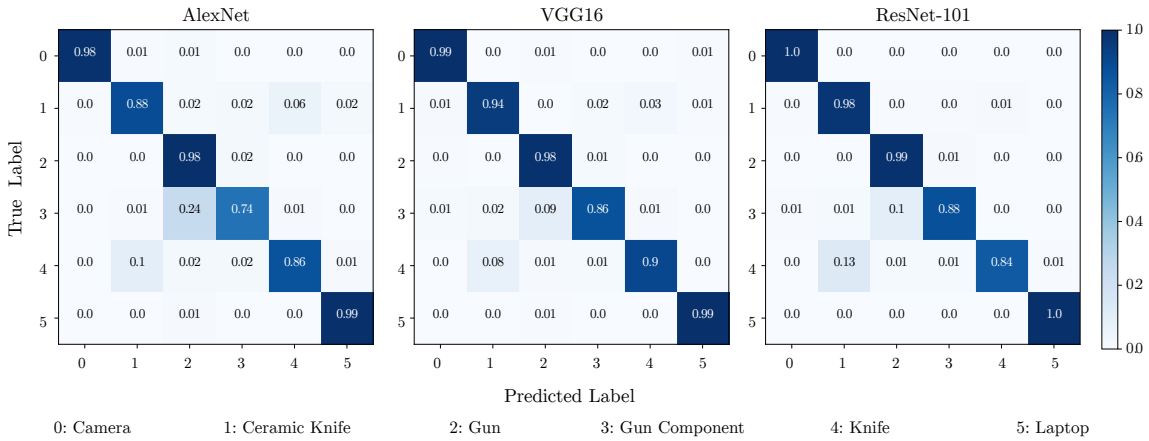


Figure 3.6: Confusion matrices for AlexNet [7], VGG16 [5] ResNet-50 [1] fine tuned for multi class problem

(VGG_M [149], VGG_{16} [5], $ResNet_{18}$ [1], $ResNet_{50}$ [1], $ResNet_{101}$ [1], Table 3.1, middle lower). The last section lists the best performing BoVW feature detector/descriptor variants trained with SVM in the work of [8] (Table 3.1, bottom).

Table 3.1 shows the performance results of firearm detection. We see that true and false positives have a general trend to decrease as the number of fine-tuned layers reduces. Likewise, freezing lower layers reduces the accuracy of the models.

Training an SVM classifier on CNN features with layer freezing yields relatively better performance than the standard end to end CNN results. We see a performance pattern such that fine-tuning more layers has a positive impact on the overall performance. For instance, SVM trained on fully fine-tuned CNN has the highest performance on all of the metrics, outperforming the prior work of [8] and [9] (Table 3.1).

For an end to end fine-tuning using contemporary architectures, we observe the direct proportion of performance and network complexity. $ResNet_{101}$ [1], for instance, is the best performing network among all of the end to end CNN networks (Table3.1).

It is also significant to note that the performance of the best feature detector/descriptor combination of BoVW approach (FAST/SURF [8]) is worse than any of the CNN features given in Table 3.1. Further comparison of BoVW+SVM against CNN+SVM proves the superiority of CNN features to traditional handcrafted features (Table 3.1).

		TP%	FP%	P	A	F
A. CNN [7] Layer Freezing	$AlexNet_{1\text{-}8}$	99.26	4.08	0.741	0.961	0.849
	$AlexNet_{2\text{-}8}$	98.53	2.40	0.832	0.983	0.902
	$AlexNet_{3\text{-}8}$	96.32	2.19	0.844	0.980	0.900
	$AlexNet_{4\text{-}8}$	95.59	2.96	0.790	0.973	0.865
	$AlexNet_{5\text{-}8}$	98.16	4.68	0.711	0.961	0.825
	$AlexNet_{6\text{-}8}$	96.32	5.15	0.693	0.954	0.806
	$AlexNet_{7\text{-}8}$	94.49	3.65	0.754	0.961	0.839
	$AlexNet_{8}$	95.22	4.21	0.733	0.960	0.828
CNN [7] + SVM Layer Freezing	$AlexNet_{1\text{-}8}$	**99.56**	**1.07**	**0.997**	**0.994**	**0.996**
	$AlexNet_{2\text{-}8}$	99.30	1.50	0.996	0.991	0.994
	$AlexNet_{3\text{-}8}$	99.18	1.93	0.995	0.989	0.993
	$AlexNet_{4\text{-}8}$	98.92	1.86	0.995	0.988	0.992
	$AlexNet_{5\text{-}8}$	98.80	2.07	0.994	0.986	0.991
	$AlexNet_{6\text{-}8}$	98.68	3.00	0.991	0.983	0.983
	$AlexNet_{7\text{-}8}$	98.64	4.15	0.989	0.980	0.980
	$AlexNet_{8}$	98.42	5.43	0.985	0.976	0.976
CNN End to End	VGG_{M} [149]	98.38	0.36	0.998	0.987	0.980
	VGG_{16} [5]	99.08	1.14	0.997	0.990	0.985
	$ResNet_{18}$ [1]	99.38	1.43	0.996	0.992	0.988
	$ResNet_{50}$ [1]	99.54	1.00	0.998	0.995	0.992
	$ResNet_{101}$ [1]	99.66	1.14	0.997	0.995	0.993
BoVW SVM [8]	SURF/SURF	79.2	3.2	0.88	0.93	0.83
	KAZE/KAZE	77.3	3.9	0.85	0.92	0.81
	FAST/SURF	83.0	3.3	0.88	0.94	0.85
	FAST/SIFT	80.9	4.3	0.85	0.92	0.83
	SIFT/SIFT	68.3	4.2	0.83	0.90	0.75

Table 3.1: Results of CNN and BoVW on Dbp_2 dataset for firearm detection. $AlexNet_{ab}$ denotes that the network is fine tuned from layer a to layer b.

	P	R	A	F
$AlexNet_{1\text{-}8}$	0.911	0.904	0.904	0.906
$AlexNet_{2\text{-}8}$	0.842	0.841	0.833	0.835
$AlexNet_{3\text{-}8}$	0.843	0.841	0.844	0.841
$AlexNet_{4\text{-}8}$	0.841	0.853	0.844	0.846
$AlexNet_{5\text{-}8}$	0.833	0.821	0.823	0.811
$AlexNet_{6\text{-}8}$	0.820	0.810	0.819	0.809
$AlexNet_{7\text{-}8}$	0.774	0.793	0.722	0.761
$AlexNet_{8}$	0.721	0.742	0.701	0.712
VGG_{M} [149]	0.928	0.932	0.923	0.926
VGG_{16} [5]	0.931	0.943	0.940	0.936
$ResNet_{18}$ [1]	0.933	0.943	0.936	0.937
$ResNet_{50}$ [1]	0.934	0.910	0.923	0.917
$ResNet_{101}$ [1]	**0.936**	**0.946**	**0.937**	**0.938**

Table 3.2: Statistical evaluation of CNN architectures (AlexNet, VGG, and ResNet) on Dbp_6 dataset for multi-class problem.

Table 3.2 shows the overall performance of the networks fine-tuned for multiple class problem. Like Table 3.1, fine-tuning the entire network yields the best performance. A conclusion can be reached from these results that fine-tuning higher-level layers and freezing lower ones have a detrimental impact on the performance of the CNN model. Similar to Table 3.1, performance and network complexity are also directly proportional. With relatively lower complexity than the rest, AlexNet [7] has the lowest accuracy of 92.4. $ResNet_{101}$ [1], on the other hand, achieves the highest on all metrics (P=93.6% R=94.6% A=93.7% F=93.8%).

In addition, results are presented on the UK government evaluation dataset [15] in Tables 3.3 and 3.4 . Within Table 3.3 and 3.4 we present results for classification only (following the approach of Section 3.2.1), where we can see comparable performance to the earlier results presented in Tables 3.1 and 3.2.

Figure 5.8 depicts the t-SNE [6] visualization of feature maps of the down-projected internal feature space representation extracted from VGG_{16} [5] fine-tuned for binary (A) and multi-class (B) problems. In both cases, classes are well separated, showing the capability of CNN features within this problem domain (Figure 5.8 Figure 3.6 depicts per-class accuracy obtained via the use of AlexNet [7] and $ResNet_{101}$ [1], the worst and best performing networks within this task. We see

	TP%	FP%	P	A	F
AlexNet [7]	99.830	0.943	0.990	0.994	0.994
VGG_M [5]	99.010	0.000	1.000	0.995	0.995
VGG_{16} [5]	99.831	0.000	1.000	0.999	0.999
$ResNet_{18}$ [1]	99.472	0.000	1.000	0.997	0.997
$ResNet_{50}$ [1]	100.00	0.923	0.990	0.995	0.995
$ResNet_{101}$ [1]	100.00	0.311	0.996	0.998	0.998

Table 3.3: Statistical evaluation of varying CNN architectures (AlexNet, VGG, and ResNet) on FFOB dataset [15].

	TP%	FP%	P	A	F
AlexNet [7]	95.088	3.527	0.960	0.958	0.958
VGG_M [5]	95.864	0.919	0.990	0.974	0.974
VGG_{16} [5]	97.238	4.217	0.954	0.965	0.964
$ResNet_{18}$ [1]	95.725	0.744	0.992	0.975	0.974
$ResNet_{50}$ [1]	99.411	1.060	0.988	0.991	0.991
$ResNet_{101}$ [1]	99.608	0.000	1.000	0.998	0.998

Table 3.4: Statistical evaluation of varying CNN architectures (AlexNet, VGG, and ResNet) on FPOB dataset [15].

that laptop and camera object classes are straightforward to classify. In contrast, networks have relatively lower classification confidence for the knife, ceramic knife vs firearm, firearm parts, which obviously stems from the similarity of the objects.

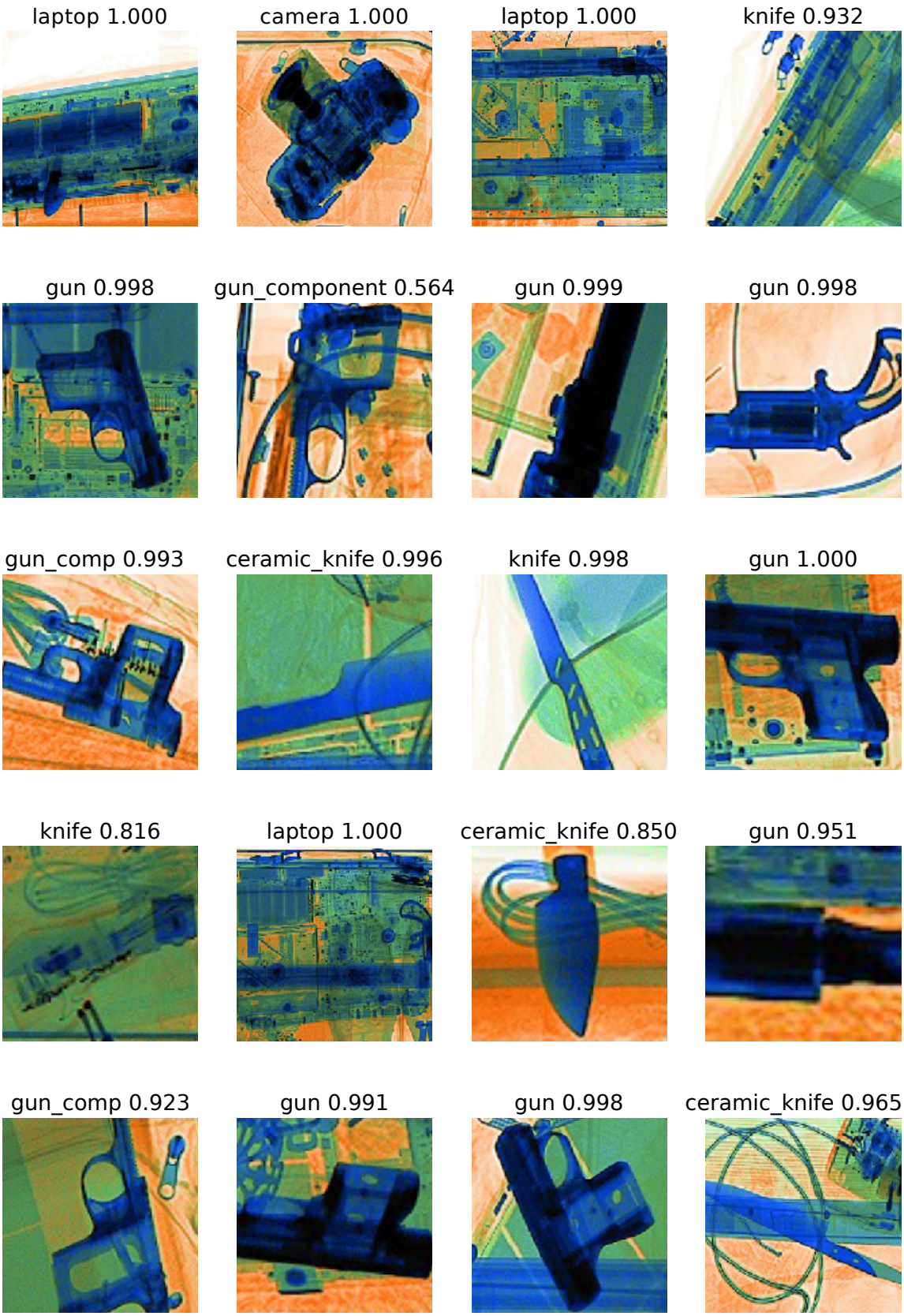


Figure 3.7: Exemplar image cases where a ResNet-50 [1] successfully classifies an object in the presence of clutter and other confusing items of interest (here: background laptop detected, knives/guns missed).

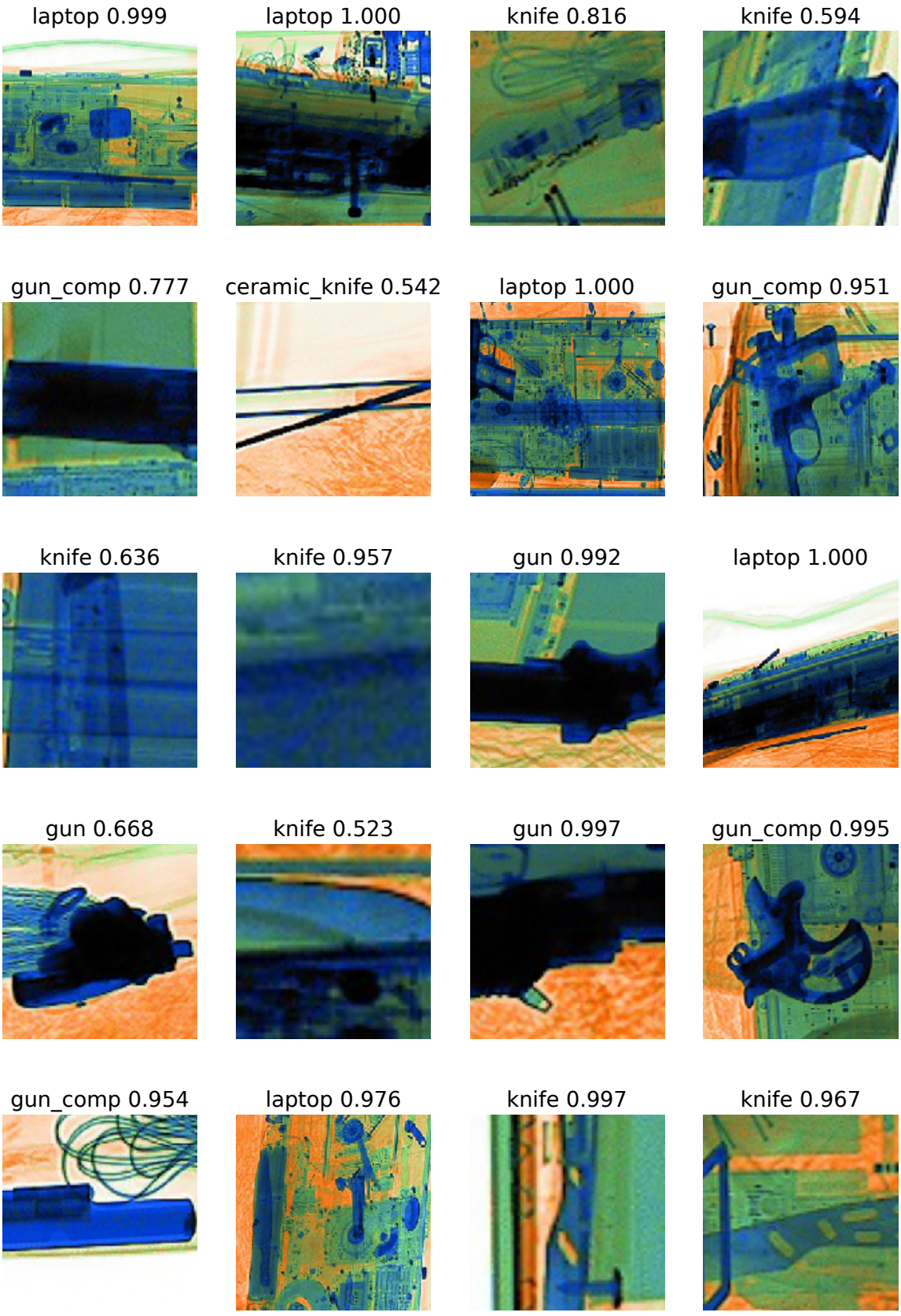


Figure 3.8: Exemplar image cases where a ResNet-50 [1] successfully classifies an object in the presence of clutter and other confusing items of interest.

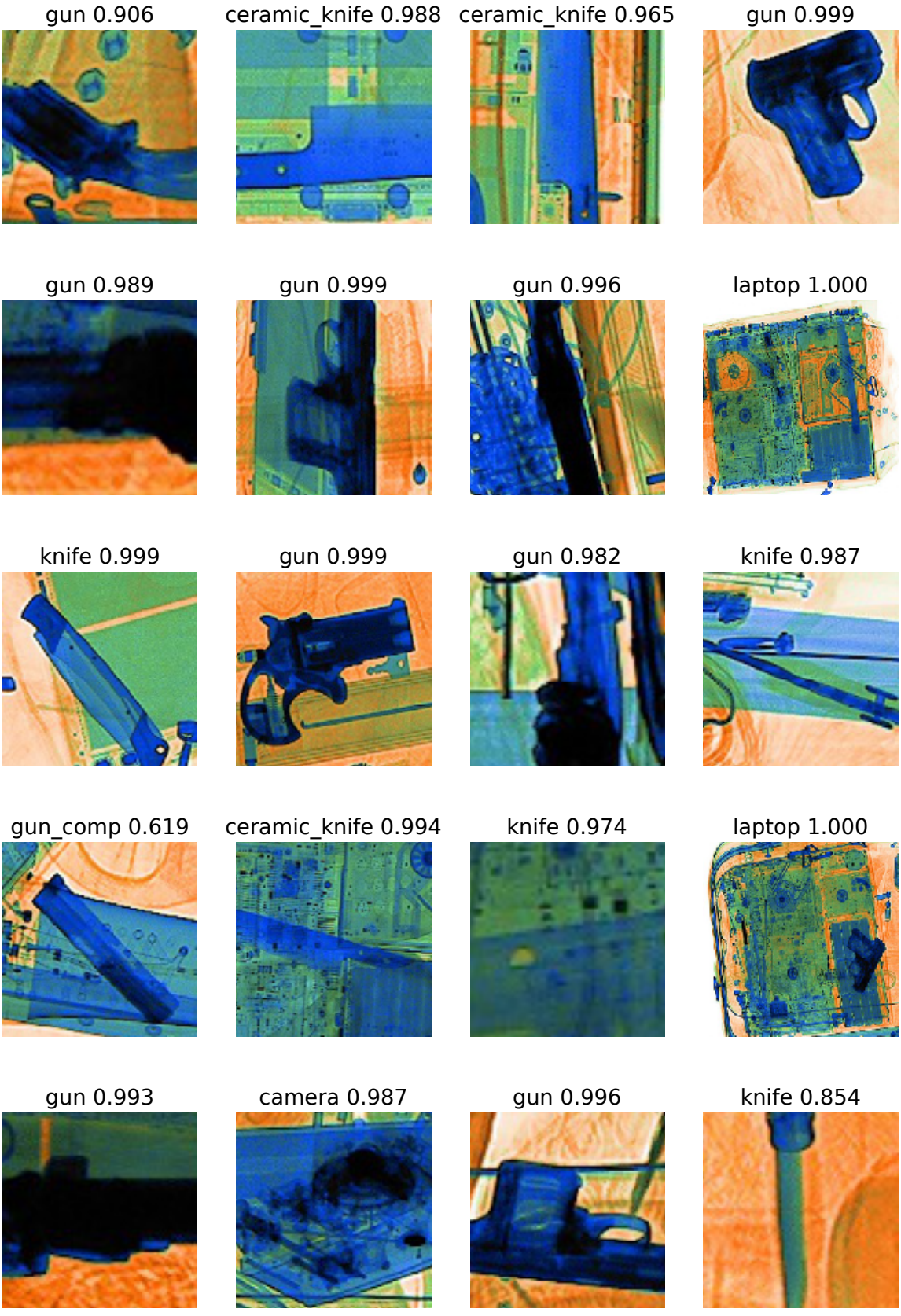


Figure 3.9: Exemplar image cases where a ResNet-50 [1] successfully classifies an object in the presence of clutter and other confusing items of interest.

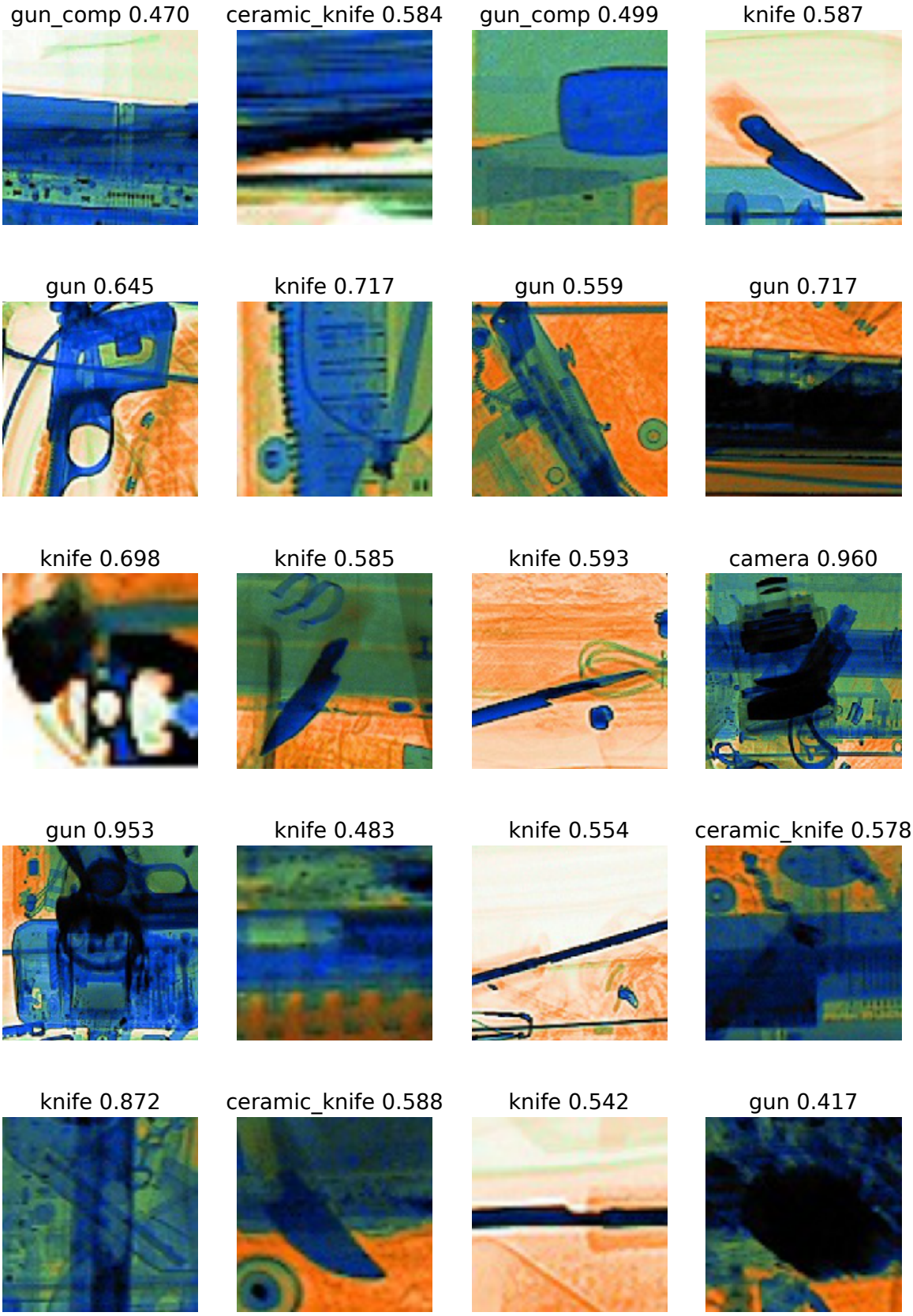


Figure 3.10: Exemplar image cases where a ResNet-50 [1] fails to detect an object in the presence of clutter and other confusing items of interest.

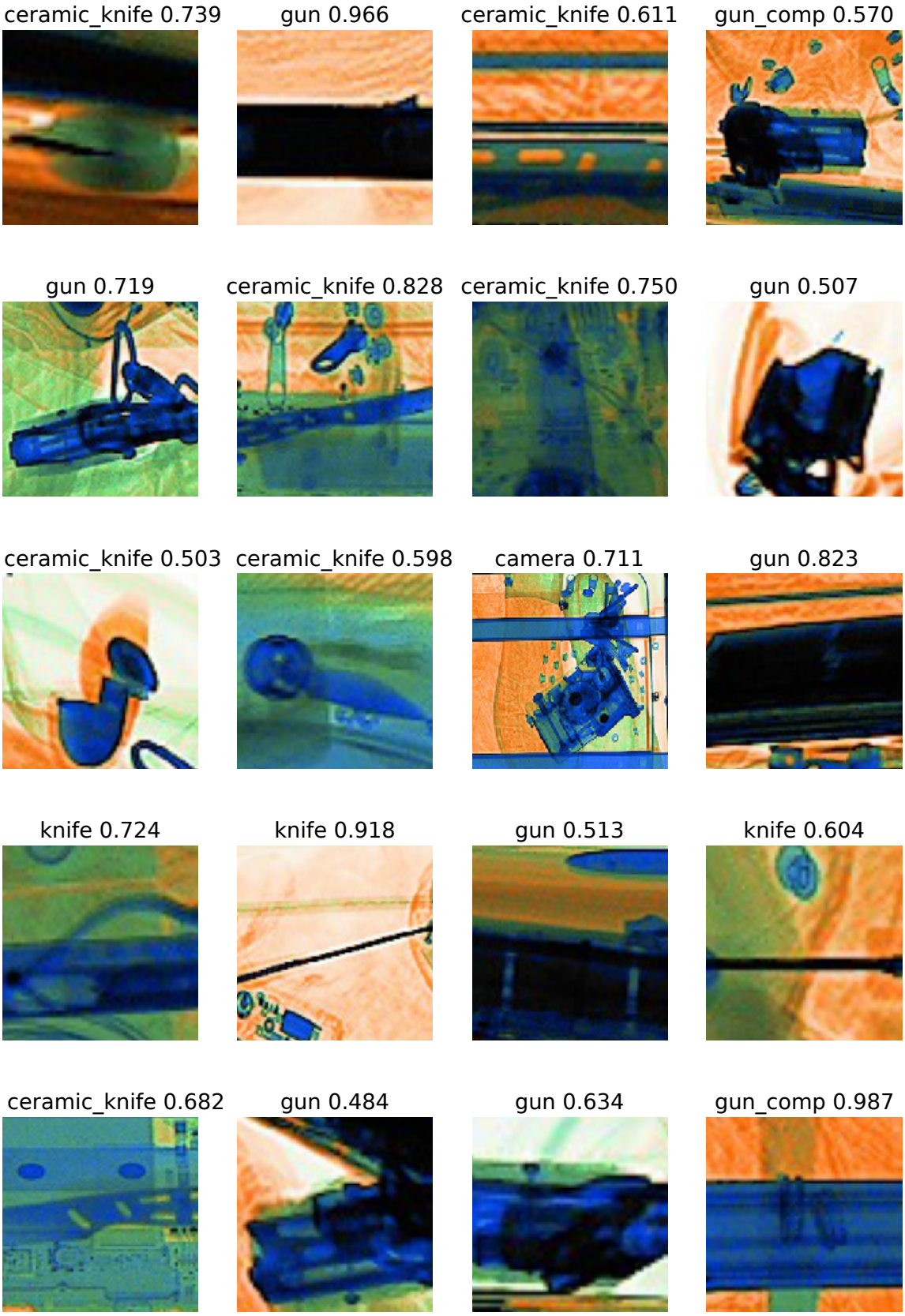


Figure 3.11: Exemplar image cases where a ResNet-50 [1] fails to detect an object in the presence of clutter and other confusing items of interest.

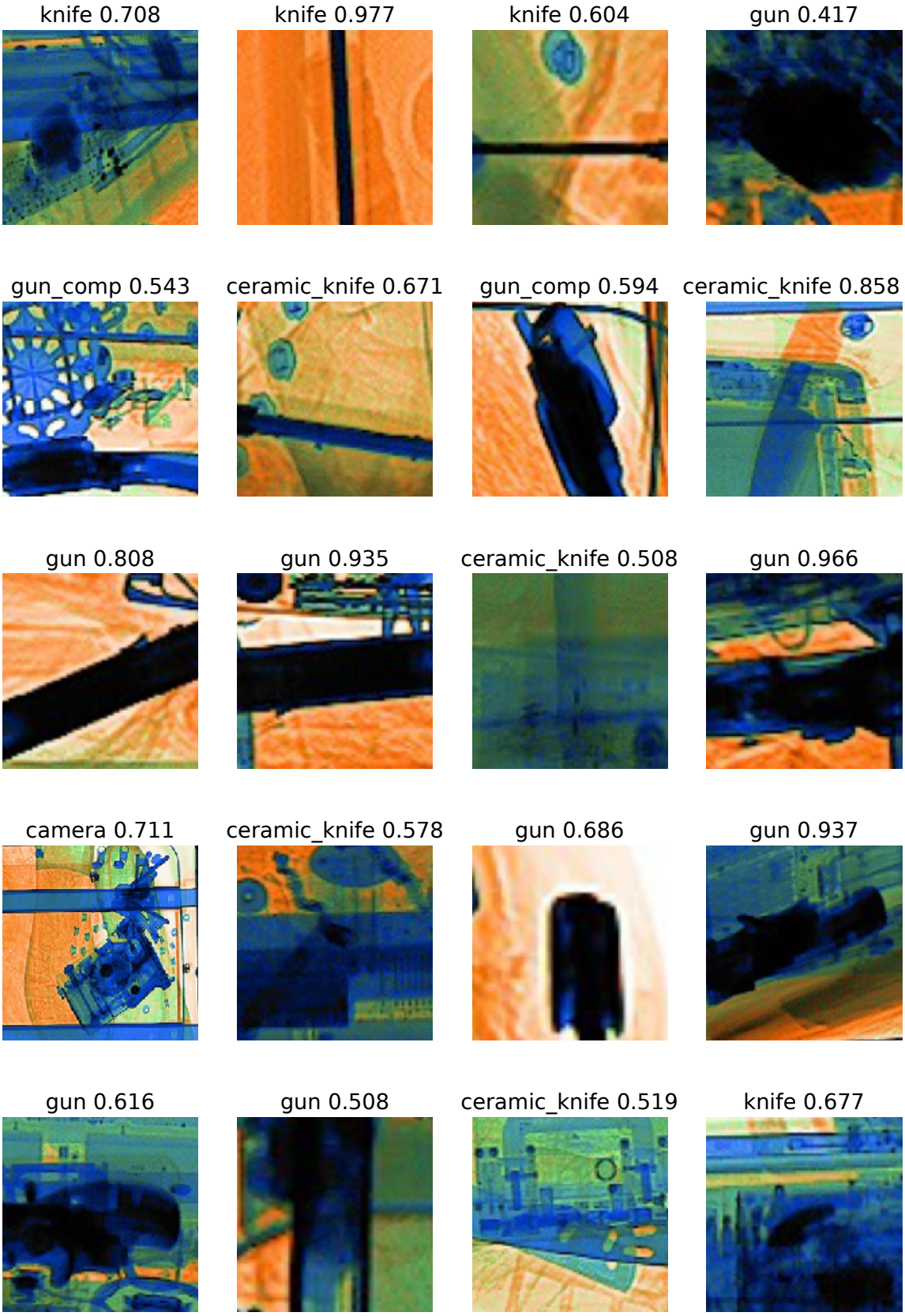


Figure 3.12: Exemplar image cases where a ResNet-50 [1] fails to detect an object in the presence of clutter and other confusing items of interest.

Limitations: Due to the cluttered nature of the input dataset, there are certain cases where CNN based classification fails to classify threats. Figure 3.10, for instance, demonstrates that CNN labels these image examples as laptops with high confidence, as the predominant object signature present in the image patch, while failing to detect the foreground objects of interest. This results in a significant increase in false-negative occurrences (Table 3.2). We consider this primarily as an object detection problem, and hence explore the contemporary object detection strategies in the subsequent part of this study.

3.3 Object Detection

We see from Section 3.2 that CNN-based classification approaches via transfer learning yield promising performance, especially for single and non-occluded X-ray image patches. When it comes to classifying multiple objects (Figures 3.10, 3.11, 3.12), however, more sophisticated approaches are needed to perform joint localization. Here we give a brief introduction to CNN based object detection algorithms for an exhaustive evaluation within X-ray baggage domain.

3.3.1 Detection Strategies

Within this work, we consider a number of competing for contemporary detection frameworks and explore their applicability and performance for generalised object detection in X-ray baggage imagery.

3.3.2 Detection within X-ray Security Imagery

We compare four localization strategies for our object detection task within X-ray security imagery: a traditional sliding window approach [99] coupled with CNN classification [150], Faster RCNN (F-RCNN) [10], R-FCN [11], and YOLOv2 [12], each of which is thoroughly explained in Appendix A.3.2.

Dataset: Instead of using the X-ray patches that we manually crop for the classification task in Section 3.2, here we use full X-ray images to perform binary and multiple class object detection.

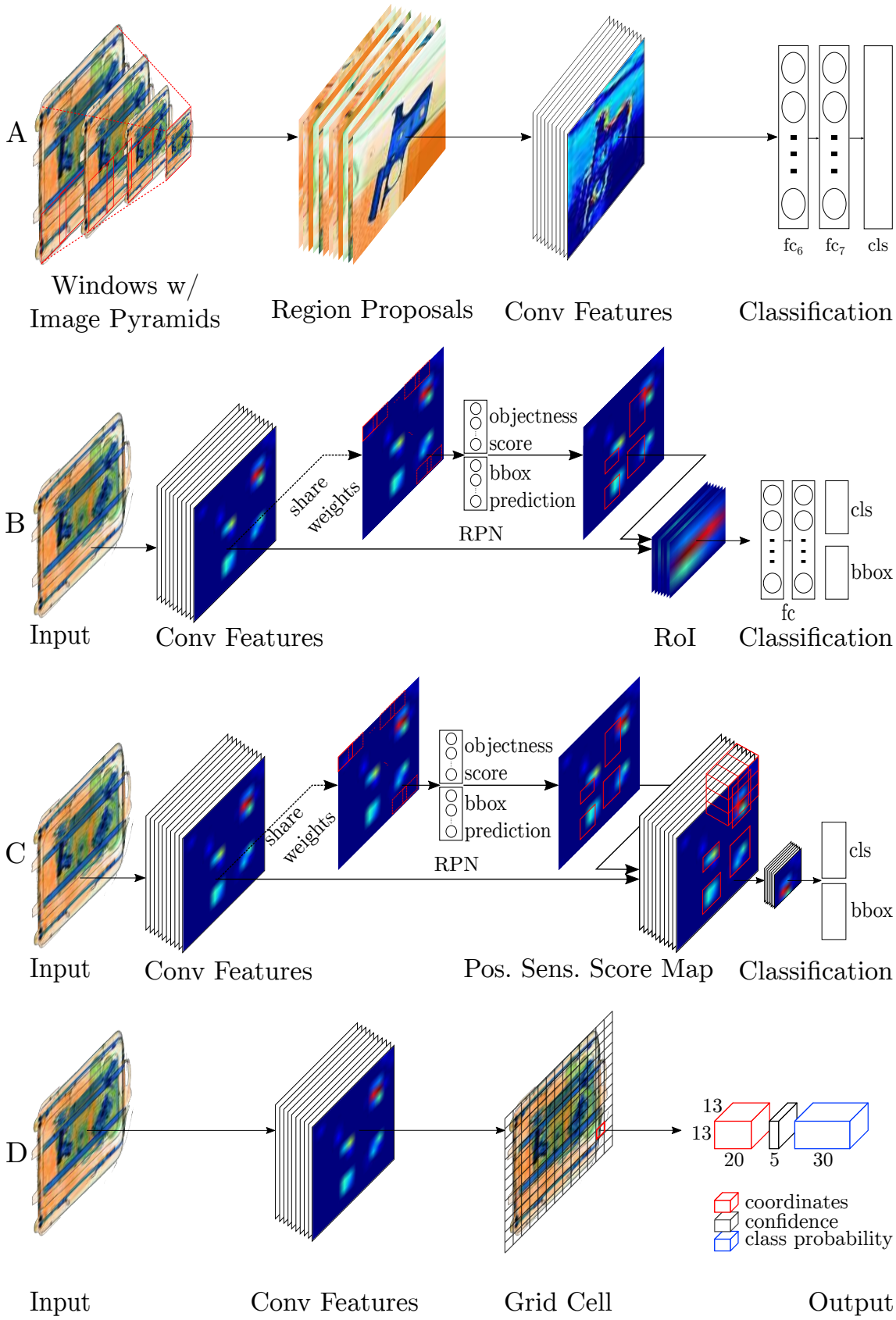


Figure 3.13: Schematics for the CNN driven detection strategies evaluated. A. Sliding Window based CNN (SW-CNN) [8, 9], B. Faster RCNN (F-RCNN) [10], C. R-FCN [11], D. YOLOv2 [12]).

Detection: For sliding window CNN (SW-CNN) we employ 800×800 input image, 256×256 fixed-size window with a step size of 32 to generate region proposals. We also use image pyramids to fit the window to varying sized objects using 9 pyramid levels. For the classification of the proposed regions we use AlexNet [7], $\text{VGG}_{\text{M, 16}}$ [5], and ResNet-{50, 101} [1] networks. Although [150] employs an extra bounding box regression layer within their SW-CNN approach, we do not perform regression as none of the prior work within this domain does so [84, 99].

For Faster RCNN [10] we use the original implementation with a few modifications, and train Faster RCNN with AlexNet [7], $\text{VGG}_{\text{M, 16}}$ [5], and ResNet-{50, 101} [1] architectures. Since R-FCN is fully convolutional by design, we only use ResNet-{50, 101} [1] networks for R-FCN to train and test. Pipeline and implementational details of these approaches are thoroughly explained in Appendix A.3.2

For the training of the detection strategies explained here, we employ a transfer learning approach and use the networks pre-trained on ImageNet dataset [55]. In so doing not only increases performance but also reduces training time significantly. We use stochastic gradient descent (SGD) with momentum and weight decay of 0.9 and 0.0005, respectively. The initial learning rate of 0.001 is divided by 10 with step down method in every $10,000$ iteration. For F-RCNN/R-FCN, the batch size is set to 256 for the RPN. All of the networks are trained by using dual-core Intel Xeon E5-2630 v4 processor and Nvidia GeForce GTX Titan X GPU.

3.3.3 Evaluation

Performance of the models is evaluated by mean average precision (mAP), used for PASCAL VOC object detection challenge [153]. To calculate mAP, we perform the following: we first sort n_d detections based on their confidence scores. Next, we calculate the area of intersection over union for the given ground truth and detected bounding boxes for each detection as

$$\Psi(B_{gt_i}, B_{dt_i}) = \frac{\text{Area}(B_{gt_i} \cap B_{dt_i})}{\text{Area}(B_{gt_i} \cup B_{dt_i})}, \tag{3.1}$$

where B_{gt_i} and B_{dt_i} are ground truth and detected bounding boxes for detection i, respectively. Assuming each detection as unique, and denoting the area as a_i, we then threshold it by $\theta = 0.5$ giving a logical b_i, where

$$b_i = \begin{cases} 1 & a_i > \theta; \\ 0 & \text{otherwise.} \end{cases} \tag{3.2}$$

This is followed by a prefix-sum giving both true positives $\vec{t}$ and false positives $\vec{f}$, where

$$\begin{aligned} t_i &= t_{i-1} + b_i, \\ f_i &= t_{i-1} + (1 - b_i). \end{aligned} \tag{3.3}$$

The precision $\vec{p}$ and recall $\vec{r}$ curves are calculated as

$$\begin{aligned} p_i &= \frac{t_i}{t_i + f_i}, \\ r_i &= \frac{t_i}{n_p}, \end{aligned} \tag{3.4}$$

where n_p is the number of positive samples. For a smoother curve, precision vector is then interpolated by using

$$p_i = max(p_i, p_{i+1}). \tag{3.5}$$

We then calculate average precision (AP) based on the area under precision ($\vec{p}$) recall ($\vec{r}$) curve

$$AP = \sum_{i}^{n_d} p_i \Delta r. \tag{3.6}$$

As shown in Eq 3.7, we finally find mAP by averaging AP values that we calculate for C classes.

$$mAP = \frac{1}{C} \sum_{c=1}^{C} AP_c \tag{3.7}$$

Model	Network	mAP	camera	laptop	gun	gun component	knife	ceramic knife
SWCNN	AlexNet	0.608	0.682	0.609	0.748	0.714	0.212	0.683
	VGG_M	0.634	0.707	0.637	0.763	0.731	0.246	0.719
	VGG_{16}	0.649	0.701	0.724	0.752	0.757	0.223	0.734
	$ResNet_{50}$	0.671	0.692	0.801	0.747	0.761	0.314	0.713
	$ResNet_{101}$	0.776	0.881	0.902	0.831	0.848	0.392	0.803
RCNN	AlexNet	0.647	0.791	0.815	0.853	0.582	0.188	0.658
	VGG_M	0.686	0.799	0.855	0.869	0.658	0.210	0.723
	VGG_{16}	0.779	0.888	**0.954**	0.876	0.832	0.304	0.819
F-RCNN	AlexNet	0.788	0.893	0.756	0.914	0.874	0.467	0.823
	VGG_M	0.823	**0.900**	0.834	0.918	0.875	0.542	0.869
	VGG_{16}	0.883	0.881	0.918	0.927	**0.938**	0.721	0.912
	$ResNet_{50}$	0.851	0.844	0.879	0.916	0.901	0.677	0.889
	$ResNet_{101}$	0.874	0.857	0.904	0.931	0.911	**0.732**	0.907
R-FCN	$ResNet_{50}$	0.846	0.894	0.928	0.932	0.918	0.506	0.896
	$ResNet_{101}$	0.856	0.887	0.906	0.942	0.925	0.556	**0.920**
YOLOv2	$Darknet_{288}$	0.810	0.821	0.861	0.914	0.904	0.551	0.814
	$Darknet_{416}$	0.851	0.888	0.883	**0.952**	0.924	0.605	0.851
	$Darknet_{544}$	**0.885**	0.896	0.894	0.943	0.933	0.728	0.913

Table 3.5: Detection results of SW-CNN, Fast-RCNN (F-RCNN) [16], Faster RCNN (F-RCNN) [10], R-FCN [11] and YOLOv2 [12] for multi-class problem (300 region proposals). Class names indicates corresponding average precision (AP) of each class, and mAP indicates mean average precision of the classes.

Tables 3.5 and 3.6 show binary and multi-class detection results for SW-CNN, F-RCNN, R-FCN with varying networks, and a fixed sized number of region proposals of 300, and for YOLOv2 with a fixed network with varying input image size. For completeness, we additionally present the comparative results for Fast R-CNN (RCNN) [16] (detection architecture pre-dating that of F-RCNN [10] and R-FCN [10]).

As a general trend, we observe that performance increases with overall network complexity such that superior performance is obtained with VGG16 and ResNet_{101} for the region-based approaches. This observation holds for both the 2-class and 6-class problems considered here. Overall, YOLOv2 yields the leading performance for both 2-class and 6-class problems. In addition to this set of experiments, we also train the detection approaches using the pre-trained weights of Dbp6 dataset introduced in Section 3.2.2. Since not observing significant nuances in results, we do not include them here.

For the multi-class detection task (Table 3.5) we see a similar performance pattern to that seen in the earlier firearm detection task. Here, SW-CNN performs worse than any network trained using a Faster RCNN or R-FCN architecture. Similirwise, overall mAP of RCNN is lower than any R-FCN and R-FCN architecture. For comparison of F-RCNN and R-FCN, we observe that Faster RCNN achieves its highest peak using VGG16, with higher mAP than ResNet-50 and ResNet101. R-FCN with ResNet-50 and ResNet_{101} yields slightly worse performance, (mAP: 0.846, 0.856) , than that of the best of Faster-RCNN. For the overall performance comparison, YOLOv2 with an input size of 544×544 shows superior performance (mAP: 0.885).

For firearm detection Table 3.6 shows that SW-CNN, even with a complex second stage classification CNN such as VGG16 and ResNet_{101}, performs poorly compared to any other detection approaches. This poor performance is primarily due to lacking a bounding box regression layer (Figure 5.3), a significant performance booster, as shown in [150, 154]. Likewise, the best performance of RCNN with VGG16 (mAP: 0.854) is worse than any F-RCNN or R-FCN. This is because the RPN within F-RCNN and R-FCN provides superior object proposals than the selective-search

approach used in RCNN. For overall performance on the binary firearm detection task, R-FCN with YOLOv2 with an input image of size 416×416 yields the highest mAP of 0.974.

Model	Network	mAP - firearm
SW-CNN	AlexNet	0.753
	VGG_M	0.772
	VGG_{16}	0.806
	$ResNet_{50}$	0.836
	$ResNet_{101}$	0.847
RCNN	AlexNet	0.823
	VGG_M	0.836
	VGG_{16}	0.854
F-RCNN	AlexNet	0.945
	VGG_M	0.948
	VGG_{16}	0.960
	$ResNet_{50}$	0.951
	$ResNet_{101}$	0.960
R-FCN	$ResNet_{50}$	0.949
	$ResNet_{101}$	0.963
YOLOv2	$Darknet_{288}$	0.931
	$Darknet_{416}$	**0.974**
	$Darknet_{544}$	0.962

Table 3.6: Detection results of SW-CNN, Fast-RCNN (RCNN) [16], Faster RCNN (F-RCNN) [10], R-FCN [11] and YOLOv2 [12] for firearm detection problem (300 region proposals).

Figure 3.14 illustrates the impact on the number of region proposals and input image sizes on both detection performance and runtime. Figure 3.14A-B demonstrate detection performance of the approaches on 2-class and 6-class detection tasks, respectively. Increase in the number of region proposals and input image size lead to a rise in detection performance. Overall, YOLOv2 achieves the highest detection on both tasks. Figure 3.14C shows mean runtime in frame per second (fps) where we can see YOLOv2 significantly outperforms the rest of the detection approaches. The lowest fps YOLOv2 achieves (50fps) is still considerably better than the best runtime R-FCN (20), F-RCNN (2.9) and SW-CNN (0.8) achieve.

Figures 3.15, 3.16 and 3.17 illustrate qualitative examples extracted from the

statistical performance analysis of Table 3.5. We see that detection approaches can cope with cluttered datasets where classification methods can fail as shown in Figures 3.10, 3.11, 3.12.

There are cases where the detection strategies fail to localise certain objects of interests. In Figure 3.18, we see that SW-CNN almost always fails to detect occluded objects such as guns and knives on a laptop. F-RCNN, R-FCN and YOLOv2 achieve relatively superior performance than SW-CNN.

Figure 3.19 demonstrate samples whose difficulty is graded as moderate. Similar to that of Figure 3.18, SW-CNN cannot detect occluded objects. R-FCN also struggles to detect occluded items. F-RCNN performs slightly better than SW-CNN and R-FCN such that it is capable of detecting 4 out 5 images, missing the laptop on the 5^{th} image. Among the detection strategies, YOLOv2 is the best performing model for the moderate samples, detecting all objects of interests.

For the difficult examples shown in Figure 3.20, SW-CNN, again, does not perform well, missing all of the concealed items. F-RCNN, R-FCN and YOLOv2, on the other and, perform better detection performance than SW-CNN; however, their detection rate is yet to be promising. It is important to note here that the objects that are missed by the strategies are rather challenging samples. These mis-detected cases could be minimised by exploiting the multiple views such that the networks could increase their detection confidence with the views showing the occluded object from a better angle.

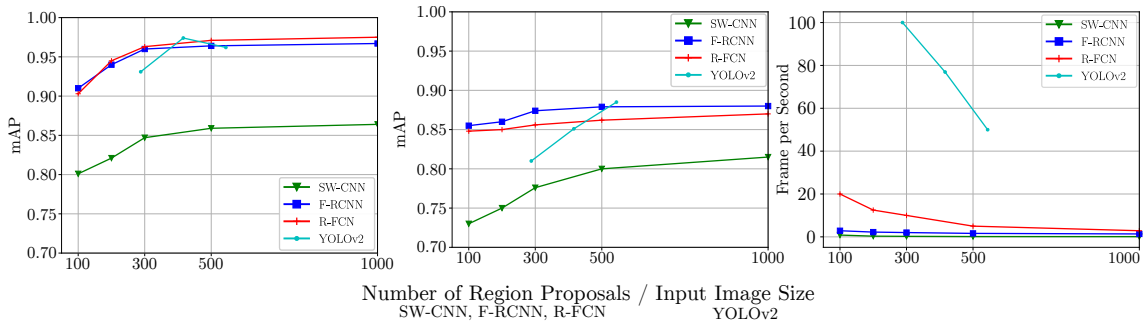


Figure 3.14: Impact of number of box proposals on performance. (A) for binary class (B) for multi-class (C) Runtime. Models are trained using ResNet_{101}

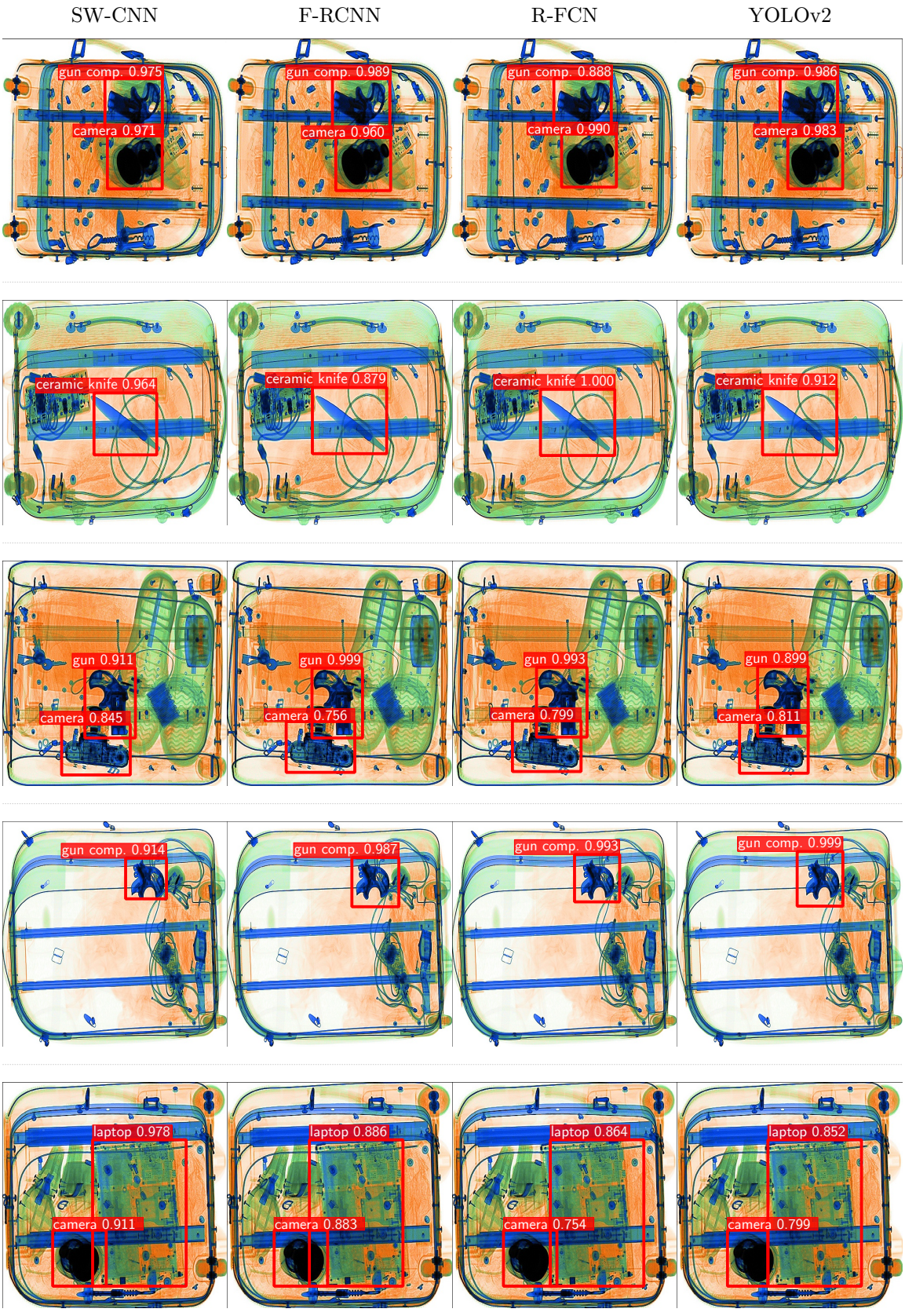


Figure 3.15: Easy examples detected by all of the detection approaches trained using ResNet_{101}. Columns: SW-CNN, Faster RCNN, R-FCN and YOLOv2.

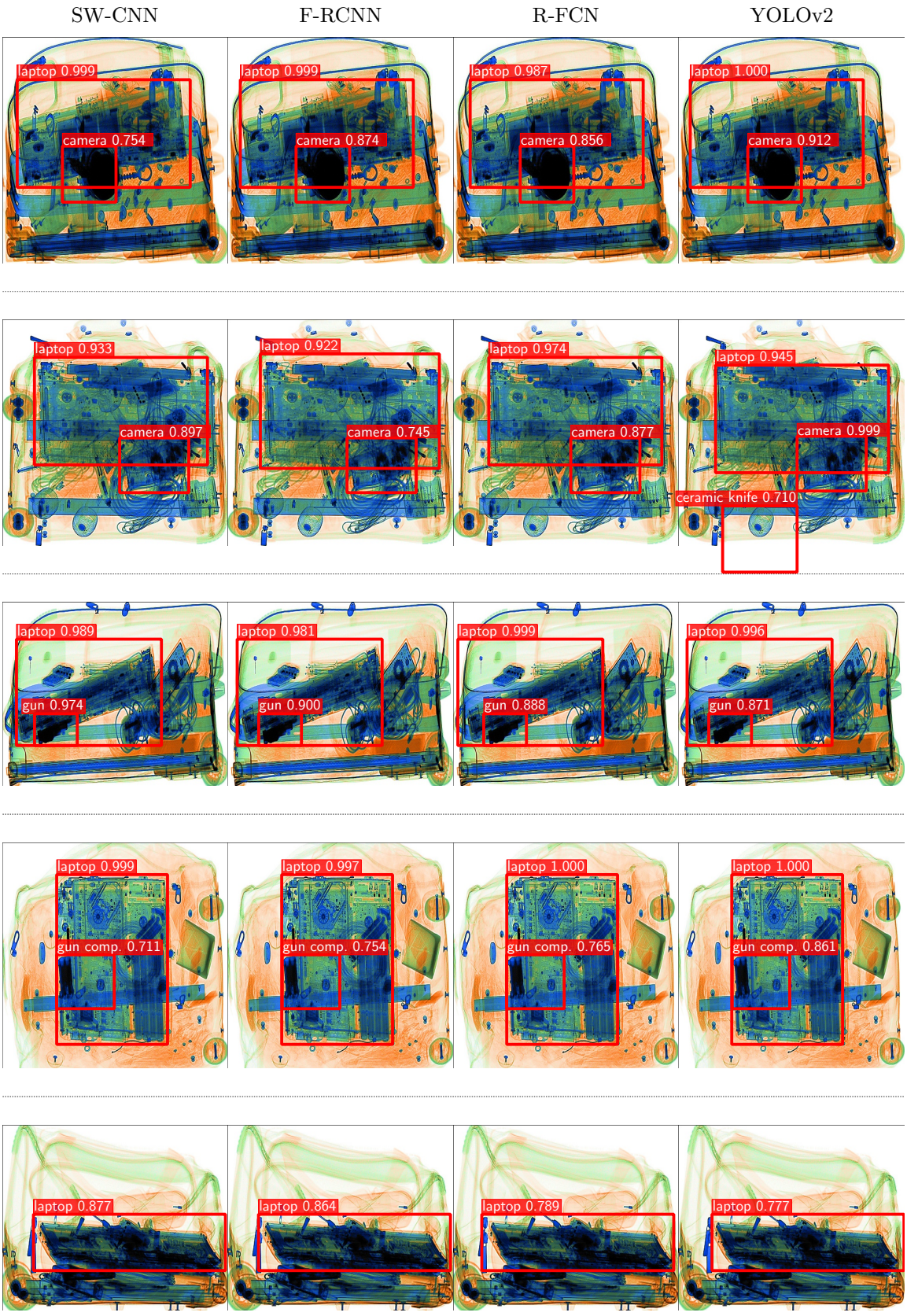


Figure 3.16: Moderate examples detected by all of the detection approaches trained using ResNet_{101}. Columns: SW-CNN, Faster RCNN, R-FCN and YOLOv2.

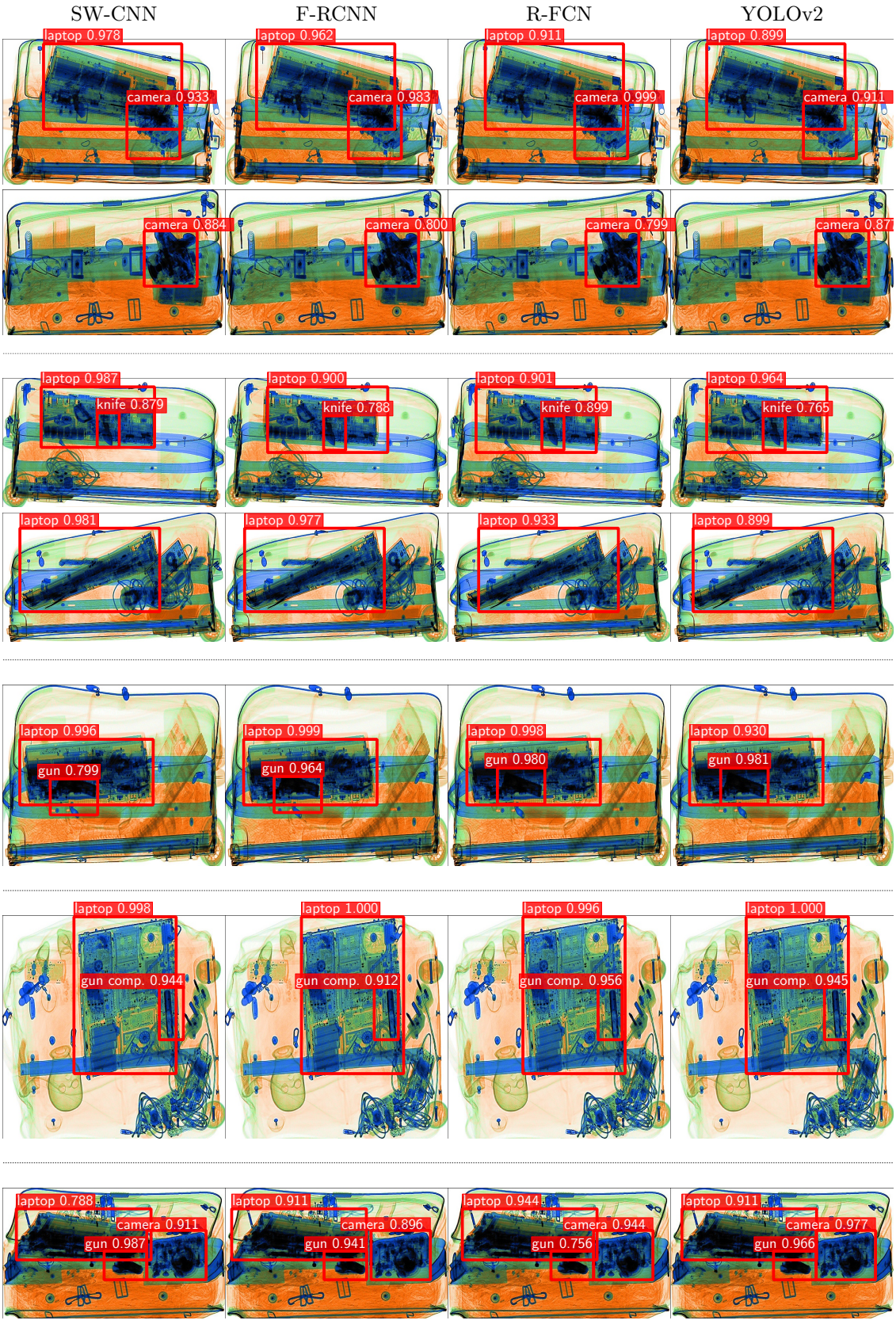


Figure 3.17: Difficult examples detected by all of the detection approaches trained using ResNet_{101}. Columns: SW-CNN, Faster RCNN, R-FCN and YOLOv2.

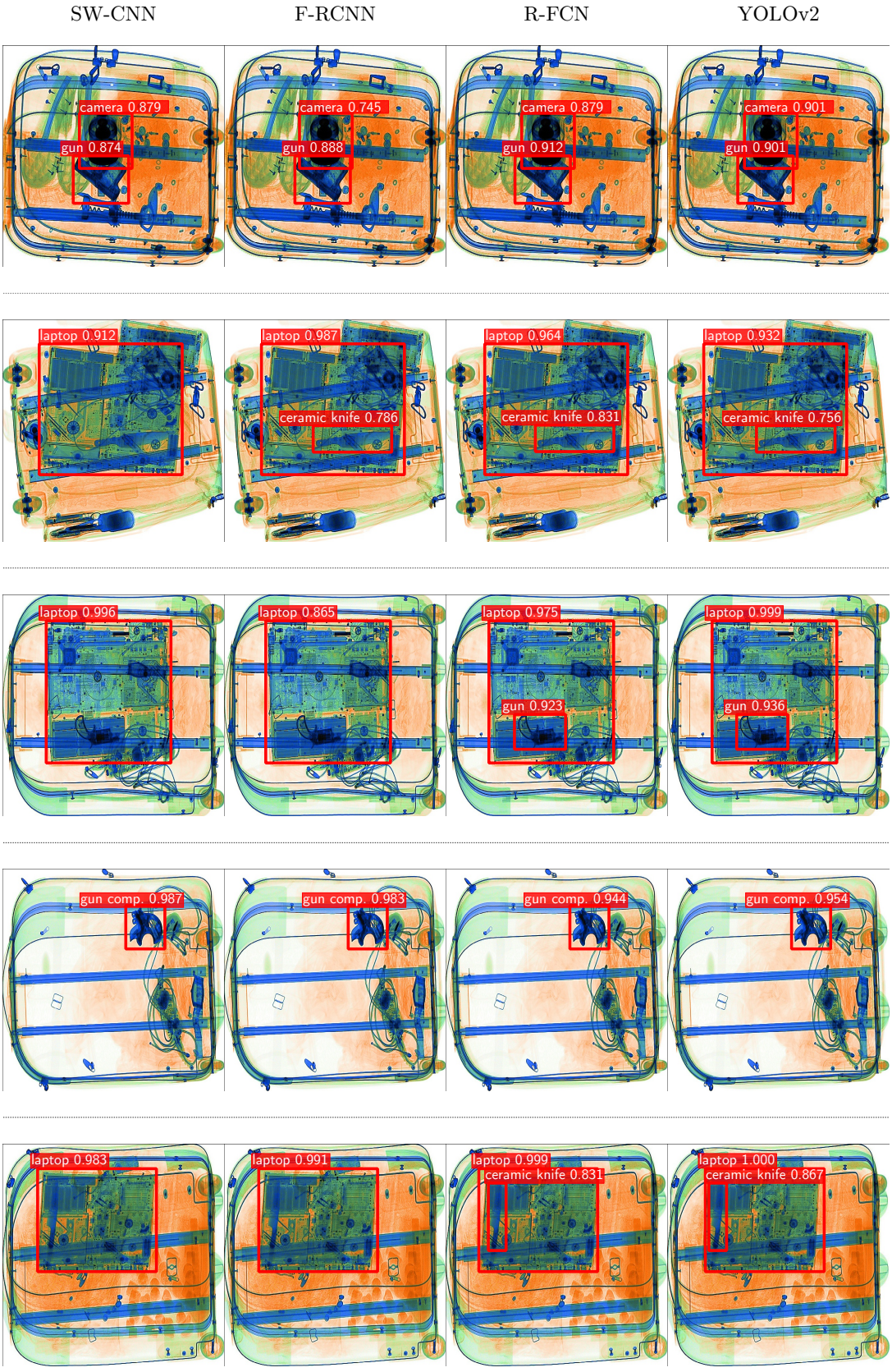


Figure 3.18: Easy examples (mis)detected by some of the detection approaches trained using ResNet_{101}. Columns: SW-CNN, Faster RCNN, R-FCN and YOLOv2.

Figure 3.19: Moderate examples (mis)detected by some of the detection approaches trained using ResNet_{101}. Columns: SW-CNN, Faster RCNN, R-FCN and YOLOv2.

Figure 3.20: Difficult examples (mis)detected by some of the detection approaches trained using ResNet_{101}. Columns: SW-CNN, Faster RCNN, R-FCN and YOLOv2.

3.4 Conclusion

This chapter exhaustively explores the use of CNN in the tasks of classification and detection within X-ray baggage imagery. For the classification problem, we make a comparison between CNN and traditional BoVW approaches based on handcrafted features. To do so, we perform layer freezing to observe the relative performance of fixed and fine-tuned sets of CNN feature maps. In addition to this, we train an SVM classifier on top of the last layer of the network to have a consistent comparison between CNN and handcrafted features. We also explore various CNN to see the impact of network complexity on overall performance.

Experimentation demonstrates that CNN features achieve superior performance to handcrafted BoVW features. Fine-tuning the entire network for this problem yields 99.6% True Positive (TP), 0.011 False Positive (FP) and 99.4 accuracy (A), a significant improvement on the best performing handcrafted feature detector/descriptor (FAST/SURF, 0.830 TP, 0.033 FP, 0.940 A). For the classification of multiple X-ray baggage objects, ResNet-50 achieves 0.986 (A), clearly demonstrating the applicability of CNN within X-ray baggage imagery, and outperforming prior reported results in the field [8, 77–79, 84].

In addition to classification, we also study object detection strategies to improve the performance of cluttered datasets further, where classification techniques fail. Hence, we examine the relative performance of traditional sliding window driven detection with CNN model [99, 150] against contemporary region-based [10, 11, 16] and single forward-pass based [12] CNN variants. We show that contemporary Faster RCNN, R-FCN, and YOLOv2 approaches outperform SW-CNN, which is already empirically shown to outperform handcrafted features, regarding both speed and accuracy.

YOLOv2 yields 0.885 and 0.974 mAP over 6-class object detection and 2-class firearm detection problems, respectively. This result illustrates the real-time applicability and superiority of such integrated region based detection models within this X-ray security imagery context.

Despite their promising performance, classification and detection models presented in this chapter, are all supervised, requiring expensive data annotation and

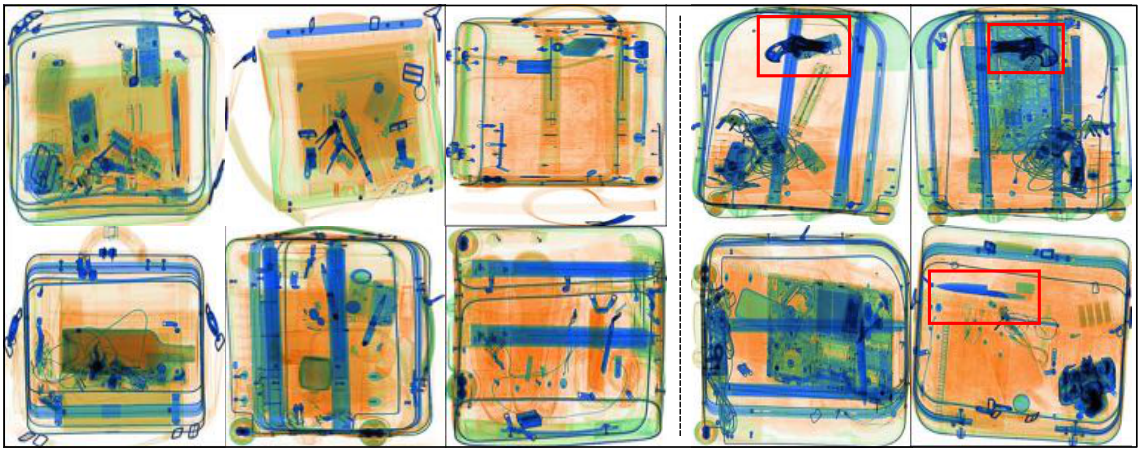
balanced datasets. Within the X-ray security screening context, however, available datasets are highly imbalanced such that the number of benign examples is significantly larger than threat samples. To cope with this class imbalance and data annotation issues, the next chapter investigates the use of unsupervised techniques to detect prohibited items within X-ray security imaging by considering them as generalised anomalies within the normal distribution of such imagery.

CHAPTER 4

GANomaly: Semi-Supervised Anomaly Detection via Adversarial Training

4.1 Introduction

Despite yielding encouraging performance over various computer vision tasks, as outlined in Chapter 3, supervised approaches heavily depend on large, labelled datasets. In many of the real-world problems, however, samples from the more unusual classes of interest are of insufficient sizes to be effectively modelled. Instead, the task of anomaly detection is to be able to identify such cases, by training only on samples considered to be *normal* and then identifying these unusual, insufficiently available samples (*abnormal*) that differ from the learned sample distribution of normality. For example, a tangible application, that is considered here within our evaluation, is that of X-ray screening for aviation or border security — where anomalous items posing a security threat are not commonly encountered, exemplary data of such can be difficult to obtain in any quantity, and the nature of any anomaly posing a potential threat may evolve due to a range of external factors. However, within this challenging context, human security operators are still competent and adaptable anomaly detectors against new and emerging anomalous threat signatures.



(a) Normal Data (X-ray Scans) (b) Normal + Abnormal Data (X-ray Scans)

Figure 4.1: Overview of our anomaly detection approach within the context of an X-ray security screening problem. Our model is trained on normal samples (a), and tested on normal and abnormal samples (b). Anomalies are detected when the output of the model is greater than a certain threshold $\mathcal{A}(x) > \phi$.

In general, in many situations, the availability of abnormal data samples is limited, and the representation of any available samples is merely a subset of all potential anomalous samples of that could be encountered within the deployment. This is a key challenge within anomaly detection.

As illustrated in Figure 4.1, a formal problem definition of the anomaly detection task is as follows: given a dataset $\mathcal{D}$ containing a large number of normal samples $\mathbf{X}$ for training, and relatively few abnormal examples $\hat{\mathbf{X}}$ for the test, a model f is optimized over its parameters θ. f learns the data distribution $p_{\mathbf{X}}$ of the normal samples during training while identifying abnormal samples as outliers during testing by outputting an anomaly score $\mathcal{A}(x)$, where x is a given test example. A larger $\mathcal{A}(x)$ indicates possible abnormalities within the test image since f learns to minimize this output score during training over the sets of normal examples. $\mathcal{A}(x)$ is general in that it can detect unseen anomalies as being non-conforming to $p_{\mathbf{X}}$.

There is a large volume of studies proposing anomaly detection models within various application domains [13, 155–158]. In addition, a considerable amount of work taxonomizes the approaches within the literature [159–163]. In parallel to the recent advances in this field, Generative Adversarial Networks (GAN) have emerged as a leading methodology across both unsupervised and semi-supervised problems. Goodfellow *et al.* [164] first proposed this approach by co-training a pair of networks

(generator and discriminator). The former network within this GAN formulation models high dimensional data from a latent vector to resemble the source data, while the latter network distinguishes the modelled (i.e., approximated) and original data samples. Several approaches followed this work to improve the training and inference stages [137, 165]. As reviewed in [158], adversarial training has also been adopted by recent work within anomaly detection.

Schlegl *et al.* [13] hypothesize that the latent vector of a GAN represents the true distribution of the data and remap to the latent vector by optimizing a pre-trained GAN based on the latent vector. The limitation is the enormous computational complexity of remapping to this latent vector space. In a follow-up study, Zenati *et al.* [14] train a BiGAN model [166], which maps from image space to latent space jointly, and report statistically and computationally superior results albeit on the simplistic MNIST benchmark dataset [167] (i.e. a leave one class out the formulation of handwritten digits recognition)

Motivated by [13, 14, 168], here we propose a generic anomaly detection architecture comprising an adversarial training framework. In a similar vein to [13], we use single-colour images as the input to our approach drawn only from an example set of *normal* (non-anomalous) training examples. However, in contrast, our approach does not require two-stage training and is both efficient for model training and later inference (run-time testing). As with [14], we also learn image and latent vector spaces jointly. Our key novelty comes from the fact that we employ adversarial autoencoder within an encoder-decoder-encoder pipeline, capturing the training data distribution within both image and latent vector space. An adversarial training architecture such as this, practically based on only *normal* training data examples, produces superior performance over challenging benchmark problems. The main contributions of this chapter are as follows:

- *semi-supervised anomaly detection* — a novel adversarial autoencoder within an encoder-decoder-encoder pipeline, capturing the training data distribution within both image and latent vector space, yielding superior results to contemporary GAN-based and traditional autoencoder-based approaches.

- *efficacy* — an efficient and novel approach to anomaly detection that yields

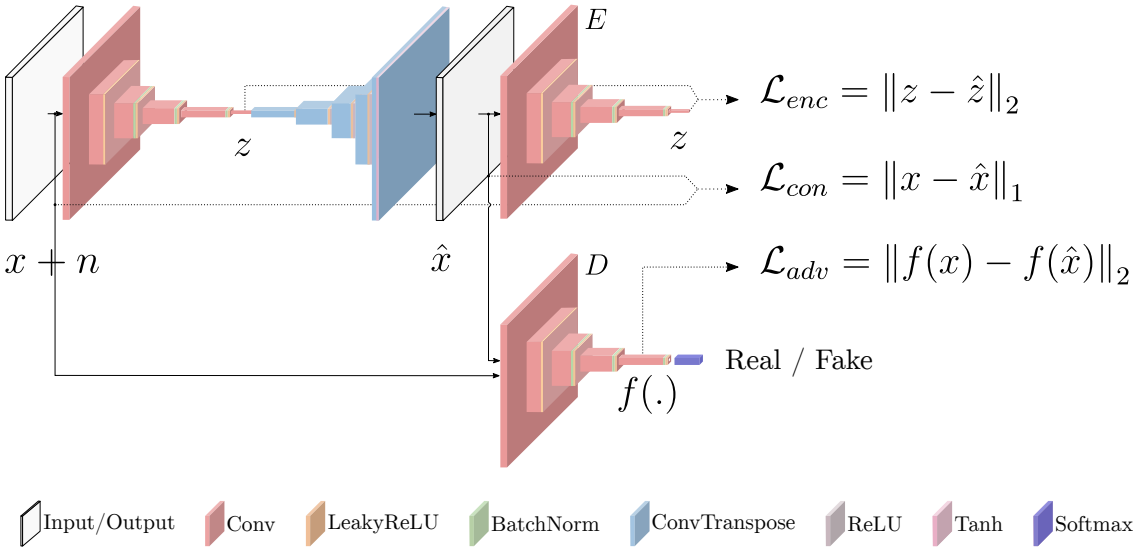


Figure 4.2: Pipeline of the proposed approach for anomaly detection.

both statistically and computationally better performance.

In addition, the chapter proposes a simple and effective algorithm such that the results could be reproduced via the code[1] made publicly available.

4.2 Our Approach: GANomaly

We denote our approach as GANomaly - the application of the GAN concept to anomaly detection via reconstructive error.

Problem Definition

Our objective is to train an unsupervised network that detects anomalies using a dataset that is highly biased towards a particular class - i.e., comprising *normal* non-anomalous occurrences only for training. The formal definition of this problem is as follows:

We are given a large training dataset $\mathcal{D}$ comprising only M normal images, $\mathcal{D} = \{X_1, \ldots, X_M\}$, and a smaller testing dataset $\hat{\mathcal{D}}$ of N normal and abnormal images, $\hat{\mathcal{D}} = \{(\hat{X}_1, y_1), \ldots, (\hat{X}_N, y_N)\}$, where $y_i \in [0, 1]$ denotes the image label. In

[1]The code is available on https://github.com/samet-akcay/ganomaly

the practical setting, the training set is significantly larger than the test set such that $M \gg N$.

Given the dataset, our goal is first to model $\mathcal{D}$ to learn its manifold, then detect the abnormal samples in $\hat{\mathcal{D}}$ as outliers to this manifold during the inference stage. The model f learns both the normal data distribution and minimizes the output anomaly score $\mathcal{A}(x)$. For a given test image $\hat{x}$, a high anomaly score of $\mathcal{A}(\hat{x})$ indicates possible anomalies within the image. The evaluation criteria for this is to threshold (ϕ) the score, where $\mathcal{A}(\hat{x}) > \phi$ indicates an anomaly instance.

Ganomaly Pipeline

Figure 4.2 illustrates the overview of our approach, which contains two encoders, a decoder, and discriminator networks, employed within three sub-networks.

First sub-network is a 'bow-tie' autoencoder network behaving as the generator part of the model. The generator learns the input data representation and reconstructs the input image via the use of an encoder and a decoder network, respectively. The formal principle of the sub-network is the following: the generator G first reads an input $x + n$, where $x \in \mathbb{R}^{w \times h \times c}$, n is a Gaussian noise with a random mean and standard deviation, and forward-passes it to its encoder network G_E. With the use of convolutional layers followed by batch-norm and leaky $ReLU()$ activation, respectively, G_E downscales x by compressing it to a vector z, where $z \in \mathbb{R}^d$. This vector, z, is also known as the bottleneck features of G and hypothesized to have the smallest dimension containing the best representation of the distribution from which x is drawn. The decoder part G_D of the generator network G adopts the architecture of a DCGAN generator [169], using convolutional transpose layers, $ReLU()$ activation and batch-norm together with a *tanh* layer at the end. This approach upscales the vector z to reconstruct the image x as $\hat{x}$. Based on these, the generator network G generates image $\hat{x}$ via $\hat{x} = G_D(z)$, where $z = G_E(x)$ (Figure 4.2 upper left).

The second sub-network is the encoder network E that compresses the image $\hat{x}$ that is reconstructed by the network G. With different parametrization, it has the same architectural details as G_E. E downscales $\hat{x}$ to find its feature representation $\hat{z} = E(\hat{x})$. The dimension of the vector $\hat{z}$ is the same as that of z for consistent

comparison. This sub-network is one of the unique parts of the proposed approach. Unlike the prior autoencoder-based approaches, in which the minimization of the latent vectors is achieved via the bottleneck features, this sub-network E explicitly learns to minimize the distance with its parametrization. During the test time, moreover, the anomaly detection is performed with this minimization (Figure 4.2 upper right).

The third sub-network is the discriminator network D whose objective is to classify the input x and the output $\hat{x}$ as real or fake, respectively. This sub-network is the standard discriminator network introduced in DCGAN [169] (Figure 4.2 lower right).

Having defined our overall multi-network architecture, as depicted in Figure 4.2, we now move on to discuss how we formulate our objective for learning.

4.2.1 Model Training

We hypothesize that when an abnormal image is forward-passed into the network G, G_D is not able to reconstruct the abnormalities even though G_E manages to map the input x to the latent vector z. This is because the network is modelled only on normal samples during training and its parametrization is not suitable for generating abnormal samples. An output $\hat{x}$ that has missed abnormalities can lead to the encoder network E mapping $\hat{x}$ to a vector $\hat{z}$ that has also missed abnormal feature representation, causing dissimilarity between z and $\hat{z}$. When there is such dissimilarity within latent vector space for an input image x, the model classifies x as an anomalous image. To validate this hypothesis, we formulate our objective function by combining three loss functions, each of which optimizes individual sub-networks.

Adversarial Loss

Following the current trend within the new anomaly detection approaches [13, 14], we also use feature matching loss for adversarial learning. Proposed by Salimans *et al.* [170], feature matching is shown to reduce the instability of GAN training. Unlike the vanilla GAN where G is updated based on the output of D (*real/fake*),

here we update G based on the internal representation of D. Formally, let f be a function that outputs an intermediate layer of the discriminator D for a given input x drawn from the input data distribution $p_{\mathbf{X}}$, feature matching computes the $\mathcal{L}_2$ distance between the feature representation of the original and the generated images, respectively. Hence, our adversarial loss of $\mathcal{L}_{adv}$ is defined as:

$$\mathcal{L}_{adv} = \mathbb{E}_{x \sim p_{\mathbf{X}}} \| f(x) - \mathbb{E}_{x \sim p_{\mathbf{X}}} f(G(x) \|_2. \tag{4.1}$$

Contextual Loss

The adversarial loss $\mathcal{L}_{adv}$ is adequate to fool the discriminator D with generated samples. However, with only an adversarial loss, the generator is not optimized towards learning contextual information about the input data. It has been shown that penalizing the generator by measuring the distance between the input and the generated images remedies this issue [138]. Isola *et al.* [138] show that the use of $\mathcal{L}_1$ yields less blurry reconstruction ($\hat{x}$) results than $\mathcal{L}_2$. Hence, we also penalize G by measuring the $\mathcal{L}_1$ distance between the original x and the generated images ($\hat{x} = G(x)$) using a contextual loss $\mathcal{L}_{con}$ defined as:

$$\mathcal{L}_{con} = \mathbb{E}_{x \sim p_{\mathbf{X}}} \| x - G(x) \|_1. \tag{4.2}$$

Encoder Loss

The two losses introduced above can enforce the generator to produce images that are not only realistic but also contextually sound. Moreover, we employ an additional encoder loss $\mathcal{L}_{enc}$ to minimize the distance between the bottleneck features of the input ($z = G_E(x)$) and the encoded features of the generated image ($\hat{z} = E(G(x))$). $\mathcal{L}_{enc}$ is formally defined as:

$$\mathcal{L}_{enc} = \mathbb{E}_{x \sim p_{\mathbf{X}}} \| G_E(x) - E(G(x)) \|_2. \tag{4.3}$$

In so doing, the generator learns how to encode features of the generated image for normal samples. For anomalous inputs, however, it will fail to minimize the distance

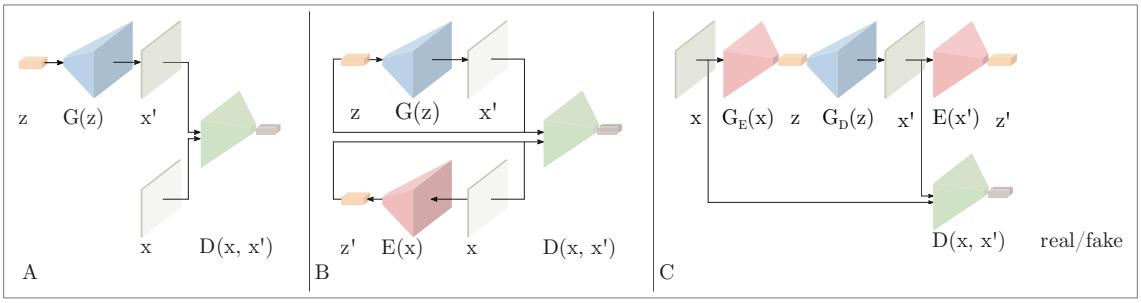


Figure 4.3: Comparison of the three models. A) AnoGAN [13], B) Efficient-GAN-Anomaly [14], C) Our Approach: GANomaly

between the input and the generated images in the feature space since both G and E networks are optimized towards normal samples only.

Overall, our objective function for the generator becomes the following:

$$\mathcal{L} = \lambda_{adv}\mathcal{L}_{adv} + \lambda_{con}\mathcal{L}_{con} + \lambda_{enc}\mathcal{L}_{enc} \tag{4.4}$$

where λ_{adv}, λ_{con} and λ_{enc} are the weighting parameters adjusting the impact of individual losses to the overall objective function.

4.2.2 Model Testing

During the test stage, the model uses $\mathcal{L}_{enc}$ given in Eq 4.3 for scoring the abnormality of a given image. Hence, for a test sample $\hat{x}$, our anomaly score $\mathcal{A}(\hat{x})$ or $s_{\hat{x}}$ is defined as:

$$\mathcal{A}(\hat{x}) = \|G_E(\hat{x}) - E(G(\hat{x}))\|_1. \tag{4.5}$$

To evaluate the overall anomaly performance, we compute the anomaly score for individual test sample $\hat{x}$ within the test set $\hat{\mathcal{D}}$, which in turn yields us a set of anomaly scores $\mathcal{S} = \{s_i : \mathcal{A}(\hat{x}_i), \hat{x}_i \in \hat{\mathcal{D}}\}$. We then apply feature scaling to have the anomaly scores within the probabilistic range of $[0, 1]$.

$$s'_i = \frac{s_i - min(\mathcal{S})}{max(\mathcal{S}) - min(\mathcal{S})} \tag{4.6}$$

The use of Eq 5.6 ultimately yields an anomaly score vector $\mathcal{S}'$ for the final evaluation of the test set $\hat{\mathcal{D}}$.

4.3 Experimental Setup

To evaluate our anomaly detection framework, we use three types of dataset ranging from the simplistic benchmark of MNIST [167], the reference benchmark of CIFAR [171] and the operational context of anomaly detection within X-ray security screening [42].

4.3.1 Datasets

MNIST

To replicate the results presented in [14], we first experiment on MNIST data [167] by treating one class being an anomaly, while the rest of the classes are considered as the normal class. In total, we have ten sets of data, each of which considers individual digits as the anomaly.

CIFAR10

Within our use of the CIFAR dataset [171], we again treat one class as abnormal and the rest as normal. We then detect the outlier anomalies as instances drawn from the former class by training the model on the latter labels.

University Baggage Anomaly Dataset — (UBA)

This sliding window patched-based dataset comprises 230,275 image patches. Normal samples are extracted via an overlapping sliding window from a full X-ray image, constructed using single conventional X-ray imagery with associated false-colour materials mapping from dual-energy [18]. Abnormal classes $(122,803)$ are of 3 sub-classes — knife $(63,496)$, gun $(45,855)$ and gun component $(13,452)$ — contain manually cropped threat objects together with sliding window patches whose intersection over union with the ground truth is greater than 0.3.

Full Firearm vs. Operational Benign — (FFOB)

In addition to these datasets, we also use the UK government evaluation dataset [15], comprising both expertly concealed firearm (threat) items and operational benign

(non-threat) imagery from commercial X-ray security screening operations (baggage/parcels). Denoted as FFOB, this dataset comprises $4,680$ firearm full-weapons as full abnormal and $67,672$ operational benign as full normal images, respectively.

4.3.2 Implementational Details

The procedure for train and test set split for the above datasets is as follows: we split the normal samples such that 80% and 20% of the samples are considered as part of the train and test sets, respectively. We then resize MNIST to 32×32, DBA and FFOB to 64×64, respectively.

Following Schlegl *et al.* [13] (AnoGAN) and Zenati *et al.* [14] (EGBAD), our adversarial training is also based on the standard DCGAN approach [169] for a consistent comparison. As such, we aim to show the superiority of our multi-network architecture regardless of using any tricks to improve the GAN training. In addition, we also compare our method against the traditional variational autoencoder architecture [168] (VAE) to show the advantage of our multi-network architecture. We implement our approach in PyTorch [172] (v1.2.0 with Python 3.7.4) by optimizing the networks using Adam [173] with an initial learning rate $lr = 2e^{-3}$, and momentums $\beta_1 = 0.5$, $\beta_2 = 0.999$. Our model is optimized based on the weighted loss $\mathcal{L}$ (defined in Equation 4.4) using the weight values $\lambda_{adv} = 1$, $\lambda_{con} = 50$ and $\lambda_{enc} = 1$, which were empirically chosen to yield optimum results. (Figure 4.5 (b)). We train the model for 15, 25, 25 epochs for MNIST, UBA and FFOB datasets, respectively. Experimentation is performed using a dual-core Intel Xeon E5-2630 v4 processor and NVIDIA GTX Titan X GPU.

4.4 Results

We report results based on the area under the curve (AUC) of the Receiver Operating Characteristic (ROC), true positive rate (TPR) as a function of false-positive rate (FPR) for different points, each of which is a TPR-FPR value for different thresholds.

Figure 4.4 (a) presents the results obtained on MNIST data using three different random seeds, where we observe the clear superiority of our approach over previous

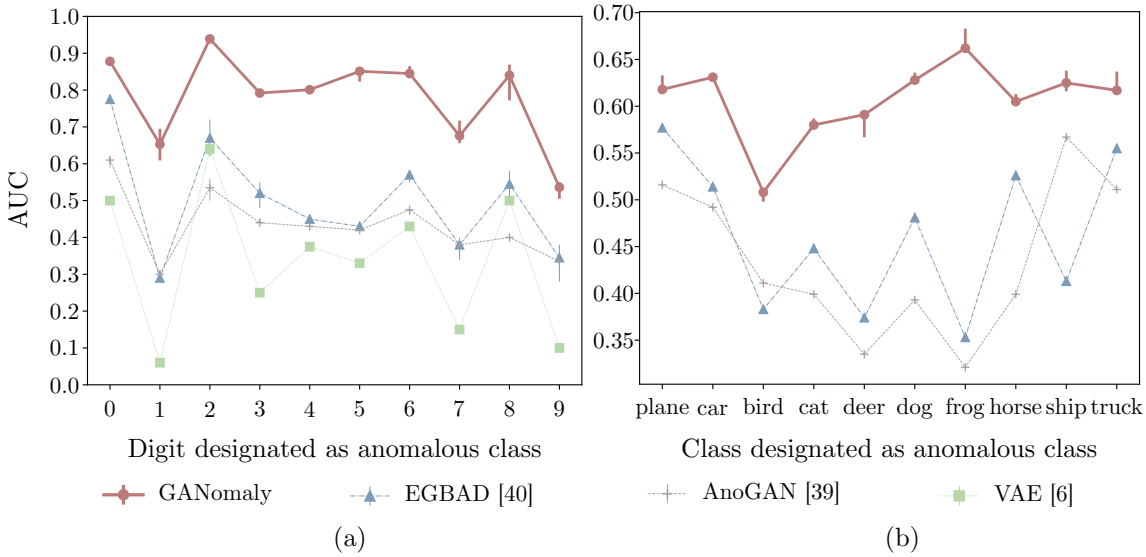


Figure 4.4: Results for MNIST (a) and CIFAR (b) datasets. Variations due to the use of 3 different random seeds are depicted via error bars. All but GANomaly results in (a) were obtained from [14].

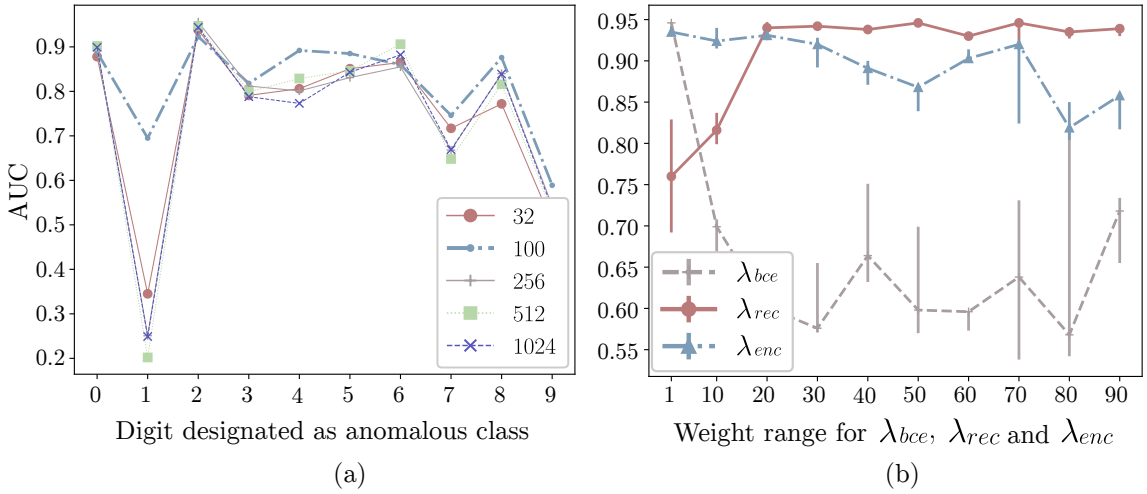
	UBA				FFOB
Method	gun	gun-parts	knife	overall	full-weapon
AnoGAN [13]	0.598	0.511	**0.599**	0.569	0.703
EGBAD [14]	0.614	0.591	0.587	0.597	0.712
GANomaly	**0.747**	**0.662**	0.520	**0.643**	**0.882**

Table 4.1: AUC results for UBA and FFOB datasets

contemporary models [13, 14, 168]. For each digit chosen as anomalous, our model achieves higher AUC than EGBAD [14], AnoGAN [13] and variational autoencoder pipeline VAE [168]. Due to showing its poor performance within a relatively unchallenging dataset, we do not include VAE in the rest of the experiments.

Figure 4.4 (b) shows the performance of the models trained on the CIFAR10 dataset. We see that our model achieves the best AUC performance for any of the class chosen as anomalous. The reason for getting relatively lower quantitative results within this dataset is that for a selected abnormal category, there exists a normal class that is similar to the abnormal (plane vs bird, cat vs dog, horse vs deer and car vs truck).

For UBA and FFOB datasets shown in Table 5.2, our model again outperforms other approaches excluding the case of the *knife*. The performance of the models for *knife* is comparable. The relatively lower performance of this class is its shape



(a) (b)

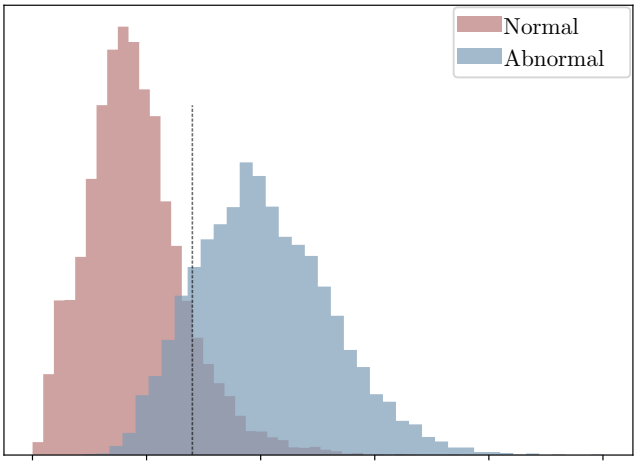
Figure 4.5: (a) Overall performance of the model based on varying size of the latent vector z. (b) Impact of weighting the losses on the overall performance. Model is trained on MNIST dataset with an abnormal digit-2

simplicity, causing an overfit and hence high false positives. For the overall performance, however, our approach surpasses the other models, yielding AUC of 0.666 and 0.882 on the UBA and FFOB datasets, respectively.

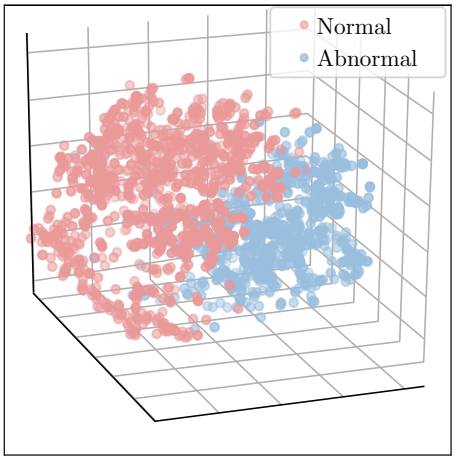
Figure 4.5 depicts how the choice of hyper-parameters ultimately affect the overall performance of the model. In Figure 4.5 (a), we see that the optimal performance is achieved when the size of the latent vector z is 100 for the MNIST dataset with an abnormal digit-2. Figure 4.5 (b) demonstrates the impact of tuning the loss function in Equation 4.4 on the overall performance. The model achieves the highest AUC when $\lambda_{bce} = 1$, $\lambda_{rec} = 50$ and $\lambda_{enc} = 1$. We empirically observe the same tuning-pattern for the rest of datasets.

Figure 4.6 provides the histogram of the anomaly scores during the inference stage (a) and t-SNE visualization of the features extracted from the last convolutional layer of the discriminator network (b). Both of the figures demonstrate a clear separation within the latent vector z and feature $f(.)$ spaces.

Table 4.2 illustrates the runtime performance of the GAN-based models. Compared to the rest of the approaches, AnoGAN [13] is computationally rather expensive since the optimization of the latent vector is needed for each example. For EGBAD [14], we report similar run-time performance to that of the original paper. Our approach, on the other hand, achieves the highest run-time performance. Run-



(a)



Perplexity: 40, LR: 140, Iter: 1000

(b)

Figure 4.6: (a) Histogram of the scores for both normal and abnormal test samples. (b) t-SNE visualization of the features extracted from the last conv. layer $f(.)$ of the discriminator

Model	MNIST	CIFAR	DBA	FFOB
AnoGAN [13]	7120	7120	7110	7223
EGBAD [14]	8.92	8.71	8.88	8.87
GANomaly	**2.79**	**2.21**	**2.66**	**2.53**

Table 4.2: Computational performance of the approaches. (Runtime in terms of millisecond)

time performance of both UBA and FFOB datasets are comparable to MNIST even though their image and network size are double than that of MNIST.

A set of examples in Figures 4.7, 4.8, 4.9 and 4.10 depict real and fake images that are respectively the input and output of our model. Left and right columns show benign and anomaly instances, respectively. Each column shows three image pairs (real vsreconstructed) for MNIST and CIFAR-10 and two pairs for DBA and FFOB datasets. Here, we qualitatively evaluate the performance of the model by checking the reconstruction error and expect the model to produce large reconstruction error for abnormal examples.

Figure 4.7 demonstrate MNIST examples, where two consecutive row shows the input and reconstructed samples for benign and abnormal digit. For the first two rows, for instance, the model is trained to detect the abnormal digit-0. As can be seen from the figure, the reconstruction error is low for both benign and abnormal samples, which contradicts to our hypothesis. This is due to the unchallenging

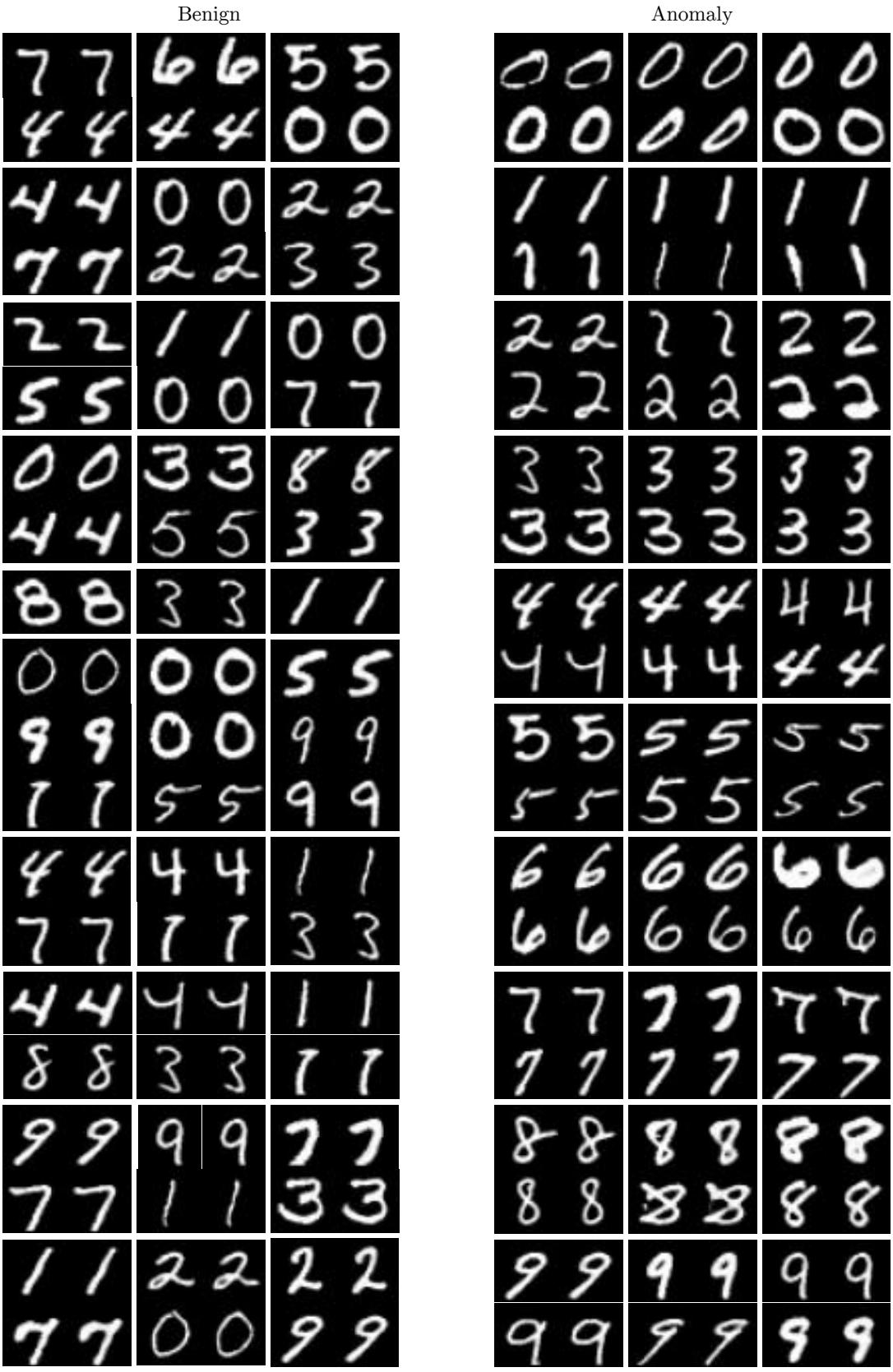


Figure 4.7: Randomly selected real and generated samples containing normal and abnormal objects in MNIST dataset. The model is capable of generating abnormal samples; and detecting the abnormality within the latent vector space.

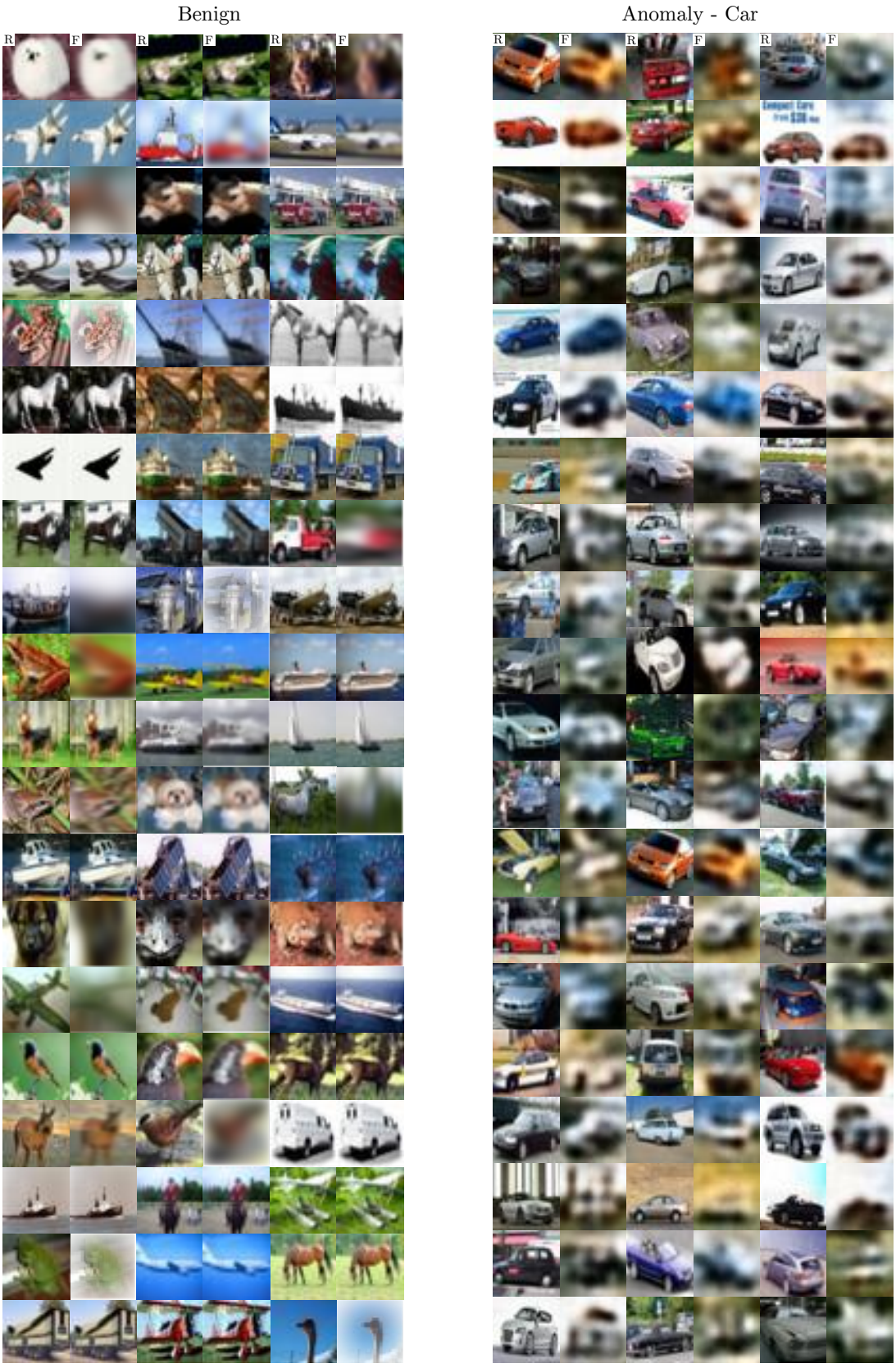


Figure 4.8: Randomly selected real and generated samples containing normal and abnormal objects in CIFAR dataset. The model fails to generate abnormal samples not being trained on.

Figure 4.9: Randomly selected real and generated samples containing normal and abnormal objects in DBA dataset. The model fails to generate abnormal samples not being trained on.

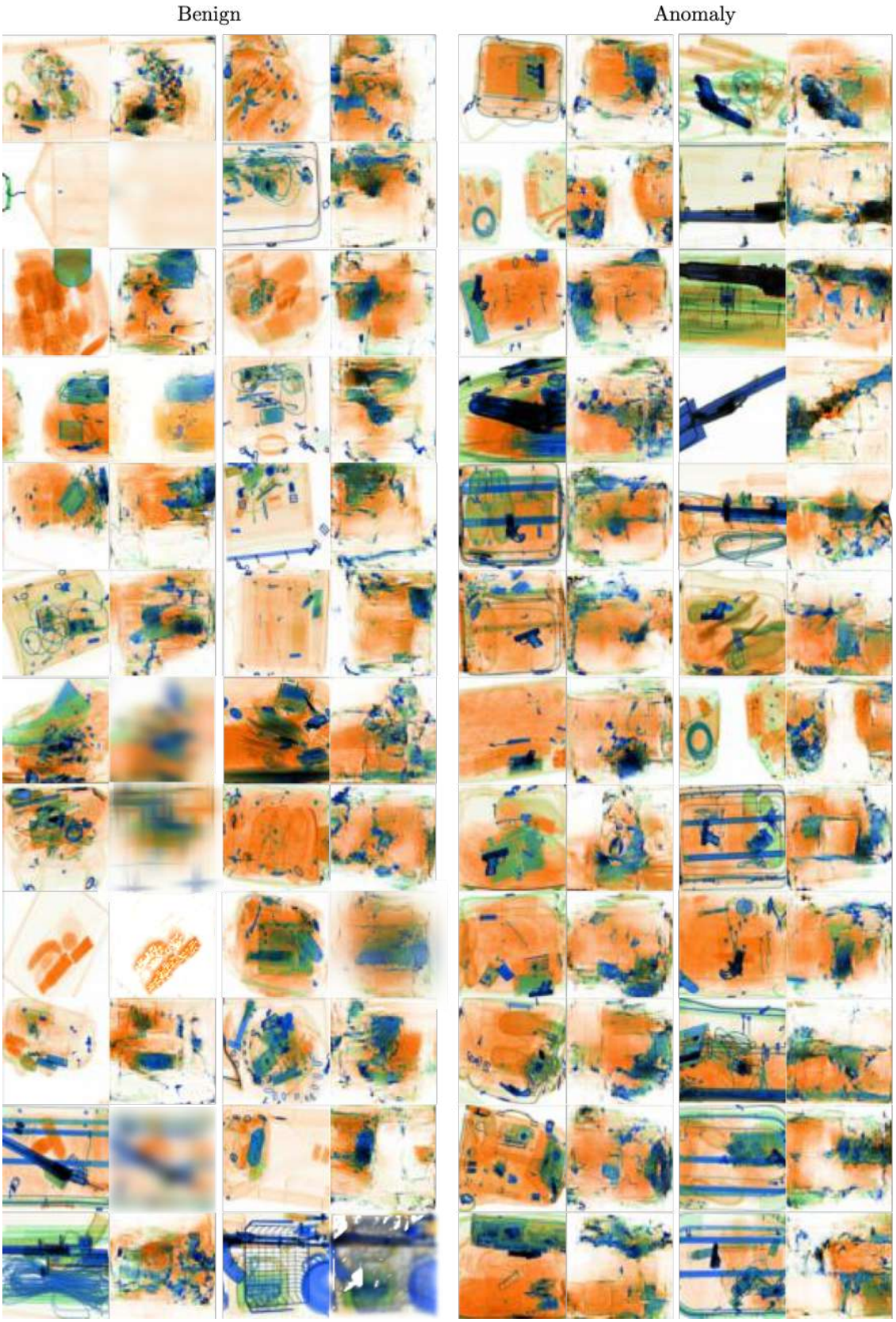


Figure 4.10: Randomly selected real and generated samples containing normal and abnormal objects in FFOB dataset. The model fails to generate abnormal samples not being trained on.

nature of MNIST, where the model learns adequate information to generate the classes not seen during training. Despite this low reconstruction error, the model is capable of detecting abnormality within its latent space.

Figure 4.8 depicts the reconstruction performance of the model for CIFAR-10 dataset, where the *car* class is chosen as the abnormal class. The figure demonstrates that the model can reconstruct benign images, while struggles to generate the abnormal class. Closer inspection to the fifth row in the figure, for instance, shows that the reconstruction error for truck, *ship* and *horse* classes are rather low, while being large for *car* samples. The figure shows the model's potential to detect abnormalities.

Similar to the case of CIFAR-10, Figure 4.9 represents the test images for UBA dataset. It is apparent that the reconstruction error for benign samples are low (Figure 4.9 left), while being large for the abnormal instances (Figure 4.9 right). Figure 4.10 demonstrates similar patterns, where the model's capability to reconstruct abnormal examples are rather limited. Unlike Figure 4.9, the reconstruction error for benign samples are relatively higher than that of Figure 4.9. This is because, being cluttered full X-ray images, FFOB is a more challenging dataset than UBA, which consists of X-ray image patches. Despite this complexity of the FFOB dataset, the model still copes well with detecting the abnormality.

Overall, these results purport that our approach yields both statistically and computationally superior results than leading state-of-the-art approaches [13, 14].

4.5 Conclusion

Despite achieving superior performance, supervised CNN-based object classification and detection methods depend on large, annotated and balanced datasets. Within the X-ray security screening context, however, anomalous objects are not commonly encountered, exemplary data of such can be challenging to obtain, and the nature of the abnormality may evolve due to a range of external factors. Such issues weaken the generalizability of the supervised CNN models and hence limits their use within any deployment.

To tackle the issues stated above, this chapter proposes an unsupervised anomaly detection algorithm that captures the distribution of the normal samples during training. Within the inference, the model detects the abnormality based on the deviation of the samples from the distribution of the normal data examples. The proposed model utilizes a novel adversarial training scheme such that the generator network comprises an encoder-decoder-encoder architectural model for superior data capturing and reconstruction. Experimentation across different benchmarks of varying complexity such as [167, 171], and within the operational anomaly detection context of X-ray security screening [15, 49], shows that the proposed method outperforms both contemporary state-of-the-art GAN-based [13, 14] and traditional autoencoder-based anomaly detection approaches [60] with generalization ability to any anomaly detection task.

Despite its superior performance improvement over the state-of-the-art techniques, there are certain limitations of the proposed model. As shown in Figures 4.7, 4.8, 4.9 and 4.10, the model suffers from huge reconstruction error such that the quality of the generated samples is worse than the original input images. In addition to this reconstruction issue, there is also network redundancy within the network pipeline that the encoder $E()$ and the discriminator $D()$ networks have different parametrization despite having the same architecture. By addressing these issues, the next chapter introduces a new unsupervised anomaly detection, which overall yields superior detection performance.

CHAPTER 5

Skip-GANomaly: Skip Connected and Adversarially Trained Encoder-Decoder Anomaly Detection

5.1 Introduction

Chapter 4 introduces a novel, unsupervised anomaly detection method with adversarial training that outperforms the previous state-of-the-art [13, 14]. Despite this significant gain, the proposed model has the following limitations: (i) incapability of generating high-quality normal images, (ii) parameter redundancy due to having two exact same network with different parametrization (See Figure 4.2).

Motivated by the promising performance and limitations of the model outlined in Chapter 4, this chapter introduces a new method for anomaly detection via adversarial training. The proposed model addresses the twofold issues of Chapter 4: first, to alleviate the high reconstruction issue, the model utilizes skip-connected encoder-decoder (convolutional neural) network architecture. While adversarial training has shown the promise of GAN in this domain [49], skip-connections within such UNet-style (encoder-decoder) [174] generator networks are known to enable the multi-scale capture of image space detail with sufficient capacity to generate high-quality normal images drawn from the distribution the model has learned. Second, the model

(a) (b)

Figure 5.1: Sub-sample of the X-ray screening application dataset used to train the proposed approach: (a) training data contains normal samples only, while the test data (b) comprises both normal and abnormal samples.

also tackles with the parameter redundancy issue by learning latent representation within the discriminator network. Similar to [13, 14, 49], the proposed approach also seeks to learn the normal distribution in both the image and latent spaces via a GAN generator-discriminator paradigm. The discriminator network not only forces the generator to learn an improved model of the distribution but also works as a feature extractor such that it learns the reconstruction of the normal distribution within a higher-dimensional latent space. Evaluation of the model on various established benchmarks [15, 171] statistically illustrates superior anomaly detection task performance over prior work by Schlegl *et al.* [13], Zenati *et al.* [14] and GANomaly [49] (Chapter 4). Subsequently, the main contributions of this chapter are as follows:

- *high-quality reconstruction* — a generator network utilising skip-connected encoder-decoder convolutional network architecture that produces high-quality images and that eliminates high-reconstruction errors reported in Chapter 4.

- *unique latent-representation learning* — a discriminator network that is capable of both identifying real *vs.* fake images and learning latent representation for normal and abnormal distributions, which overall eliminates the parameter redundancy issue outlined in Chapter 4.

- *efficacy* — an efficient anomaly detection algorithm achieving quantitatively and qualitatively superior performance against prior state-of-the-art approaches [13, 14, 49].

In addition, the chapter proposes a simple and effective algorithm such that the results could be reproduced via the code[1] made publicly available.

5.2 Proposed Approach

Before proceeding to explain our proposed approach, it is important to introduce the fundamental concepts.

Problem Definition

This work proposes an unsupervised approach for anomaly detection.

We adversarially train our proposed convolutional network architecture in an unsupervised manner such that the conceptual model is trained on normal samples only, and yet tested on both normal and abnormal ones. Mathematically, we define and formulate our problem as the following:

As also discussed in Chapter 4, we are given a large training set $\mathcal{D}$ and a test set $\hat{\mathcal{D}}$ such that $\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$ contains m normal samples, where $y_i = 0$ denotes the normal class. The test set $\hat{\mathcal{D}} = \{(x_1, y_1), (x_2, y_2), ..., (x_m, y_m)\}$ comprises n normal and abnormal samples, where $y_i \in [0, 1]$ for normal and abnormal classes, respectively. In practical settings, $m \gg n$.

Based on the dataset defined above, we train our model f on $\mathcal{D}$ and evaluate its performance on $\hat{\mathcal{D}}$. The training objective ($\mathcal{J}$) of the model f is to capture the distribution of $\mathcal{D}$ within not only image space but also hidden latent vector space.

[1]The code is available on https://github.com/samet-akcay/skip-ganomaly

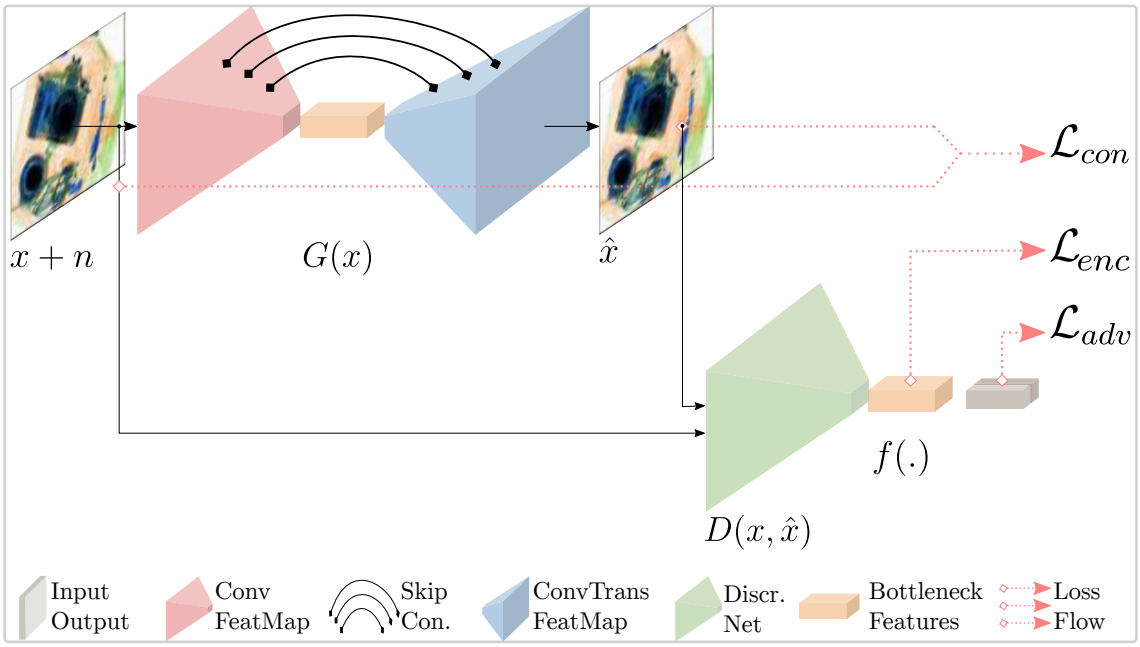


Figure 5.2: Overview of the proposed adversarial training procedure.

Capturing the distribution within both dimensions by minimizing $\mathcal{J}$ enables the network to learn higher and lower level features that are unique to normal images. We hypothesize that defining an anomaly score $\mathcal{A}(.)$ based on the training objective $\mathcal{J}$ would yield minimal anomaly scores for training samples —*normal samples*, but greater scores for abnormal images. Hence a higher anomaly score $\mathcal{A}(x)$ for a given sample x would indicate whether x is normal or abnormal concerning the distribution of normal data learned by f from $\mathcal{D}$ during training.

Pipeline

Similar to GANomaly [49] pipeline described in Section 4.2, the proposed approach comprises a generator (G) and a discriminator (D) network, as depicted in Figure 5.2. Unlike GANomaly [49] that utilizes an encoder-decoder-encoder generator network, this work adopts a bow-tie architecture for the network G by using an encoder (G_E) and a decoder (G_D) network. The encoder network captures the distribution of the input data by mapping the image ($x + n$) into lower-dimensional latent representation (z) such that $G_E : x + n \rightarrow z$, where $x \in \mathbb{R}^{w \times h \times c}$, n is a Gaussian noise with random mean and standard deviation and $z \in \mathbb{R}^d$. As illustrated in Figure 5.3, the network G_E reads input x through five blocks containing Convolutional and

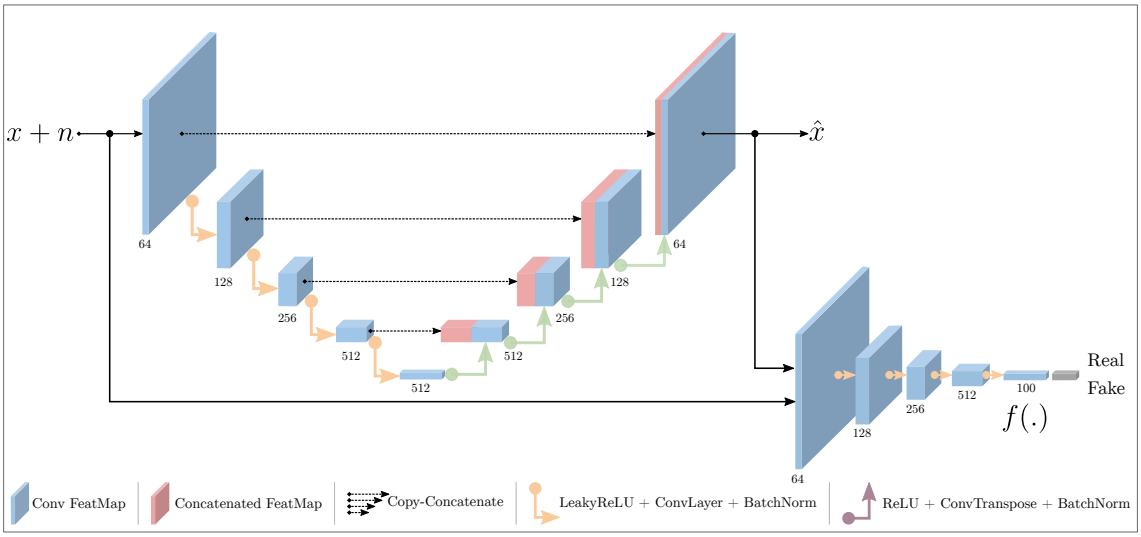


Figure 5.3: Details of the proposed network architecture.

BatchNorm layers as well as LeakyReLU activation function and outputs the latent representation z, which is also known as the bottleneck features that carries a unique representation of the input.

Being symmetrical to G_E, the decoder network G_D up-samples the latent vector z back to the input image dimension and reconstructs the output, denoted as $\hat{x}$. In contrast to GANomaly [49], here the decoder G_D adopts skip-connection approach such that each down-sampling layer in the encoder network is concatenated to its corresponding up-sampling decoder layer (Figure 5.3). This use of skip connections provides substantial advantages via direct information transfer between the layers, preserving both local and global (multi-scale) information, and hence yielding better reconstruction.

The second network within the pipeline, shown in Figure 5.3 (b), called discriminator (D), predicts the class label of the given input. Like GANomaly [49], its initial task is to classify real images (x) from the fake ones ($\hat{x}$), generated by the network G. The network architecture of the discriminator D follows the same structure as the discriminator of the DCGAN approach presented in [169]. In addition to being a classifier, the network D is also used as a feature extractor such that the latent representation of the input image x and the reconstructed image $\hat{x}$ is computed. Learning the latent space representation within the network D is another novelty of the proposed approach compared to the previous approaches [13, 14, 49].

Based on this multi-network architecture, explained above and shown in Figure 5.3, the next section describes the proposed training objective and inference scheme.

5.2.1 Training Objective

As explained in Section 5.2, the idea proposed in this work is to train the model only on normal samples, and test on both normal and abnormal ones. Similar to the one explained in Section 4.2.1, the motivation is that we expect the model to be able to correctly reconstruct the normal samples either in image or latent vector space. The hypothesis is that the network is conversely expected to fail to reconstruct the abnormal samples as it is never trained on such abnormal examples. Hence, for abnormal samples, one would expect a higher loss for the reconstruction of the output image $\hat{x}$ or the latent representation $\hat{z}$. To validate this, we propose to combine three loss values (*Adversarial, Contextual, Latent*—Encoder), each of which has its contribution to make within the overall training objective.

Adversarial Loss

Unlike GANomaly [49] that uses feature matching loss [170], this model utilises the adversarial loss [164] to maximize the reconstruction capability for the normal images x during training. This loss, shown in Equation 5.1, ensures that the network G reconstructs a normal image x to $\hat{x}$ as realistically as possible, while the discriminator network D classifies the real and the (fake) generated samples. The task here is to minimize this objective for G, and maximize for D to achieve $\min_G \max_D \mathcal{L}_{adv}$, where $\mathcal{L}_{adv}$ is denoted as

$$\mathcal{L}_{adv} = \mathop{\mathbb{E}}_{x \sim p_x} [\log D(x)] + \mathop{\mathbb{E}}_{x \sim p_x} [\log(1 - D(\hat{x})]. \tag{5.1}$$

Contextual Loss

The adversarial loss defined in Section 5.2.1 forces the model to generate realistic samples but does not guarantee to learn contextual information regarding the input. As proposed for GANomaly [49] in Section 4.2.1, we apply an L_1 loss between the input x and the reconstructed output $\hat{x}$ to explicitly learn this contextual information

to sufficiently capture the input data distribution for the normal samples. This loss component ensures that the model is capable of generating contextually similar images to normal samples. The contextual loss of the training objective is shown below:

$$\mathcal{L}_{con} = \mathop{\mathbb{E}}_{x \sim p_x} ||x - \hat{x}||_1. \tag{5.2}$$

Latent Loss

With the adversarial and contextual losses defined above, the model can generate realistic and contextually similar images. In addition to these objectives, we aim to reconstruct latent representations for the input x and the generated normal samples $\hat{x}$ as similar as possible. This is to ensure that the network is capable of producing contextually sound latent representations for common examples.

Unlike GANomaly [49] that minimises the latent representation by taking the $\mathcal{L}_2$ norm of the bottleneck features of the input ($z = G_E(x)$) and the encoded features of the generated image ($\hat{z} = E(\hat{x})$), this model uses the final convolutional layer of the discriminator D, and extract the features of x and $\hat{x}$ to reconstruct their latent representations as $z = f(x)$ and $\hat{z} = f(\hat{x})$ (See Figures 4.2 and 5.2). The latent representation loss therefore becomes:

$$\mathcal{L}_{enc} = \mathop{\mathbb{E}}_{x \sim p_x} ||f(x) - f(\hat{x})||_2. \tag{5.3}$$

Finally, total training objective becomes a weighted sum of the losses above.

$$\mathcal{L} = \lambda_{adv}\mathcal{L}_{adv} + \lambda_{con}\mathcal{L}_{con} + \lambda_{enc}\mathcal{L}_{enc}, \tag{5.4}$$

where λ_{adv}, λ_{con} and λ_{enc} are the weighting parameters adjusting the dominance of the individual loss components within the overall objective function.

5.2.2 Inference

To find the anomalies during the testing and subsequent deployment, we adopt the anomaly score, proposed in [13] and also employed in [14]. For a given test image

$\dot{x}$, its anomaly score becomes:

$$\mathcal{A}(\dot{x}) = \lambda R(\dot{x}) + (1 - \lambda)L(\dot{x}), \tag{5.5}$$

where $R(\dot{x})$ is the reconstruction score measuring the contextual similarity between the input and the generated images based on Equation 5.2. $L(\dot{x})$ denotes the latent representation score measuring the difference between the input and generated images based on Equation 5.3. λ is the weighting parameter controlling the relative importance of the score functions.

Based on Equation 5.5, we then compute the anomaly scores for each individual test sample $\dot{x}$ in the test set $\hat{\mathcal{D}}$, and denote as anomaly score vector $\boldsymbol{A}$ such that $\boldsymbol{A} = \{A_i : \mathcal{A}(\dot{x}_i), \dot{x}_i \in \hat{\mathcal{D}}\}$. Finally, following the same procedure proposed in [49], we also apply feature scaling to $\boldsymbol{A}$ to scale the anomaly scores within the probabilistic range of $[0, 1]$. Hence, the updated anomaly score for an individual test sample $\dot{x}$ becomes:

$$\hat{\mathcal{A}}(\dot{x}) = \frac{\mathcal{A}(\dot{x}) - min(\boldsymbol{A})}{max(\boldsymbol{A}) - min(\boldsymbol{A})}. \tag{5.6}$$

Equation 5.6 finally yields an anomaly score vector $\hat{\boldsymbol{A}}$ for the final evaluation of the test set $\hat{\mathcal{D}}$, which is explained in Sections 5.3.3 and 5.4.

5.3 Experimental Setup

This section introduces the datasets, training and implementational details as well as the evaluation criteria used within the experimentation.

5.3.1 Datasets

To demonstrate the proof of concept of the proposed approach, we follow the same experimental setup presented in Chapter 4 and validate our model on CIFAR-10 [171], UBA [49] and FFOB [15] datasets.

5.3.2 Training Details

The training loss $\mathcal{L}$ from Equation 5.4 is optimized via Adam [173] optimizer with an initial learning rate $lr = 2e^{-3}$ with a lambda decay, and momentums $\beta_1 = 0.5$, $\beta_2 = 0.999$. The weighting parameters of $\mathcal{L}$ is chosen as $\lambda_{adv} = 1$, $\lambda_{con} = 40$ and $\lambda_{enc} = 1$, empirically shown to yield the optimal performance (See Figure 5.5). The model is initially set to be trained for 15 epochs; however, in most cases, it learns sufficient information within fewer training cycles. Therefore, we save the parameters of the network when the performance of the model starts to decrease since this reduction is a strong indication of over-fitting. The model is implemented using PyTorch [172] (v1.2.0, Python 3.7.4, CUDA 10.1 and CUDNN 7.6). Experiments are performed using an NVIDIA Titan X GPU.

5.3.3 Evaluation

Similar to the previous work [13,14,49], the performance of the model is evaluated by the area under the curve (AUC) of the receiver operating characteristics (ROC) [175], a function plotted by the true positive rates (TPR) and false positive rates (FPR) with varying threshold values (as per prior work in the field [13, 14, 49]).

5.4 Results

Before presenting results for the full pipeline, it is essential to show how hyper-parameters affect the overall performance. Figure 5.4 demonstrates the impact of the dimension of the latent space. The x-axis shows the CIFAR-10 classes when chosen abnormal vs normal (e.g. *bird* vs rest) with various dimensionality of z and the y-axis depicts the corresponding AUC performance. We see that in eight out of ten cases, $nz = 100$ yields the highest AUC. We ,therefore, set $nz = 100$ for the rest of the experimentation.

Similar to Figure 5.4, we observe the performance change by tuning the parameters of the overall loss function shown in Equation 5.4. Figure 5.5 illustrates for the abnormal case of *car* from CIFAR-10 that weighting parameters significantly influences the overall performance. Observing the similar performance outcome for

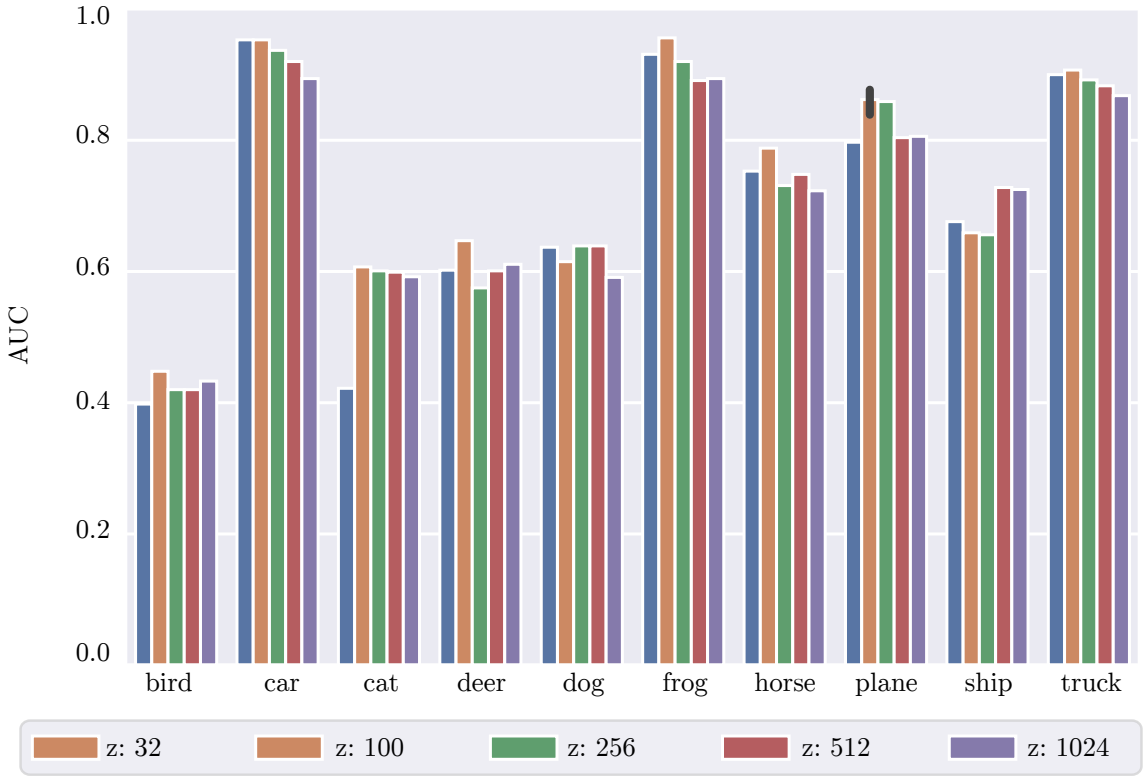


Figure 5.4: Hyper-parameter tuning for the model. The model achieves the most optimum performance when $nz = 100$.

the rest of the classes and datasets, we choose the following weighting parameters for Equation 5.4: $\lambda_{adv} = 40$, $\lambda_{rec} = 1$ and $\lambda_{enc} = 1$. Again, the rest of the results presented in this section are based on these parameters.

For the CIFAR-10 dataset, Table 5.1 and Figure 5.6 demonstrate that with the exception of abnormal classes *bird* and *dog*, the proposed model yields superior results to the prior work.

	CIFAR-10									
Model	bird	car	cat	deer	dog	frog	horse	plane	ship	truck
AnoGAN [13]	0.411	0.492	0.399	0.335	0.393	0.321	0.399	0.516	0.567	0.511
EGBAD [14]	0.383	0.514	0.448	0.374	0.481	0.353	0.526	0.577	0.413	0.555
GANomaly [49]	**0.510**	0.631	0.587	0.593	**0.628**	0.683	0.605	0.633	0.616	0.617
Proposed	0.448	**0.953**	**0.607**	**0.602**	0.615	**0.931**	**0.788**	**0.797**	**0.659**	**0.907**

Table 5.1: AUC results for CIFAR-10 dataset.

Table 5.2 presents the experimental results for UBA and FFOB datasets. It is apparent from this table that the proposed method significantly outperforms the prior work in each anomaly cases of the datasets. Of significance, the best AUC of

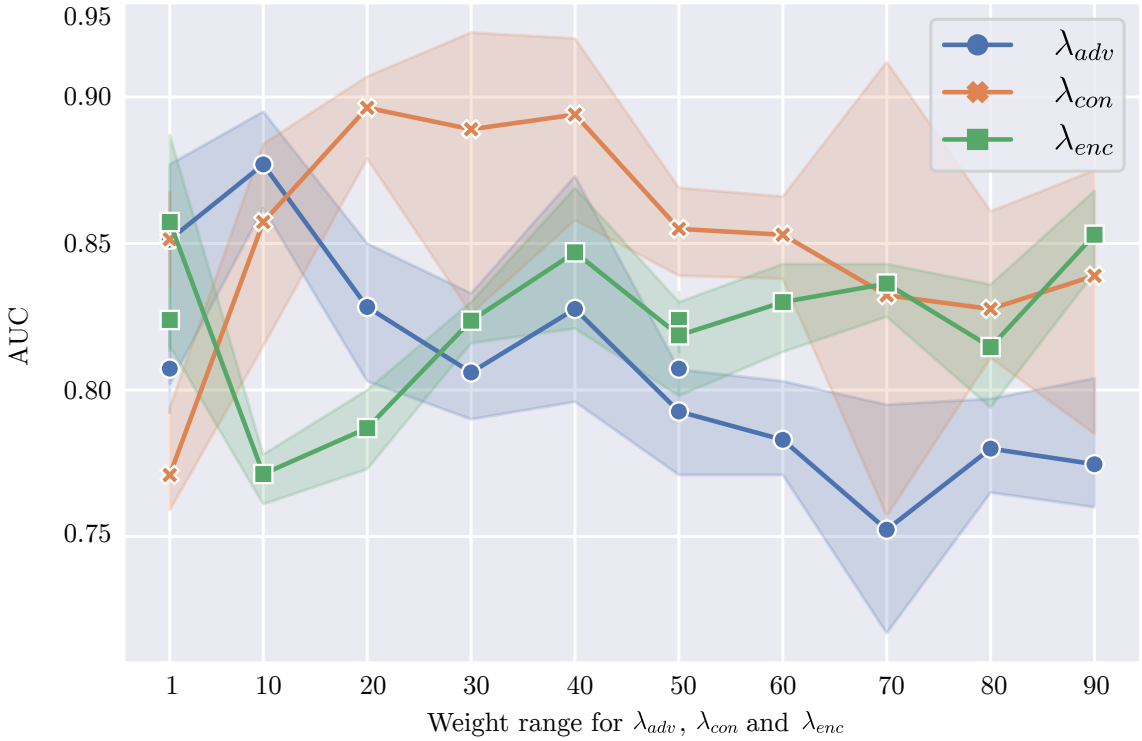


Figure 5.5: Hyper-parameter tuning for the model. The model achieves the most optimum performance when $\lambda_{adv} = 1$, $\lambda_{con=40} = 1$ and $\lambda_{con} = 1$.

the prior work is 0.599 for the most challenging abnormality case – *knife*, while the method proposed here achieves AUC of 0.904.

	UBA				FFOB
Method	gun	gun-parts	knife	overall	full-weapon
AnoGAN [13]	0.598	0.511	0.599	0.569	0.703
EGBAD [14]	0.614	0.591	0.587	0.597	0.712
GANomaly [49]	0.747	0.662	0.520	0.643	0.882
Proposed	**0.972**	**0.945**	**0.904**	**0.940**	**0.903**

Table 5.2: AUC results for UBA and FFOB datasets.

Figures 5.9 and **??** depicts exemplary test images for the datasets used in the experimentation. A significant result emerging from the examples presented within the Figures is that the proposed model is capable of generating both normal and abnormal reconstructed outputs at test time, meaning that it captures the distribution of both domains. This is probably due to the use of skip connections enabling reconstruction even for the abnormal test samples.

The qualitative results of Figures 5.9, **??**, supported by the quantitative results

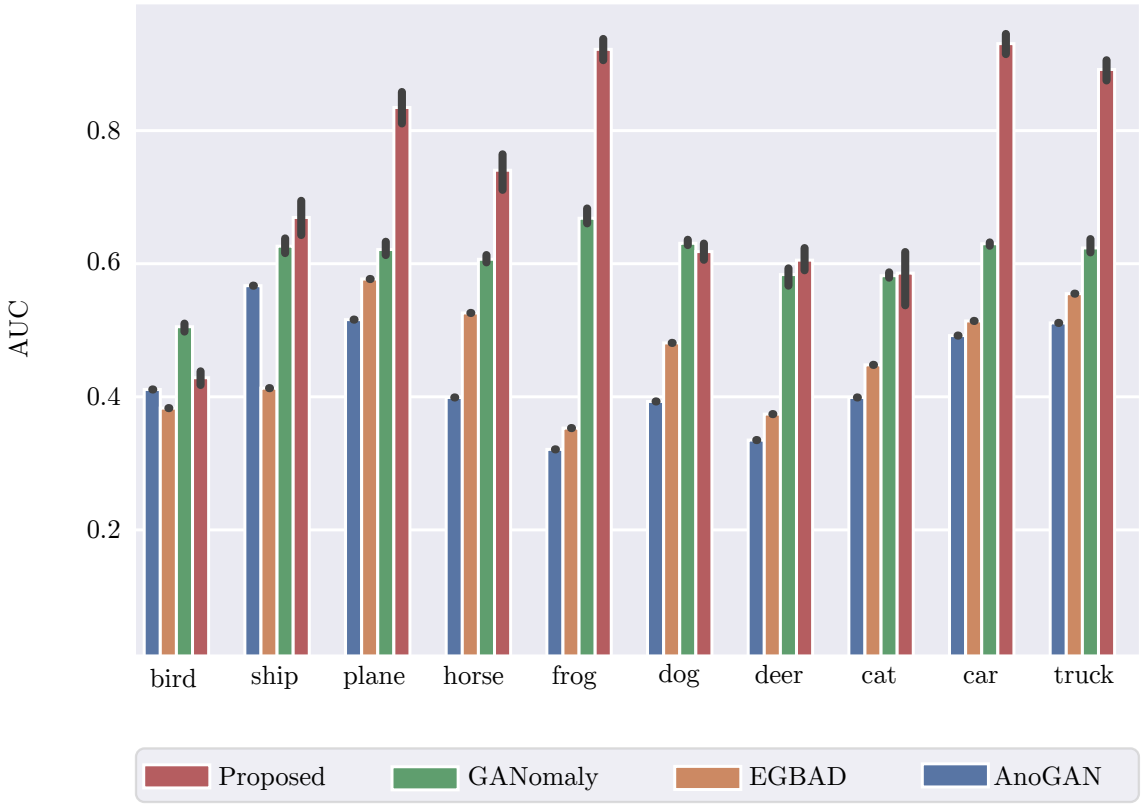


Figure 5.6: AUC results for CIFAR-10 dataset. Shaded areas in the plot represent variations due to the use of 3 random seeds.

of Table 5.2, reveal that abnormality detection is successfully made in latent object space of the model that emerges from our adversarial training over the proposed skip-connected architecture.

Figures 5.7 and 5.8 show the histogram plot (a) of the normal and abnormal scores for the test data, and the t-SNE plot (b) of the normal and abnormal features extracted from the last convolutional layer ($f(.)$) of the discriminator (see Figure 5.3). Closer inspection of the figures reveals that the model yields promising separation within both the output anomaly (reconstruction) score and the preceding convolutional feature spaces.

Figure 5.9 and **??** show that the proposed model successfully classifies the images as abnormal. It is important to note that the generator network is capable of producing abnormal examples. Apart from particular finer details and certain checkerboard artefacts on some images, the generated samples look almost the same as the real ones. Despite this low-reconstruction error on abnormal samples, the discriminator network can cope with classifying the abnormal examples within its

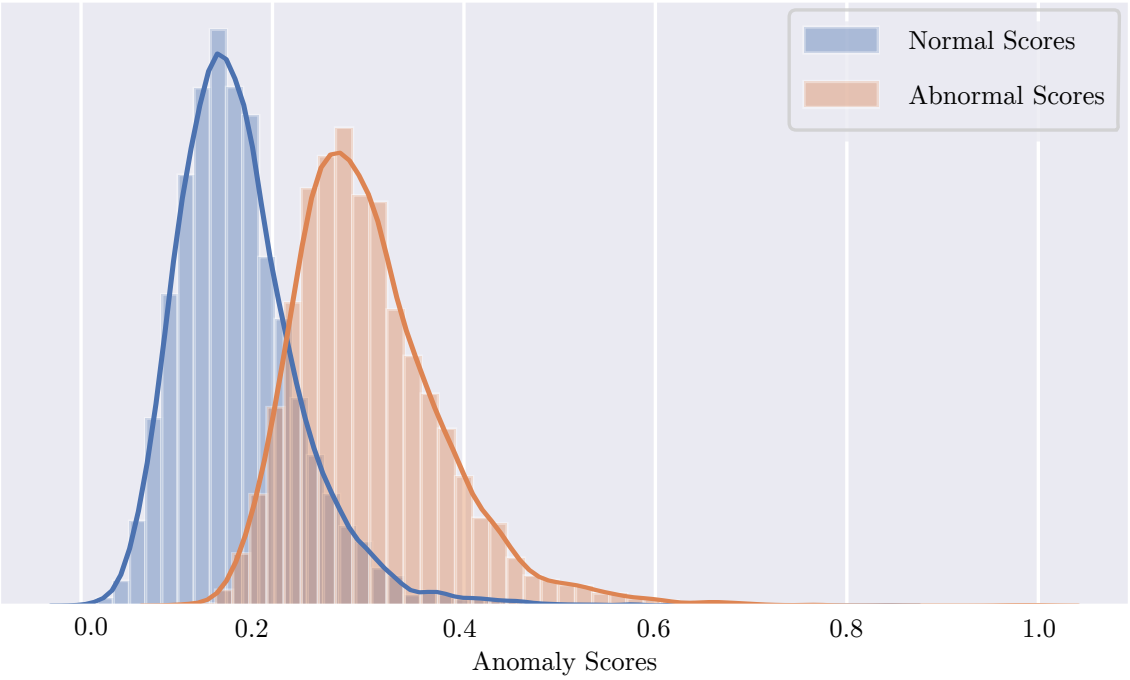


Figure 5.7: (a) Histogram of the normal and abnormal scores for the test data.

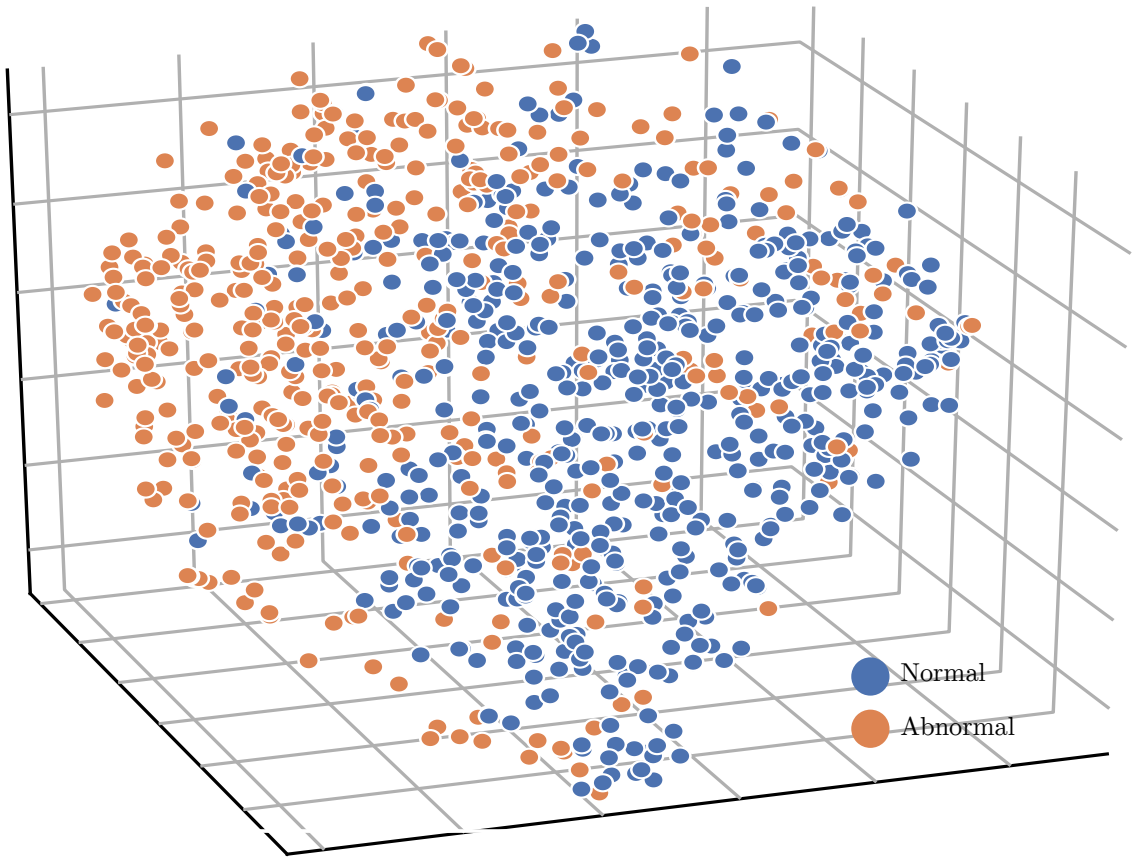


Figure 5.8: (b) t-SNE plot of the 1000 subsampled normal and abnormal features extracted from the last convolutional layer ($f(.)$) of the discriminator (Figure 5.3).

latent space.

Figure 5.10, on the other hand, illustrates exemplary images, where the model misclassifies normal and abnormal samples. Some of these misclassifications stem from mislabeled examples (left and right images of the top two rows and the third row). Misclassified examples on the fourth row are because the model labels the metallic objects as abnormal. Examples on the fourth row, finally, are incorrectly classified as benign since the model misses tiny firearm-parts.

Overall, these results indicate that the proposed approach yields superior anomaly detection performance to the model presented in Chapter 4 and to the previous state-of-the-art approaches [13, 14].

5.5 Conclusion

This chapter introduces an anomaly detection method designed to address the limitations of GANomaly [49], presented in Chapter 4. Despite the superior results, the high-reconstruction error of normal/abnormal samples and redundant parametrisation limits GANomaly for deployment.

The model presented in this chapter, on the other hand, tackles with these issues by (i) utilising skip-connected networks [174] that reconstruct high-quality image outputs, (ii) removing the second encoder network E from GANomaly (See Figure 4.2 upper right) and learning the latent space representation within the discriminator network (Figure 5.3).

Evaluating the model on various datasets such as CIFAR-10 [171], UBA [49] and FFOB [15] show that the proposed model significantly outperforms Schlegl *et al.* [13], Zenati *et al.* [14] and GANomaly [49] (Chapter 4). The empirical findings in this study provide an insight into the generalization capability of the proposed method to any anomaly detection task.

Despite these promising results, certain issues need further research. As discussed in Section 5.4, and depicted in Figures 5.9, **??** and 5.10, the generator network is capable of producing realistic samples even for abnormal images, an indication that the distribution of the abnormal samples is a subset of the normal distribution

learned during the training stage. Although the discriminator network can classify the abnormal images within the latent space, further research could concentrate on a training regime such that the generator network learns the normal distribution from a small subset of normal data in order not to capture the distribution of the abnormality.

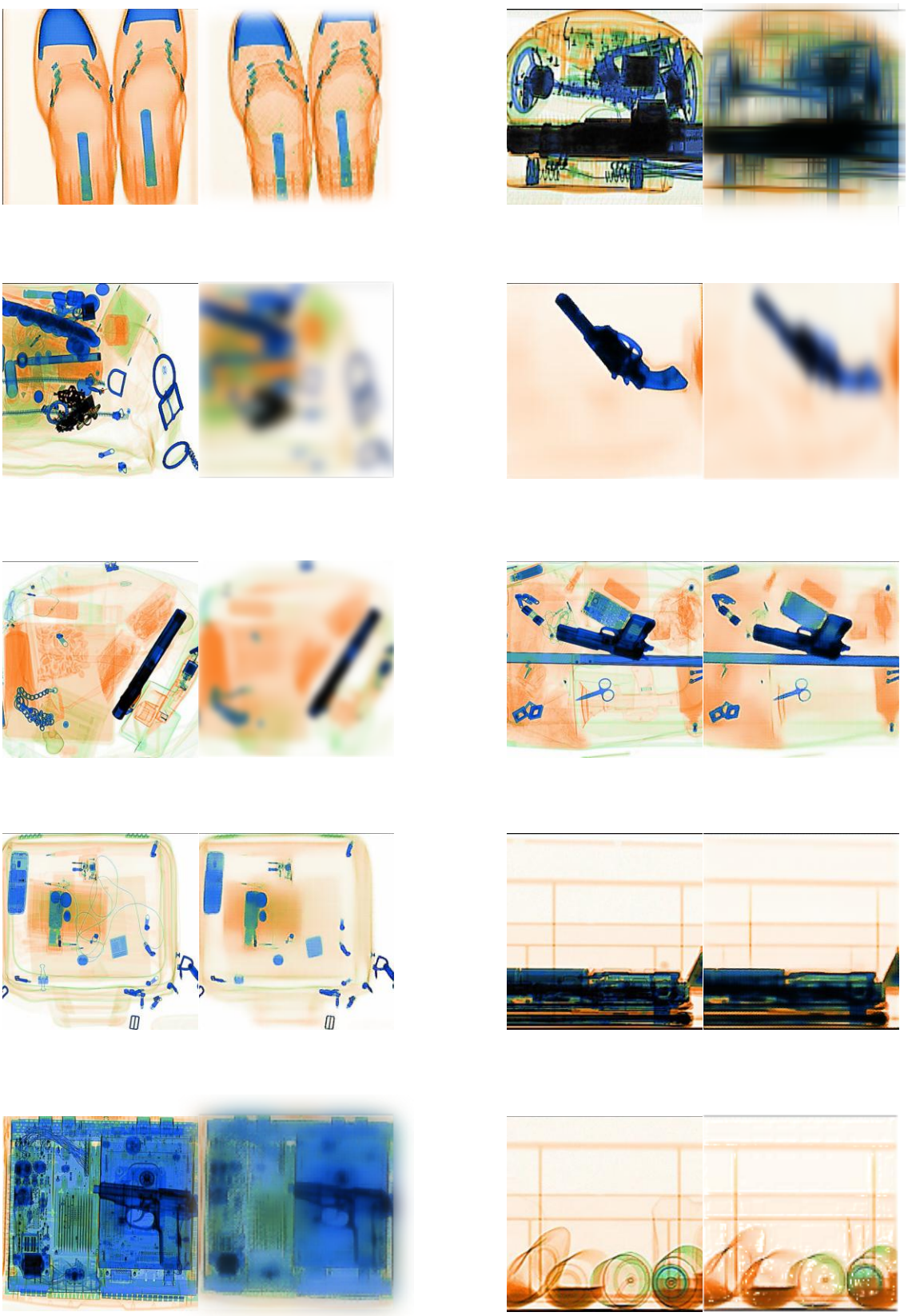


Figure 5.9: Randomly selected normal and abnormal test images. The generator has a tendency to blur out the images not seen during training.

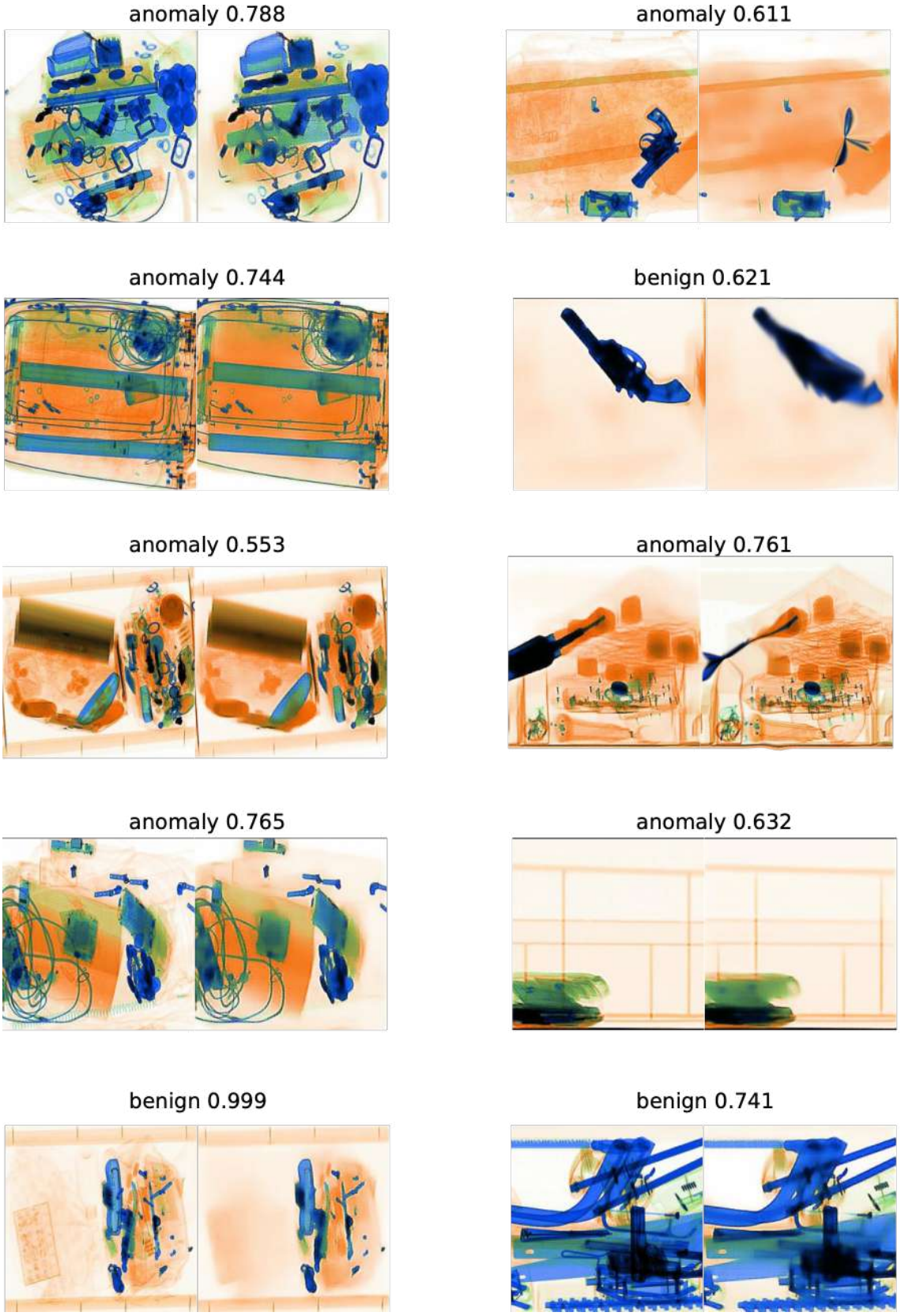


Figure 5.10: Randomly selected normal and abnormal test images. In most cases, the model predicts the metallic objects as threats and classifies them as an anomaly.

CHAPTER 6

Conclusion

Recent developments in the field of machine and deep learning have led to an increased interest in automated X-ray security screening systems. Despite the considerable literature grown up, the primary scope of the prior work is mainly limited to classical machine learning or supervised deep learning methods. This thesis initially gives an overview of such methods and provides a thorough evaluation of the use of state-of-the-art deep learning algorithms within classification and detection tasks in X-ray security imaging. By employing the transfer learning paradigm, the thesis shows that the use of supervised deep learning algorithms significantly outperforms conventional learning techniques. Despite the performance gain, limitations of the supervised deep learning approaches, stemming from imbalanced X-ray datasets (benign $\gg$ threat) are also pointed out, which ultimately transitions the scope of the thesis towards deep unsupervised methods.

To address the severe class imbalance issue, the thesis introduces two novel unsupervised anomaly detection algorithm whereby the models are trained on benign samples to learn the distribution of the non-threat material and are tested on both benign and threat images to detect illicit materials as outliers. The following section outlines the main contributions of the thesis.

6.1 Contributions

Chapter 3 exhaustively evaluates the use of state-of-the-art classification architectures against the prior work in the field. The work presented here compares end-to-end transfer-learned CNN classification and the final stage SVM classification on the extracted CNN features as well as SVM classification on conventional hand-crafted features. Empirical findings indicate that fine-tuned CNN features yield superior performance to conventional hand-crafted features on object classification tasks within this context. Overall, the highest accuracy is achieved by the AlexNet features trained with Support Vector Machine (SVM) classifier (0.994) on the firearm classification problem.

In addition to the classification task, an additional set of experiments compares the region based object detection/localization strategies of [10,142] against the prior strategies proposed in [99,150]. Contrasting performance results are obtained against the prior published studies of [8,9] over a comprehensive dataset of $11,627$ examples making this one of the largest combined X-ray object detection and classification study in the literature to date. With the use of YOLOv2 [12], using input images of size 544×544, we achieve 0.885 mean average precision (mAP) for a six-class object detection problem. The same approach with an input of size 416×416 yields 0.974 mAP for the two-class firearm detection problem and requires approximately 100ms per image.

Moreover, the evaluation is strengthened further by using UK government evaluation dataset (CAST) [15]. VGG16 [5] network yields 0.999 accuracy on *Full Firearm vs Operational Benign* dataset extracted from the CAST dataset. Overall, the Chapter identifies the classification approaches and the detection strategies that outperform the prior work of [9, 84, 99] in a supervised fashion.

Addressing the difficulties of supervised learning-based methods, and imbalanced nature of X-ray security imaging datasets, Chapter 4 presents a generic unsupervised anomaly detection architecture comprising an adversarial training framework. The proposed approach uses single colour images as the input drawn only from *normal* (non-anomalous) training examples. Unlike the previous algorithms requiring two-stage training, the proposed approach has single-stage training and is both efficient

for model training and later inference (run-time testing). The novelty of the proposed algorithm comes from the adversarial autoencoder scheme within an encoder-decoder-encoder pipeline, capturing the training data distribution within both image and latent vector space. An adversarial training architecture such as this, practically based on only *normal* training data examples, produces superior performance to the prior work [13, 14] over challenging benchmark problems [15, 167, 171].

Chapter 5 further extends the anomaly detection algorithm presented in Chapter 4 via adversarial training over a skip-connected encoder-decoder (convolutional neural) network architecture. Whilst adversarial training has shown the promise of GAN in this domain, as demonstrated in Chapter 4, skip-connections within such UNet-style (encoder-decoder) [174] generator networks are known to enable the multi-scale capture of image space detail with sufficient capacity to generate high-quality normal images drawn from the distribution the model has learned. The proposed approach also seeks to learn the normal distribution in both the image and latent spaces via a GAN generator-discriminator paradigm. The discriminator network not only forces the generator to learn an improved model of the distribution but also works as a feature extractor such that it learns the reconstruction of the normal distribution within a lower-dimensional latent space This proposed pipeline outperforms the previous work [13, 14, 49] on challenging anomaly-detection problems [15, 167, 171].

Overall, this thesis aims to initially provide an overview to the supervised deep learning methods, and advance the literature by proposing two novel unsupervised anomaly detection algorithms for the classification of the threat items within X-ray security imaging. The evaluation of deep supervised approaches demonstrates promising detection performance in case of having a well-balanced and annotated datasets. The second significant finding of the thesis is that unsupervised deep anomaly detection algorithms could yield encouraging performance, where the datasets are highly biased towards certain classes and lack annotations. Taken together, these results suggest that the use of the proposed algorithms could help human operators to detect threat items, strengthening security screening. The generalisability of these results is subject to certain limitations, as discussed in Section

6.2. Further studies need to be carried out in order to further investigate the limitations.

6.2 Limitations and Future Work

Despite the promising performance of the proposed approaches, there are still some identifiable limitations. This section discusses the challenges and future directions based on the weaknesses and strengths of the current approaches presented in this thesis and the broader literature, including concurrent work to that presented here.

6.2.1 Data

Although the use of transfer learning improves the performance of small X-ray datasets, the lack of large datasets limits contemporary deep model training. Relatively large datasets in the field such as SIXray, FFOB are highly biased towards certain classes, limiting to train reliable supervised methods. Hence, it is essential to build large, homogeneous, realistic and publicly available datasets, collected either by (i) manually scanning numerous bags with different objects and orientations in a lab environment or (ii) generating synthetic datasets via contemporary algorithms.

There are advantages and disadvantages to both methods. Although manual data collection enables to gather realistic samples with the flexibility to produce any combination, it is rather expensive, requiring tremendous human effort and time.

Synthetic dataset generation, on that hand, is another method, currently achieved by TIP [68,69] or GAN [44,106]. A recent study empirically demonstrates that using a TIP dataset for a detection task adversely impacts the detection performance [176]. In future work, therefore, more advanced algorithms such as image translation or domain adaptation [138, 177] could be considered such that the model would learn to translate between benign and threat domains, which overall would yield superior projection/translation to TIP.

The literature has also seen another type of synthetic datasets generated by GAN algorithms. The limitation of current GAN datasets [44, 106], however, is that the

models are currently capable of producing only objects but full X-ray images. Moreover, the quality of the generated images is far from being realistic. Further studies, taking these issues into account, will need to be undertaken. It might be feasible to create more realistic X-ray images by using contemporary GAN algorithms [178].

6.2.2 Exploiting Multiple-View Information

Existing research recognises the critical role played by multiple-view imagery, especially when the detection of an object from a particular viewpoint is challenging [45, 46, 125]. Two key studies [45] and [46] investigate utilising multiple-view integration inside/outside a CNN. Despite the incremental performance improvement reported, further work is required to investigate other possible ways to utilise multiple-view imagery better.

6.2.3 Generalization Ability – *Transferring Between Domains*

As pointed out in [41] and [144], transferring models between different scanners could be challenging due to the unknown intrinsics of the scanners. Future work would utilize domain adaptation [177], where the source domain contains images from one scanner, and the target domain would be of another X-ray scanner. Training with even unbalanced datasets would learn the intrinsic, and map from one to the other.

6.2.4 Improving Unsupervised Anomaly Detection Approaches

As explained in Section 2.4.1, the current datasets available within the literature are highly imbalanced such that a number of benign samples are significantly larger than threat images [49–51].

To address this issue, Chapters 4 and 5 employ unsupervised algorithms trained on only benign samples, and tested on both benign and threat examples. The primary idea here is to learn to reconstruct only normal samples within the image and latent spaces such that the model would fail to reconstruct abnormal images. However, more research on this topic needs to be undertaken to devise better reconstruction techniques that thoroughly learn the characteristics of the normality

from which the abnormality would be detected. In addition, current techniques proposed in these chapters scores the anomalies based on a threshold that is manually chosen and is data specific. For a better anomaly detection without needing for manual thresholding, therefore, more sophisticated classification approaches could be proposed.

6.2.5 Use of the Material Information

In dual-energy X-ray systems attenuation between high and low energies yields a unique value for different materials, which could be utilised further for more accurate object classification/detection [179,180]. Even though recent research [104,134] have examined the use of material information, the research outcome present inconsistent results. Morris *et al.* [134], for instance, show that Z-effective, when trained itself, achieves the highest detection performance. Rogers *et al.* [104], on the other hand, demonstrate that networks fed with 4-channel inputs ($\{-\log L, L, H, -\log H\}$) yield the highest classification accuracy. Hence, a further study thoroughly investigating the material information is suggested.

Bibliography

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 7, pp. 770–778, IEEE, 2016. ix, x, 22, 27, 32, 33, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 51, 127, 128

[2] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," tech. rep., 2018. ix, 22, 25

[3] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," in *International Conference on Computer Vision (ICCV)*, pp. 2961–2969, IEEE, 2017. ix, 22, 27, 132

[4] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization," in *International Conference on Computer Vision (ICCV)*, pp. 618–626, IEEE, 2017. ix, 33, 34, 38

[5] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in *International Conference on Learning Representations (ICLR)*, 2015. ix, x, 22, 23, 24, 25, 26, 32, 33, 34, 37, 38, 39, 40, 41, 42, 51, 102, 127

[6] L. van der Maaten and G. Hinton, "Visualizing Data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008. x, 37, 41

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Conference on Neural Information Processing Systems (NeurIPS)*, pp. 1097–1105, Curran Associates, Inc., 2012. x, 4, 10, 22, 31, 32, 33, 37, 38, 39, 40, 41, 42, 51, 125

[8] M. Kundegorski, S. Akçay, M. Devereux, A. Mouton, and T. Breckon, "On using Feature Descriptors as Visual Words for Object Detection within X-ray Baggage Security Screening," in *International Conference on Imaging for*

Crime Detection and Prevention (ICDP), pp. 12 (6 .)–12 (6 .), IET, 2016. x, 10, 15, 18, 31, 32, 34, 37, 39, 40, 50, 63, 102

[9] S. Akçay, M. E. Kundegorski, M. Devereux, and T. P. Breckon, "Transfer Learning Using Convolutional Neural Networks for Object Classification within X-ray Baggage Security Imagery," in *International Conference on Image Processing (ICIP)*, pp. 1057–1061, IEEE, 2016. x, 4, 10, 18, 21, 22, 28, 31, 32, 33, 39, 50, 102

[10] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 39, no. 6, pp. 1137–1149, 2017. x, xiii, xiv, xv, 25, 32, 49, 50, 51, 53, 54, 55, 63, 102, 128, 129

[11] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object Detection via Region-based Fully Convolutional Networks," in *International Conference on Neural Information Processing Systems (NeurIPS)*, pp. 379–387, 2016. x, xiv, xv, 32, 49, 50, 53, 55, 63, 130

[12] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," in *Conference on Computer Vision and Pattern Recognition*, pp. 7263–7271, IEEE, 2017. x, xiv, xv, 32, 49, 50, 53, 55, 63, 102

[13] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10265 LNCS, pp. 146–147, 2017. xi, 5, 66, 67, 70, 72, 74, 75, 76, 77, 82, 83, 84, 85, 86, 88, 90, 92, 93, 94, 97, 103

[14] H. Zenati, C. S. Foo, B. Lecouat, G. Manek, and V. R. Chandrasekhar, "Efficient GAN-Based Anomaly Detection," *CoRR*, 2018. xi, 5, 67, 70, 72, 73, 74, 75, 76, 77, 82, 83, 84, 85, 86, 88, 90, 92, 93, 94, 97, 103

[15] Centre for Applied Scienceand Technology (CAST), "OSCT Borders X-ray Image Library," tech. rep., UK Home Office, 2016. xiv, 20, 32, 35, 41, 42, 73, 83, 85, 91, 97, 102, 103

[16] R. Girshick, "Fast R-CNN," in *International Conference on Computer Vision (ICCV)*, pp. 1440–1448, IEEE, 2015. xiv, xv, 53, 54, 55, 63, 128

[17] A. Mouton and T. P. Breckon, "A Review of Automated Image Understanding within 3D Baggage Computed Tomography Security Screening," *Journal of X-Ray Science and Technology*, vol. 23, no. 5, pp. 531–555, 2015. 1, 3, 9, 35

[18] T. W. Rogers, N. Jaccard, E. J. Morton, and L. D. Griffin, "Automated X-ray Image Analysis for Cargo Security: Critical Review and Future Promise," *Journal of X-Ray Science and Technology*, vol. 25, no. 1, pp. 33–56, 2017. 1, 3, 9, 14, 73

[19] N. C. Murray and K. Riordan, "Evaluation of Automatic Explosive Detection Systems," in *International Carnahan Conference on Security Technology*, pp. 175–179, IEEE, 1995. 1, 3, 8

[20] G. Zentai, "X-ray Imaging for Homeland Security," in *International Workshop on Imaging Systems and Techniques*, pp. 1–6, IEEE, 2008. 1, 3, 8

[21] K. Wells and D. Bradley, "A Review of X-ray Explosives Detection Techniques for Checked Baggage," *Applied Radiation and Isotopes*, vol. 70, pp. 1729–1746, aug 2012. 1, 3, 8

[22] J. S. Caygill, F. Davis, and S. P. J. Higson, "Current Trends in Explosive Detection Techniques," *Talanta*, vol. 88, pp. 14–29, 2012. 1, 3, 8

[23] S. Singh and M. Singh, "Explosives Detection Systems (EDS) for Aviation Security," *Signal Processing*, vol. 83, no. 1, pp. 31–55, 2003. 1, 3, 8

[24] L. Swann, V. Popovic, A. L. Blackler, and B. J. Kraal, "Airport Security Screeners Expertise and Implications For Interface Design," in *Design Research Society Conference 2014*, 2014. 2

[25] A. Chavaillaz, A. Schwaninger, S. Michel, and J. Sauer, "Expertise, Automation and Trust in X-Ray Screening of Cabin Baggage," *Frontiers in Psychology*, vol. 10, 2019. 2

[26] S. Michel, S. M. Koller, J. C. de Ruiter, R. Moerland, M. Hogervorst, and A. Schwaninger, "Computer-Based Training Increases Efficiency in X-Ray Image Interpretation by Aviation Security Screeners," in *International Carnahan Conference on Security Technology*, pp. 201–206, IEEE, 2007. 2, 13

[27] T. Halbherr, A. Schwaninger, G. R. Budgell, and A. Wales, "Airport Security Screener Competency: A Cross-Sectional and Longitudinal Analysis," *The International Journal of Aviation Psychology*, vol. 23, no. 2, pp. 113–129, 2013. 2

[28] S. Baeriswyl, A. Krause, and A. Schwaninger, "Emotional Exhaustion and Job Satisfaction in Airport Security Officers – Work–Family Conflict as Mediator in the Job Demands–Resources Model," *Frontiers in Psychology*, vol. 7, p. 663, 2016. 2

[29] Y. Sterchi, N. Hättenschwiler, and A. Schwaninger, "Detection Measures For Visual Inspection of X-ray Images of Passenger Baggage," *Attention, Perception, & Psychophysics*, 2019. 2

[30] A. Chavaillaz, A. Schwaninger, S. Michel, and J. Sauer, "Automation in Visual Inspection Tasks: X-ray Luggage Screening Supported by A System Of Direct, Indirect or Adaptable Cueing with Low and High System Reliability," *Ergonomics*, vol. 61, no. 10, pp. 1395–1408, 2018. 2

[31] 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 *International Conference on Research in Air Transportation*, p. 8, 2008. 3

[32] 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 *Symposium of the Austrian HCI and Usability Engineering Group* (A. Holzinger, ed.), Lecture Notes in Computer Science, pp. 419–438, Springer Berlin Heidelberg, 2008. 3

[33] A. Wales, T. Halbherr, and A. Schwaninger, "Using speed measures to predict performance in x-ray luggage screening tasks," in *International Carnahan Conference on Security Technology*, pp. 212–215, IEEE, 2009. 3

[34] M. Mendes, A. Schwaninger, N. Strebel, and S. Michel, "Why Laptops Should Be Screened Separately When Conventional X-ray Screening Is Used," in *International Carnahan Conference on Security Technology (ICCST)*, pp. 267–273, IEEE, 2012. 3

[35] M. Mendes, A. Schwaninger, and S. Michel, "Can Laptops Be Left Inside Passenger Bags If Motion Imaging Is Used in X-ray Security Screening?," *Frontiers in Human Neuroscience*, vol. 7, 2013. 3

[36] B. R. Abidi, D. L. Page, and M. A. Abidi, "A Combinational Approach to the Fusion, De-noising and Enhancement of Dual-Energy X-Ray Luggage Images," in *Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, vol. 3, p. 2, IEEE, 2005. 3, 8, 9, 10, 12

[37] B. R. Abidi, Y. Zheng, A. V. Gribok, and M. A. Abidi, "Improving Weapon Detection In Single Energy X-ray Images Through Pseudocoloring," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 36, no. 6, pp. 784–796, 2006. 3, 8

[38] Q. Lu and R. Conners, "Using Image Processing Methods to Improve the Explosive Detection Accuracy," *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, vol. 36, no. 6, pp. 750–760, 2006. 3, 8, 9, 10, 17

[39] M. Singh and S. Singh, "Image Segmentation Optimisation For X-ray Images Of Airline Luggage," in *International Conference on Computational Intelligence for Homeland Security and Personal Safety (CIHSPS)*, pp. 10–17, IEEE, 2004. 3, 8, 10, 17

[40] G. Heitz and G. Chechik, "Object Separation in X-ray Image Sets," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2093–2100, IEEE, 2010. 3, 8, 9, 10, 17

[41] M. Caldwell, M. Ransley, T. W. Rogers, and L. D. Griffin, "Transferring X-ray Based Automated Threat Detection Between Scanners With Different Energies And Resolution," in *Counterterrorism, Crime Fighting, Forensics, and Surveillance Technologies*, p. 15, SPIE, 2017. 3, 8, 10, 21, 24, 25, 28, 105

[42] S. Akçay, M. E. Kundegorski, C. G. Willcocks, and T. P. Breckon, "Using Deep Convolutional Neural Network Architectures for Object Classification and Detection within X-ray Baggage Security Imagery," *IEEE Transactions on Information Forensics and Security*, 2018. 3, 8, 10, 18, 20, 21, 73

[43] C. Miao, L. Xie, F. Wan, C. Su, H. Liu, J. Jiao, and Q. Ye, "SIXray : A Large-scale Security Inspection X-ray Benchmark for Prohibited Item Discovery in Overlapping Images," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2019. 3, 8, 9, 10, 19, 21, 23, 28

[44] J. Yang, Z. Zhao, H. Zhang, and Y. Shi, "Data Augmentation for X-Ray Prohibited Item Images Using Generative Adversarial Networks," *IEEE Access*, pp. 1–1, 2019. 3, 8, 21, 23, 28, 104

[45] K. Liang, C. Gregory, S. O. Diallo, K. Roe, G. Heilmann, L. Carin, D. Carlson, G. Spell, and J. Sigman, "Automatic Threat Recognition Of Prohibited Items At Aviation Checkpoint With X-ray Imaging: A Deep Learning Approach," in *Anomaly Detection and Imaging with X-Rays (ADIX) III* (A. Ashok, M. A. Neifeld, M. E. Gehm, and J. A. Greenberg, eds.), p. 2, SPIE, 2018. 3, 8, 21, 25, 28, 105

[46] J.-M. O. Steitz, F. Saeedan, and S. Roth, "Multi-view X-Ray R-CNN," in *German Conference on Pattern Recognition (GCPR)*, pp. 153–168, 2019. 3, 8, 9, 10, 21, 25, 28, 105

[47] Z. Liu, J. Li, Y. Shu, and D. Zhang, "Detection and Recognition of Security Detection Object Based on Yolo9000," in *International Conference on Systems and Informatics (ICSAI)*, pp. 278–282, IEEE, 2018. 3, 8, 10, 21, 25, 28

[48] J. T. A. Andrews, E. J. Morton, and L. D. Griffin, "Detecting Anomalous Data Using Auto-encoders," *International Journal of Machine Learning and Computing*, vol. 6, no. 1, p. 21, 2016. 3, 8, 10, 26, 28

[49] S. Akcay, A. Atapour-Abarghouei, and T. P. Breckon, "GANomaly: Semi-supervised Anomaly Detection via Adversarial Training," in *Asian Conference on Computer Vision - ACCV*, pp. 622–637, Springer, 2019. 3, 8, 10, 20, 21, 26, 28, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 97, 103, 105

[50] S. Akcay, A. Atapour-Abarghouei, and T. P. Breckon, "Skip-GANomaly: Skip Connected and Adversarially Trained Encoder-Decoder Anomaly Detection," in *International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, IEEE, 2019. 3, 8, 9, 10, 20, 21, 26, 28, 105

[51] L. D. Griffin, M. Caldwell, J. T. A. Andrews, and H. Bohler, ""Unexpected Item in the Bagging Area": Anomaly Detection in X-Ray Security Images,"

IEEE Transactions on Information Forensics and Security, vol. 14, no. 6, pp. 1539–1553, 2019. 3, 8, 9, 10, 26, 28, 105

[52] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015. 4

[53] M. D. Zeiler and R. Fergus, "Visualizing and Understanding Convolutional Networks," pp. 818–833, Springer, Cham, 2014. 4

[54] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based Learning Applied To Document Recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998. 4, 9, 26, 127

[55] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision*, vol. 115, pp. 211–252, dec 2015. 4, 10, 33, 34, 37, 38, 51

[56] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How Transferable Are Features In Deep Neural Networks?," in *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 3320–3328, 2014. 4, 10, 33

[57] D. Mery, E. Svec, M. Arias, V. Riffo, J. M. Saavedra, and S. Banerjee, "Modern Computer Vision Techniques for X-Ray Testing in Baggage Inspection," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 4, pp. 682–692, 2017. 4, 9, 10, 16, 21, 31, 32, 33

[58] N. Jaccard, T. W. Rogers, E. J. Morton, and L. D. Griffin, "Tackling The X-ray Cargo Inspection Challenge Using Machine Learning," in *Anomaly Detection and Imaging with X-Rays* (A. Ashok, M. A. Neifeld, and M. E. Gehm, eds.), vol. 9847, p. 98470N, SPIE, 2016. 4, 10, 19, 22, 28

[59] C. Badue, R. Guidolini, R. V. Carneiro, P. Azevedo, V. B. Cardoso, A. Forechi, L. Jesus, R. Berriel, T. Paixão, F. Mutz, L. Veronese, T. Oliveira-Santos, and A. F. De Souza, "Self-Driving Cars: A Survey," *CoRR*, vol. October, 2019. 4

[60] J. An and S. Cho, "SNU Data Mining Center 2015-2 Special Lecture on IE Variational Autoencoder based Anomaly Detection using Reconstruction Probability," tech. rep., 2015. 5, 83

[61] D. Mery, E. Svec, and M. Arias, "Object Recognition in Baggage Inspection Using Adaptive Sparse Representations of X-ray Images," in *Pacific-Rim Symposium on Image and Video Technology (PSIVT)*, pp. 709–720, Springer, Cham, 2016. 9, 10, 15, 32

[62] M. Mitckes, "Threat Image Projection – An Overview," tech. rep., 2003. 9, 10, 13

[63] B. R. Abidi, J. Liang, M. Mitckes, and M. A. Abidi, "Improving The Detection Of Low-density Weapons In X-ray Luggage Scans Using Image Enhancement And Novel Scene-decluttering Techniques," *Journal of Electronic Imaging*, vol. 13, no. 3, pp. 523–539, 2004. 10, 12

[64] M. Singh and S. Singh, "Optimizing Image Enhancement For Screening Luggage At Airports," in *International Conference on Computational Intelligence for Homeland Security and Personal Safety (CIHSPS)*, pp. 131–136, IEEE, 2005. 10, 12

[65] B. Abidi, Y. Zheng, A. Gribok, and M. Abidi, "Screener Evaluation of Pseudo-Colored Single Energy X-ray Luggage Images," in *Conference on Computer Vision and Pattern Recognition (CVPR) - Workshops*, vol. 3, pp. 35–35, IEEE, 2005. 10, 12

[66] T. W. Rogers, J. Ollier, E. J. Morton, and L. D. Griffin, "Reduction Of Wobble Artefacts In Images From Mobile Transmission X-ray Vehicle Scanners," in *International Conference on Imaging Systems and Techniques (IST)*, pp. 356–360, IEEE, 2014. 10, 12

[67] T. W. Rogers, J. Ollier, E. J. Morton, and L. D. Griffin, "Measuring And Correcting Wobble In Large-scale Transmission Radiography," *Journal of X-ray Science and Technology*, vol. 25, pp. 57–77, 2017. 10, 12

[68] T. W. Rogers, N. Jaccard, E. D. Protonotarios, J. Ollier, E. J. Morton, and L. D. Griffin, "Threat Image Projection (TIP) into X-ray Images of Cargo Containers for Training Humans and Machines," in *International Carnahan Conference on Security Technology (ICCST)*, pp. 1–7, IEEE, 2016. 10, 13, 19, 104

[69] D. Mery and A. K. Katsaggelos, "A Logarithmic X-Ray Imaging Model for Baggage Inspection: Simulation and Object Detection," in *Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 251–259, IEEE, 2017. 10, 13, 104

[70] R. Paranjape, M. Sluser, and E. Runtz, "Segmentation of handguns in dual energy X-ray imagery of passenger carry-on baggage," in *Canadian Conference on Electrical and Computer Engineering*, vol. 1, pp. 377–380, IEEE, 1998. 10, 17

[71] M. Sluser and R. Paranjape, "Model-based Probabilistic Relaxation Segmentation Applied To Threat Detection In Airport X-ray Imagery," in *Canadian Conference on Electrical and Computer Engineering*, vol. 2, pp. 720–726, IEEE, 1999. 10, 17

[72] J. Ding, Y. Li, X. Xu, and L. Wang, "X-ray Image Segmentation by Attribute Relational Graph Matching," in *International Conference on Signal Processing*, IEEE, 2006. 10, 17

[73] O. Kechagias-Stamatis, N. Aouf, C. Belloni, and D. Nam, "Automatic X-ray Image Segmentation And Clustering For Threat Detection," in *Target and Background Signatures III*, p. 24, SPIE, 2017. 10, 18

[74] C. Oertel and P. Bock, "Identification of Objects-of-Interest in X-Ray Images," in *Applied Imagery and Pattern Recognition Workshop (AIPR)*, pp. 17–17, IEEE, 2006. 10

[75] R. Gesick, C. Saritac, and C.-C. Hung, "Automatic image analysis process for the detection of concealed weapons," in *Annual Workshop on Cyber Security and Information Intelligence Research Cyber Security and Information Intelligence Challenges and Strategies (CSIIRW)*, p. 1, ACM Press, 2009. 10

[76] K. Fu, C. Guest, and P. Das, "Segmentation of suspicious objects in an x-ray image using automated region filling approach," in *Signal and Data Processing of Small Targets 2009*, vol. 7445, p. 744510, International Society for Optics and Photonics, 2009. 10

[77] M. Baştan, M. R. Yousefi, and T. M. Breuel, "Visual Words on Baggage X-Ray Images," in *International Conference on Computer Analysis of Images and Patterns*, Lecture Notes in Computer Science, pp. 360–368, Springer Berlin Heidelberg, 2011. 10, 14, 31, 32, 63

[78] D. Turcsany, A. Mouton, and T. P. Breckon, "Improving feature-based object recognition for X-ray baggage security screening using primed visualwords," in *International Conference on Industrial Technology (ICIT)*, pp. 1140–1145, IEEE, 2013. 10, 14, 31, 32, 63

[79] M. Bastan, W. Byeon, and T. M. Breuel, "Object Recognition in Multi-View Dual Energy X-ray Images – Executive summary," *British Machine Vision Conference (BMVC)*, pp. 1–11, 2013. 10, 16, 31, 63

[80] Y. Zheng and A. Elmaghraby, "A Vehicle Threat Detection System Using Correlation Analysis And Synthesized X-ray Images," in *Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XVIII*, vol. 8709, p. 87090V, International Society for Optics and Photonics, 2013. 10, 15

[81] J. Zhang, L. Zhang, Z. Zhao, Y. Liu, J. Gu, Q. Li, and D. Zhang, "Joint Shape and Texture Based X-Ray Cargo Image Classification," in *Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 266–273, IEEE, 2014. 10, 15

[82] N. Jaccard, T. W. Rogers, and L. D. Griffin, "Automated Detection Of Cars In Transmission X-ray Images Of Freight Containers," in *International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 387–392, IEEE, 2014. 10, 14

[83] S. Kolkoori, N. Wrobel, A. Deresch, B. Redmer, and U. Ewert, "Dual High-energy X-ray Digital Radiography For Material Discrimination In Cargo Containers," in *European Conference on Non-Destructive Testing (ECNDT)*, pp. 6–10, 2014. 10

[84] M. Bastan, "Multi-view Object Detection In Dual-energy X-ray Images," *Machine Vision and Applications*, vol. 26, no. 7-8, pp. 1045–1060, 2015. 10, 16, 31, 32, 51, 63, 102

[85] E. Morton, T. Rogers, L. Griffin, and N. Jaccard, "Detection Of Cargo Container Loads From X-ray Images," in *International Conference on Intelligent Signal Processing 2015 (ISP)*, pp. 6 .–6 ., IET, 2015. 10, 14

[86] N. Zhang and J. Zhu, "A Study Of X-ray Machine Image Local Semantic Features Extraction Model Based On Bag-ofwords For Airport Security," *International Journal on Smart Sensing and Intelligent Systems*, vol. 8, no. 1, pp. 45–64, 2015. 10, 15

[87] N. Zhang, "A Study On Optimization Methods Of X-ray Machine Recognition For Aviation Security System," *International Journal on Smart Sensing and Intelligent Systems*, vol. 8, no. 2, pp. 1313–1332, 2015. 10, 15

[88] O. Abusaeeda, J. Evans, D. Downes, and J. Chan, "View Synthesis Of KDEX Imagery For 3d Security X-ray Imaging," in *International Conference on Imaging for Crime Detection and Prevention (ICDP)*, pp. P40–P40, IET, 2011. 10

[89] D. Mery, "Automated Detection In Complex Objects Using A Tracking Algorithm In Multiple X-ray Views," in *Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 41–48, IEEE, 2011. 10, 16

[90] D. Mery, G. Mondragon, V. Riffo, and I. Zuccar, "Detection Of Regular Objects In Baggage Using Multiple X-ray Views," *Insight - Non-Destructive Testing and Condition Monitoring*, vol. 55, no. 1, pp. 16–20, 2013. 10, 15

[91] D. Mery, V. Riffo, I. Zuccar, and C. Pieringer, "Automated X-Ray Object Recognition Using an Efficient Search Algorithm in Multiple Views," in *Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 368–374, IEEE, 2013. 10, 16

[92] D. Mery and V. Riffo, "Automated Object Recognition In Baggage Screening Using Multiple X-ray Views," *Annual Conference of the British Institute of Non-Destructive Testing (NDT)*, vol. 4860, no. 143, pp. 1–12, 2013. 10

[93] D. Mery, G. Mondragon, V. Riffo, and I. Zuccar, "Detection of regular objects in baggage using multiple X-ray views," *Insight - Non-Destructive Testing and Condition Monitoring*, vol. 55, pp. 16–20, mar 2013. 10, 16

[94] D. Mery, "Inspection of Complex Objects Using Multiple-X-Ray Views," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 1, pp. 338–347, 2015. 10

[95] D. Mery, E. Svec, and M. Arias, "Object Recognition in X-ray Testing Using Adaptive Sparse Representations," *Journal of Nondestructive Evaluation*, vol. 35, p. 45, mar 2016. 10, 16

[96] V. Riffo and D. Mery, "Automated Detection of Threat Objects Using Adapted Implicit Shape Model," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 4, pp. 472–482, 2016. 10, 16

[97] P. C. Cañizares, M. G. Merayo, and A. Núñez, "FORTIFIER: a FORmal disTrIbuted Framework to Improve the dEtection of thReatening objects in baggage," *Journal of Information and Telecommunication*, vol. 2, no. 1, pp. 2–18, 2018. 10

[98] L. Schmidt-Hackenberg, M. R. Yousefi, and T. M. Breuel, "Visual Cortex Inspired Features For Object Detection In X-ray Images," in *International Conference on Pattern Recognition (ICPR)*, (Tsukuba, Japan), pp. 2573–2576, IEEE, 2012. 10, 16

[99] T. Franzel, U. Schmidt, and S. Roth, "Object Detection in Multi-view X-Ray Images," in *Pattern Recognition: Joint DAGM and OAGM Symposium*, pp. 144–154, Springer Berlin Heidelberg, 2012. 10, 17, 32, 49, 51, 63, 102

[100] E. Svec P., *Sparse KNN - A Method For Object Recognition Over X-ray Images Using Knn Based In Sparse Reconstruction.* PhD thesis, Pontificia Universidad Catolica De Chile, 2016. 10, 16, 28

[101] N. Jaccard, T. W. Rogers, E. J. Morton, and L. D. Griffin, "Using Deep Learning On X-ray Images To Detect Threats," in *Defence and Security Doctoral Symposium Paper*, pp. 1–12, Cranfield University, 2016. 10, 19, 22, 28

[102] N. Jaccard, T. W. Rogers, and E. J. Morton, "Detection Of Concealed Cars In Complex Cargo X-ray Imagery Using Deep Learning," *Journal of X-Ray Science and Technology*, vol. 25, no. 3, pp. 323–339, 2017. 10, 19, 21, 22

[103] N. Jaccard, T. Rogers, E. Morton, and L. Griffin, "Automated Detection Of Smuggled High-risk Security Threats Using Deep Learning," in *International Conference on Imaging for Crime Detection and Prevention (ICDP)*, pp. 11 (4 .)–11 (4 .), IET, 2016. 10, 19, 21, 23

[104] T. W. Rogers, N. Jaccard, and L. D. Griffin, "A Deep Learning Framework For The Automated Inspection Of Complex Dual-energy X-ray Cargo Imagery," in *Conference on Anomaly Detection and Imaging with X-Rays (ADIX) II*, SPIE, 2017. 10, 19, 21, 23, 28, 106

[105] J. Yuan and C. Guo, "A Deep Learning Method for Detection of Dangerous Equipment," in *International Conference on Information Science and Technology (ICIST)*, pp. 159–164, IEEE, 2018. 10, 28

[106] Z. Zhao, H. Zhang, and J. Yang, "A GAN-Based Image Generation Method for X-Ray Security Prohibited Items," in *Chinese Conference on Pattern Recognition and Computer Vision (PRCV)*, Lecture Notes in Computer Science, pp. 420–430, Springer International Publishing, 2018. 10, 21, 23, 28, 104

[107] M. Xu, H. Zhang, and J. Yang, "Prohibited Item Detection in Airport X-Ray Security Images via Attention Mechanism Based CNN," in *Chinese Conference on Pattern Recognition and Computer Vision (PRCV)*, Lecture Notes in Computer Science, pp. 429–439, Springer International Publishing, 2018. 10, 21, 25, 28

[108] S. Akçay and T. P. Breckon, "An Evaluation Of Region-Based Object Detection Strategies Within X-ray Baggage Security Imagery," in *IEEE International Conference on Image Processing (ICIP)*, pp. 1337–1341, IEEE, 2017. 10, 18, 21, 25, 28

[109] J. Tuszynski, J. T. Briggs, and J. Kaufhold, "A Method For Automatic Manifest Verification Of Container Cargo Using Radiography Images," *Journal of Transportation Security*, vol. 6, no. 4, pp. 339–356, 2013. 10

[110] J. T. A. Andrews, N. Jaccard, T. W. Rogers, T. Tanay, and L. D. Griffin, "Anomaly Detection for Security Imaging," in *Defence and Security Doctoral Symposium*, Cranfield University, 2016. 10, 19, 21

[111] J. T. A. Andrews, N. Jaccard, T. W. Rogers, and L. D. Griffin, "Representation-learning For Anomaly Detection In Complex X-ray Cargo Imagery," in *Anomaly Detection and Imaging with X-Rays (ADIX) II*, SPIE, 2017. 10, 19, 21, 28

[112] J. Chan, P. Evans, and X. Wang, "Enhanced Color Coding Scheme For Kinetic Depth Effect X-ray (KDEX) Imaging," in *International Carnahan Conference on Security Technology*, pp. 155–160, IEEE, 2010. 12

[113] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001. 12, 14, 23

[114] A. Schwaninger, F. Hofer, and O. E. Wetter, "Adaptive Computer-Based Training Increases on the Job Performance of X-Ray Screeners," in *International Carnahan Conference on Security Technology*, pp. 117–124, IEEE, 2007. 13

[115] V. Cutler and S. Paddock, "Use Of Threat Image Projection (TIP) To Enhance Security Performance," in *International Carnahan Conference on Security Technology*, pp. 46–51, IEEE, 2009. 13

[116] D. Mery, *Computer Vision for X-Ray Testing*. Cham: Springer International Publishing, 2015. 14

[117] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004. 14, 15, 16, 18

[118] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-Up Robust Features (SURF)," *Computer Vision and Image Understanding*, vol. 110, pp. 346–359, jun 2008. 14

[119] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A K-Means Clustering Algorithm," *Applied Statistics*, vol. 28, no. 1, p. 100, 1979. 14, 16

[120] M. Hearst, S. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support Vector Machines," *IEEE Intelligent Systems and their Applications*, vol. 13, pp. 18–28, jul 1998. 14, 15, 16, 17, 23, 26

[121] L. D. Griffin, M. Lillholm, M. Crosier, and J. van Sande, "Basic Image Features (BIFs) Arising from Approximate Symmetry Type," pp. 343–355, Springer, Berlin, Heidelberg, 2009. 14, 15, 23

[122] M. A. Fischler and R. C. Bolles, "Random Sample Consensus: A Paradigm For Model Fitting With Applications To Image Analysis And Automated Cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981. 15, 18

[123] A. Dixit, "Adaptive kmeans Clustering for Color and Gray Image. - File Exchange - MATLAB Central," 2014. 16

[124] T. Cover and P. Hart, "Nearest Neighbor Pattern Classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, 1967. 16, 17

[125] S. Michel and A. Schwaninger, "Human-machine Interaction In X-ray Screening," in *International Carnahan Conference on Security Technology*, pp. 13–19, IEEE, 2009. 16, 105

[126] C. C. V. Bastian, A. Schwaninger, and S. Michel, "Do Multi-view X-ray Systems Improve X-ray Image In- Terpretation In Airport Security Screening ?," *Zeitschrift für Arbeitswissenschaft 3*, no. December, pp. 166–173, 2008. 16

[127] D. Mery, V. Riffo, I. Zuccar, and C. Pieringer, "Object Recognition In X-ray Testing Using An Efficient Search Algorithm In Multiple Views," *Insight - Non-Destructive Testing and Condition Monitoring*, vol. 59, no. 2, pp. 85–92, 2017. 16, 32

[128] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, pp. 886–893, IEEE, 2005. 17

[129] S. Dasgupta, "Learning Mixtures of Gaussians," in *Annual Symposium on Foundations of Computer Science*, pp. 634–, IEEE, 1999. 17

[130] K. Mikolajczyk and C. Schmid, "A Performance Evaluation Of Local Descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005. 18

[131] 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*, vol. 34, no. 4, p. 42, 2015. 19

[132] D. Mery and C. Arteta, "Automatic Defect Recognition in X-Ray Testing Using Computer Vision," in *Winter Conference on Applications of Computer Vision (WACV)*, pp. 1026–1035, IEEE, 2017. 21

[133] Dhiraj and D. K. Jain, "An Evaluation Of Deep Learning-Based Object Detection Strategies For Threat Object Detection In Baggage Security Imagery," *Pattern Recognition Letters*, vol. 120, pp. 112–119, 2019. 21

[134] T. Morris, T. Chien, and E. Goodman, "Convolutional Neural Networks for Automatic Threat Detection in Security X-Ray Images," in *International Conference on Machine Learning and Applications (ICMLA)*, pp. 285–292, IEEE, 2018. 21, 24, 28, 106

[135] A. Bosch, A. Zisserman, and X. Munoz, "Representing Shape With A Spatial Pyramid Kernel," in *International Conference On Image And Video Retrieval (CIVR)*, pp. 401–408, ACM Press, 2007. 23

[136] Q. Chen, D. Li, and C.-K. Tang, "KNN Matting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, pp. 2175–2188, sep 2013. 23

[137] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," in *International Conference on Machine Learning (ICML)*, pp. 214—-223, PMLR, 2017. 23, 67

[138] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-Image Translation with Conditional Adversarial Networks," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5967–5976, IEEE, 2017. 23, 71, 104

[139] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium," in *Conference on Neural Information Processing Systems (NeurIPS)*, pp. 6629–6640, Curran Associates, Inc., 2017. 23

[140] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1800–1807, IEEE, 2017. 24

[141] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818–2826, IEEE, 2016. 24, 128

[142] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object Detection via Region-based Fully Convolutional Networks," in *International Conference on Neural Information Processing Systems (NeurIPS)*, pp. 379–387, 2016. 25, 102

[143] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single Shot MultiBox Detector," in *European Conference on Computer Vision (ECCV)*, pp. 21–37, Springer, Cham, 2016. 25

[144] Y. Gaus, N. Bhowmik, S. Akcay, and T. Breckon, "Evaluating the Transferability and Adversarial Discrimination of Convolutional Neural Networks for Threat Object Detection and Classification within X-Ray Security Imagery," in *Procedings of the International Conference on Machine Learning Applications*, IEEE, 2019. 25, 27, 105

[145] Y. Sterchi, N. Hättenschwiler, S. Michel, and A. Schwaninger, "Relevance of visual inspection strategy and knowledge about everyday objects for X-ray baggage screening," in *International Carnahan Conference on Security Technology (ICCST)*, pp. 1–6, IEEE, 2017. 26

[146] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation-Based Anomaly Detection," *ACM Transactions on Knowledge Discovery from Data*, vol. 6, no. 1, pp. 1–39, 2012. 26

[147] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning," in *AAAI Conference on Artificial Intelligence*, 2017. 27, 128

[148] A. Islam, Y. Zhang, D. Yin, O. Camps, and R. J. Radke, "Correlating Belongings with Passengers in a Simulated Airport Security Checkpoint," in *International Conference on Distributed Smart Cameras (ICDSC)*, pp. 1–7, ACM Press, 2018. 28

[149] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the Devil in the Details: Delving Deep into Convolutional Nets," in *Proceedings of the British Machine Vision Conference 2014*, pp. 6.1–6.12, British Machine Vision Association, 2014. 32, 39, 40, 41

[150] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks," in *International Conference on Learning Representations (ICLR)*, 2014. 32, 49, 51, 54, 63, 102

[151] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and Transferring Mid-Level Image Representations using Convolutional Neural Networks," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1717–1724, IEEE, 2014. 33

[152] C. Szegedy, Wei Liu, Yangqing Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, IEEE, 2015. 33, 127

[153] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010. 51

[154] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 580–587, IEEE, 2014. 54

[155] A. Abdallah, M. A. Maarof, and A. Zainal, "Fraud detection system: A survey," *Journal of Network and Computer Applications*, vol. 68, pp. 90–113, 2016. 66

[156] M. Ahmed, A. Naser Mahmood, and J. Hu, "A Survey Of Network Anomaly Detection Techniques," *Journal of Network and Computer Applications*, vol. 60, pp. 19–31, 2016. 66

[157] M. Ahmed, A. N. Mahmood, and M. R. Islam, "A Survey Of Anomaly Detection Techniques In Financial Domain," *Future Generation Computer Systems*, vol. 55, pp. 278–288, 2016. 66

[158] B. Kiran, D. Thomas, and R. Parakkal, "An Overview of Deep Learning Based Methods for Unsupervised and Semi-Supervised Anomaly Detection in Videos," *Journal of Imaging*, vol. 4, no. 2, p. 36, 2018. 66, 67

[159] M. Markou and S. Singh, "Novelty Detection: A Review—part 1: Statistical Approaches," *Signal Processing*, vol. 83, no. 12, pp. 2481–2497, 2003. 66

[160] M. Markou and S. Singh, "Novelty Detection: A Review—part 2: Neural Network Based Approaches," *Signal Processing*, vol. 83, no. 12, pp. 2499–2521, 2003. 66

[161] V. Hodge and J. Austin, "A Survey of Outlier Detection Methodologies," *Artificial Intelligence Review*, vol. 22, no. 2, pp. 85–126, 2004. 66

[162] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection," *ACM Computing Surveys*, vol. 41, no. 3, pp. 1–58, 2009. 66

[163] M. A. Pimentel, D. A. Clifton, L. Clifton, and L. Tarassenko, "A review of novelty detection," *Signal Processing*, vol. 99, pp. 215–249, 2014. 66

[164] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Nets," in *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 2672—-2680, 2014. 66, 89, 134

[165] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved Training of Wasserstein GANs," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017. 67

[166] J. Donahue, P. Krähenbühl, and T. Darrell, "Adversarial Feature Learning," in *International Conference on Learning Representations (ICLR)*, 2017. 67

[167] Y. LeCun, C. Cortes, and C. J. Burges, "MNIST handwritten digit database, Yann LeCun, Corinna Cortes and Chris Burges," 2010. 67, 73, 83, 103

[168] H. Xu, Y. Feng, J. Chen, Z. Wang, H. Qiao, W. Chen, N. Zhao, Z. Li, J. Bu, Z. Li, Y. Liu, Y. Zhao, and D. Pei, "Unsupervised Anomaly Detection via Variational Auto-Encoder for Seasonal KPIs in Web Applications," in *World Wide Web Conference on World Wide Web (WWW)*, pp. 187–196, ACM Press, 2018. 67, 74, 75

[169] A. Radford, L. Metz, and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," in *International Conference on Learning Representations (ICLR)*, 2016. 69, 70, 74, 88

[170] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved Techniques for Training GANs," in *International Conference on Neural Information Processing Systems (NeurIPS)*, pp. 2234–2242, 2016. 70, 89

[171] A. Krizhevsky, "Learning Multiple Layers of Features from Tiny Images," tech. rep., University of Toronto, 2009. 73, 83, 85, 91, 97, 103

[172] A. Paszke, S. Gross, and A. Lerer, "Automatic differentiation in PyTorch," in *Advances in Neural Information Processing Systems (NeurIPS) Workshops*, 2017. 74, 92

[173] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *International Conference on Learning Representations (ICLR)*, 2015. 74, 92

[174] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI*, pp. 234–241, Springer, 2015. 84, 97, 103, 132

[175] C. X. Ling, J. Huang, and H. Zhang, "AUC," in *International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 519–524, 2003. 92

[176] N. Bhowmik, Q. Wang, Y. F. A. Gaus, M. Szarek, and T. P. Breckon, "The Good, the Bad and the Ugly: Evaluating Convolutional Neural Networks for Prohibited Item Detection Using Real and Synthetically Composited X-ray Imagery," in *British Machine Vision Conference (BMVC) Workshops*, 2019. 104

[177] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired Image-to-Image Translation," in *International Conference on Computer Vision (ICCV)*, pp. 2223—-2232, IEEE, 2017. 104, 105

[178] T. Karras, S. Laine, and T. Aila, "A Style-Based Generator Architecture for Generative Adversarial Networks," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2019. 105

[179] G. Chen, G. Bennett, and D. Perticone, "Dual-energy X-ray Radiography For Automatic High- Z Material Detection," *Nuclear Instruments and Methods in Physics Research B*, vol. 261, pp. 356–359, 2007. 106

[180] K. Fu, D. Ranta, P. Das, and C. Guest, "Layer Separation For Material Discrimination Cargo Imaging System," in *Image Processing: Machine Vision Applications III*, vol. 7538, p. 75380Y, SPIE, 2010. 106

[181] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," in *International Conference on Machine Learning (ICML)*, 2015. 126, 128

[182] G. Hinton, N. Srivastava, and K. Swersky, "Neural Networks for Machine Learning Lecture 6a Overview of mini--batch gradient descent," tech. rep. 128

[183] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788, IEEE, 2016. 131

APPENDIX A

Fundamentals of Deep Learning Approaches in X-ray Security Imaging

This section briefly introduces the recent deep learning approaches employed within the X-ray security domain. It is important to note that the scope of the deep learning approaches discussed here is limited to those only used in X-ray security imaging.

A.1 Background on Neural Networks

One of the fundamental neural network approaches, called multi-layer perceptron consists of a single layer h or stack of n multiple layers $\mathbf{h} = \{h_0, h_1, \dots h_n\}$, each of which comprises of set of neurons, activations (a) and non-linear transformation (σ). An activation of a layer i, denoted as $a^{(i)}$ is the linear combination of the input $\mathbf{x}^{(i-1)}$ and parameters $\theta^{(i)} = \{\mathbf{W}^{(i)}, b^{(i)}\}$, where $\mathbf{W}^{(i)}$ and $b^{(i)}$ are the weights and the biases such that $a^{(i)} = \mathbf{w_i^T x^{(i-1)}} + b^{(i)}$. The i^{th} layer $h^{(i)}$ is a function, where a non-liner transformation $\sigma^{(i)}$ is applied to the activation $a^{(i)}$ such that $h^{(i)} = \sigma^{(i)}(a^{(i)})$. Hence, the output of the i^{th} layer is $h^{(i)} = \sigma^{(i)}(\mathbf{w_i^T x} + b)$.

Overall, an n layer network, comprises of n hidden layers, where $\mathbf{h} = \{h_0, h_1, \dots h_n\}$, and parameters $\Theta = \{\mathcal{W}, \mathcal{B}\}$, where $\mathcal{W} = \{\mathbf{w_1}, \mathbf{w_2}, \dots, \mathbf{w_n}\}$ and $\mathcal{B} = \{b_1, b_2, \dots, b_n\}$.

The overall network f is the composition of the hidden layers such that

$$f(\mathbf{x};\Theta) = f(h_n \circ \ldots \circ h1 \circ h_0). \tag{A.1.1}$$

The final layer $h^{(n)}$ of the model f outputs C vectors, where C is the number of classes within the dataset $\mathcal{D}$.

For an n layer network f, the output activation is $a^{(n)} = \mathbf{w}_n^T \mathbf{x}^{(n-1)} + b^{(n)}$. For a simpler notation let $z = a^{(n)}$. The network outputs C vectors $\mathbf{z} = \{z_1, z_2, \ldots, z_C\}$, where C is the number of classes, and each z is the feature encoding for class j. The next step is to classify the input x, based on its feature encoding $\mathbf{z}$. $softmax$ is the most common function used to classify the feature encoding in neural networks. A $softmax$ function takes an input feature encoding $\mathbf{z}$, and returns a probabilistic output, representing the likelihood of the input belonging to class j. Hence, the $softmax$ output of z for class j is:

$$P(y_j|\mathbf{x};\Theta) = \sigma(z)_j = \frac{e^{z_j}}{\sum_{c=1}^{C} e^{z_c}}, \quad j = 1, \ldots, C. \tag{A.1.2}$$

The performance of a task is optimised via an objective (loss) function. In a classification task, for instance, cross-entropy is used to measure the performance of the probabilistic output of the $softmax$ function. Cross-entropy, also known as log-loss, penalises the model as the predicted probability deviates from the ground-truth. The mathematical definition of this loss function is as follows:

$$\mathcal{L} = \underset{\Theta}{\operatorname{argmin}} \Big(- \sum_{c=1}^{C} y_c log(p_c) \Big), \tag{A.1.3}$$

where y_c is a binary label indicating whether the label c is correct for the sample, and p_c is the probability predicted for the class c by $softmax$.

A.2 Convolutional Neural Networks – CNN

Convolutional neural networks (CNN) are considered to be modern neural networks with key distinctions. Unlike MLP that connect all neurons to other neurons, which is impractical, CNN uses local receptive fields, also known as filter or kernels, that

spatially connect neurons to their local region. The use of local receptive fields makes CNN equivariant to image translations. Each layer h consists of K kernels with weights $\mathcal{W} = \{\mathbf{w_1}, \mathbf{w_2}, \ldots, \mathbf{w_K}\}$ and biases $\mathcal{B} = \{b_1, b_2, \ldots, b_K\}$. The second major difference is weight sharing that shares weights of the filters $\mathcal{W}$ across individual feature map of each layer h. Weight sharing radically decreases the number of parameters needed to train a deep neural network.

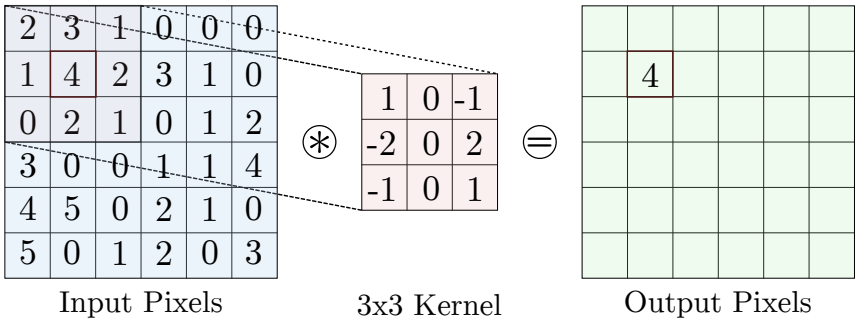


Figure A.1: 2D Convolutional operation. Output is the linear combination of $n \times n$ kernel and the corresponding pixels slid through the entire input.

Another difference is that, within the network, convolutional layers are usually followed by pooling layers which down-samples the current representation (image) and hence reduces the number of parameters carried forward in-addition to improving overall computational efficiency.

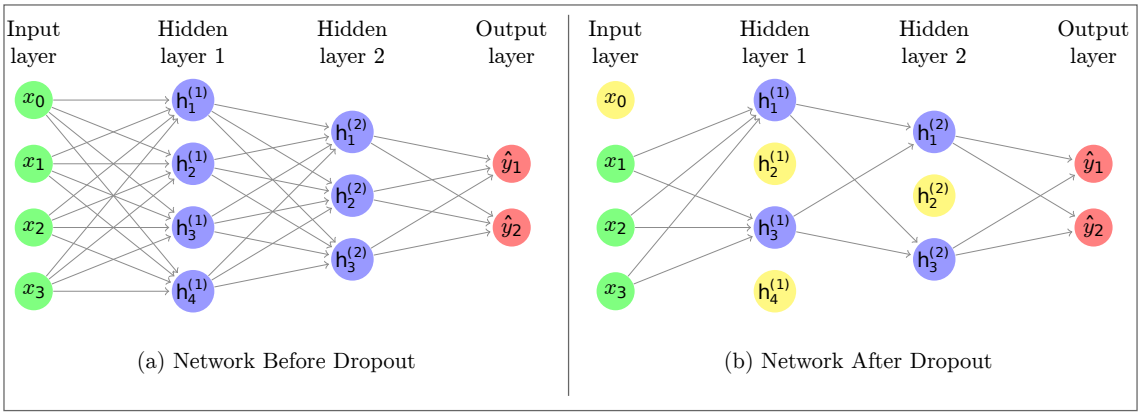


Figure A.2: Application of dropout, whereby the neurons are randomly removed from the network to avoid over-fitting.

The high-level of parametrisation, and hence representational capacity, make CNN susceptible to over-fitting in the traditional sense. The use of dropout [7], whereby hidden neurons are randomly removed during the training process, is used

to avoid over-fitting such that performance dependence on individual network elements is reduced in favour of cumulative error reduction and representational responsibility for the problem space.

The design of deep CNN poses instability issues during training. The use of batch normalisation, called BatchNorm, [181], whereby its non-linearity normalises the input for each hidden layer resolves the stability issues. For the i^{th} hidden layer $h^{(i)}$, the output of the layer would typically be

$$\begin{aligned} x^{(i)} &= \sigma_i(\mathbf{w_i^T} x^{(i-1)}) \\ &= \sigma_i(a^{(i)}) \end{aligned} \tag{A.2.4}$$

where σ is the non-linearity function, and $\mathbf{w_i^T} x^{(i-1)}$ is activation $a^{(i)}$. BatchNorm normalises the activation $a^{(i)}$ such that

$$\tilde{a}^{(i)} = \frac{a^{(i)} \mathbb{E}[a^{(i)}]}{\sqrt{\mathbb{V}[a^{(i)}]}}, \tag{A.2.5}$$

where $\mathbb{E}[a^{(i)}]$ and $\mathbb{V}[a^{(i)}]$ are the mean and the variance of the activation $a^{(i)}$, respectively. Normalising the outputs of the layer based on the above equations minimises the massive gradients during the optimisation, and hence leads to faster convergence. All of the unique differences of CNN listed above make them much more efficient and reliable compared to traditional MLP. The following subsections introduce well-known CNN strategies proposed for classification, detection, segmentation, and also applied within X-ray security imaging.

A.3 Supervised CNN Architectures

A.3.1 Classification Architectures

This section explores the contemporary classification strategies proposed during the deep learning era, and applied within X-ray security imaging.

AlexNet

Similar to [54] but deeper and wider, this network is of 5 *conv* layers with 11×11 receptive filters and 3 fc layers, and 60 million parameters in total. To eliminate the network's tendency to over-fitting, caused by the high number of parametrisation, the network employs the use of dropout, by randomly removing neurons during the training. Besides, the network utilises ReLUs for non-linearity to accelerate the training process.

VGG

Following the AlexNet's breakthrough within the field, Simonyan and Zisserman (VGG) [5] investigate the depth on classification performance by designing CNN via stacking convolutional layers with small 3×3 receptive fields with a stride of 1. Small receptive fields increase non-linearity of the network but decrease the total number of the parameters of the network. This design choice is shown to improve state-of-the-art significantly.

ResNet

Proposed by He *et al.* [1], this architecture is also designed to train deeper networks via its residual connections. The difficulty of training deep networks is that the training becomes unstable as the network goes deeper, which is due to losing the gradients of the input. He *et al.* propose a residual connection approach that forward-passes input to a stack of two convolutional layers (residual) and sums it with the input (identity). With the residual connection, the network not only keeps the gradient of the input as it goes deeper, but also it learns the residuals to be the same as the input. Empirical evidence demonstrates that ResNet could be trained up to 1000 layers.

Inception

Instead of only deepening, Szegedy *et al.* [152] design a network by widening the architecture. For each layer, the network uses three different filter sizes (i.e., 1×1, 3×3, 5×5) and 3×3 max pooling layer. The overall network is designed to be upto 22 layers. This design overcomes the scaling issue of the objects within the image.

The next version of the paper, named *InceptionV2* [141], considers factorizing the receptive fields such that $n \times n$ filter is reformulated as the stack of $1 \times n$ and $n \times 1$ filters, found to be 33% cheaper. The third version of the network utilises RMSProp [182], 7×7 factorized filters, BatchNorm [181] and label smoothing to avoid over-fitting. The final modification made to the network [147] is to use residual blocks as proposed in [1].

A.3.2 Detection Architectures

Here we introduce state-of-the-art object detection strategies utilized in X-ray security imaging.

Sliding Window-based CNN

As shown in Figure A.3a, this classical object detection approach consists of two main stages: (i) object proposal (ii) object classification. The first stage generates objects of interests via a fixed-sized $n \times n$ window slides over the image horizontally and vertically. One issue of using a fixed size window is large objects within the image does not fit within the frame. A possible solution to overcome this issue is to use image pyramids by a multi-scale sampling of the image and subsequent image interpolation of window regions at differing scale to a fixed size classification region input size.

Faster Region-based CNN (F-RCNN)

Proposed by Ren *et al.* [10], and depicted in Figure A.3, F-RCNN is designed to have two sub-networks: (i) Region Proposal Network and (ii) Fast RCNN network [16]. RPN network generates objects of interests with varying anchors by sliding a 3×3 window through the convolutional feature map, shared with the Fast RCNN sub-network (Figure A.3b). Based on the feature map from the convolutional layer, a set of fully connected layers predicts the bounding box and objectness score of the region (*an object* or *background*). The Region of Interest (RoI) pooling layer then resizes the regions of interests generated by the RPN with varying aspect ratios. fc layers then create the final feature map to be used by bounding box regression and $softmax$ layers.

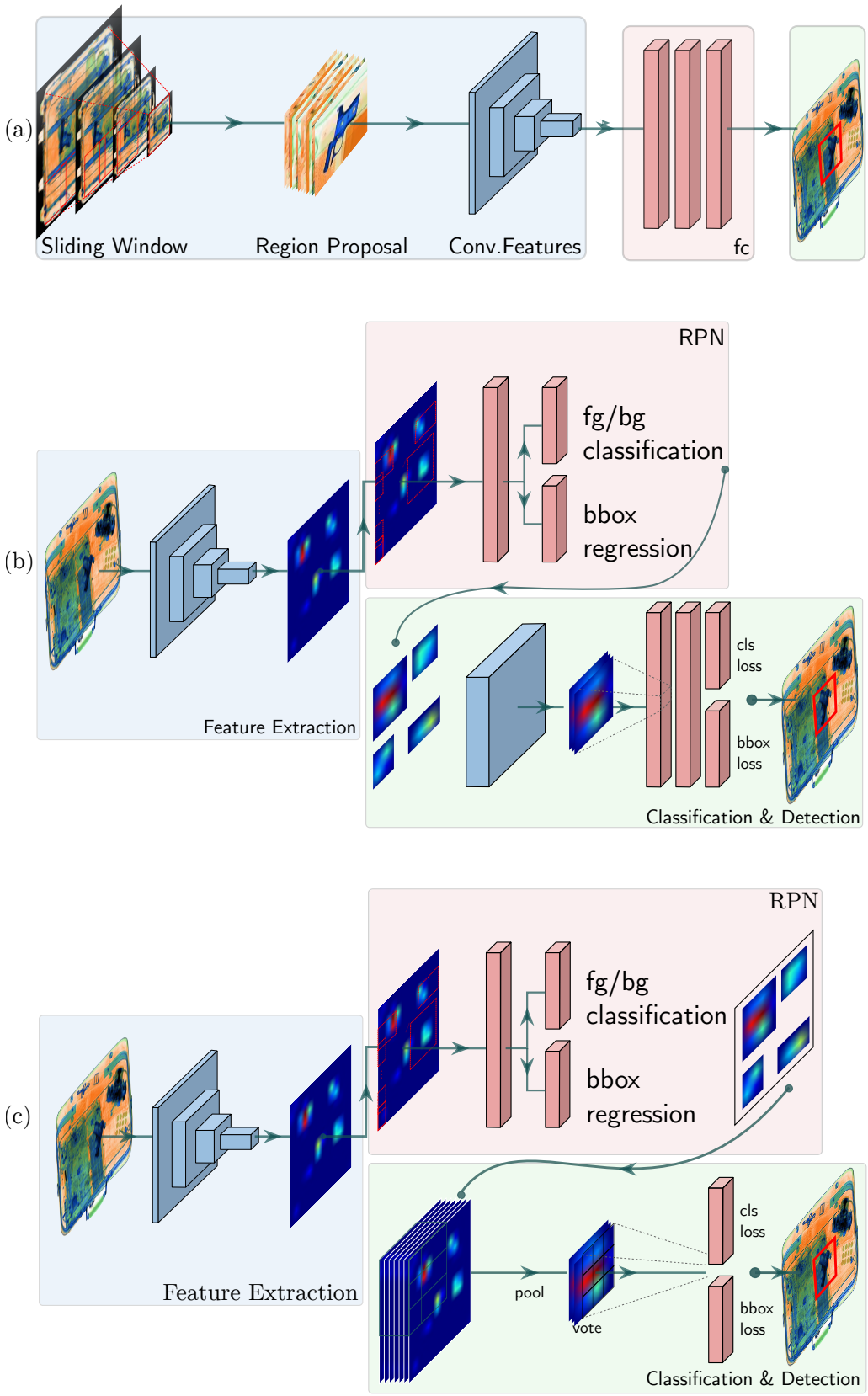


Figure A.3: Region-based fully convolutional neural network (R-FCN), proposed by [10], removes fully-connected fc layers from F-RCNN to accelerate training.

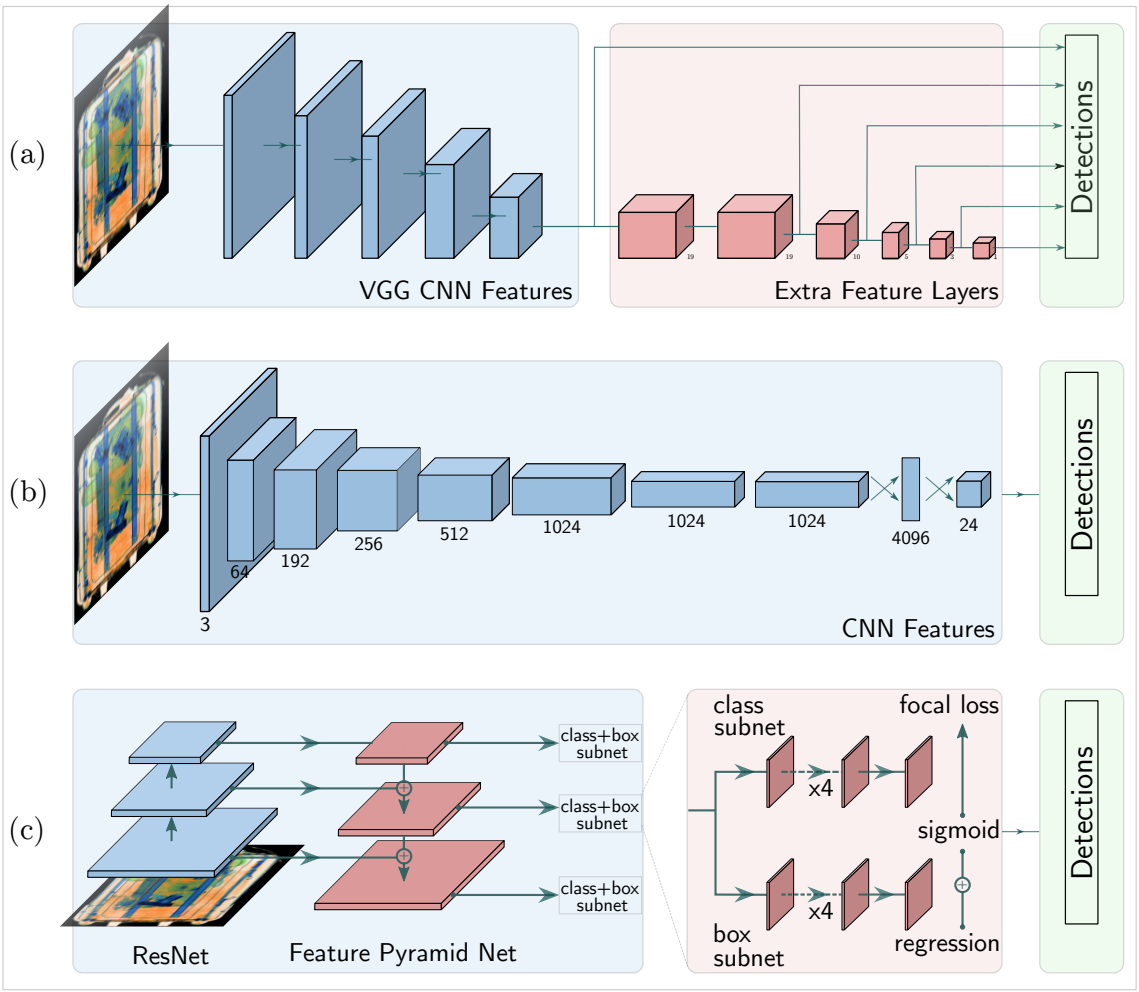


Figure A.4: The pipeline of the Single Shot Multi-Box Detector.

Region-based Fully CNN (R-FCN)

Proposed by Dai et al. [11], R-FCN removes the fc layers from F-RCNN to accelerate the training. The use of fc layers in F-RCNN leads to multiple computations of the region proposals, which is rather expensive (Figure A.3c). Instead of using fc layers, R-FCN employs convolutional layers, and a unique scoring map, called *position sensitive scoring map* [11], which achieves the similar performance much more efficiently.

Single Shot Multi-Box Detector (SSD)

As the name suggests, this approach performs object detection with single forward-pass, without the need for another region proposal sub-network. The architecture utilises a VGG network by replacing its fc layers with convolutional layers, which

helps to extract features from multiple scales, and also to reduce the size of the output for the next layer. The input image is split into 4×4 and 8×8 feature maps, whereby six manually configured bounding boxes are predicted per feature map cell. The objective of the architecture is to optimise a loss function that combines objectness loss and bounding box location loss, computed via a cross-entropy and $\mathcal{L}_1$ losses, respectively. Training the model based on the proposed loss yields statistically reliable and computationally efficient results.

YOLO

Similar to SSD, Redmon *et al.* [183] also propose a fully convolutional object detection network that needs only one forward-pass (Figure A.4b). Similar to Faster RCNN, YOLO also utilises several anchors to handle the objects with varying aspect ratios. Unlike Faster RCNN that uses fixed size anchors, however, YOLO clusters the ground-truth bounding boxes via k-means clustering to learn the data specific anchor parameters. Minor modifications to the approach such as BatchNorm and the use of higher resolution input images together with multi-scale training yield better detection performance. For instance, the network can train images of sizes that range between 350×350 to 600×600. It divides the input into 13×13 grid cells, each of which predicts 5 bounding box coordinates for each anchor. Moreover, for individual predicted bounding boxes, the network outputs confidence score showing the similarity between the bounding boxes and the ground truth. Finally, the output also includes the probability distribution of the classes that the predicted bounding boxes belong.

RetinaNet

Despite the speed, the downside of single-shot detection algorithms, introduced in Sections A.3.2 and A.3.2, is the poorer detection performance compared to region-based approaches explained in Sections A.3.2 and A.3.2. Another issue is the imbalanced class size as the number of background samples is significantly higher than objects of interests, which adversely dominates the loss. RetinaNet addresses these issues and is based on two main contributions: (i) Feature Pyramid Networks

(FPN), (ii) Focal Loss. The notion of FPN is somewhat similar to UNet [174], whereby high-level features (first layers) and low-level (higher layers) feature maps are combined. The advantage of using focal loss during training is the elimination of class imbalance posed by a large number of background samples. Focal-loss weights the well-classified samples for the objective function to focus more on hard and interesting examples to learn. Hence the proposed method addresses the issues of the single-shot methods and achieves comparable performance to region-based detection approaches.

A.3.3 Segmentation Architectures

The use of segmentation approaches in X-ray security imaging is limited due to the expense of collecting large segmentation datasets. Recently, there are few recent research applying segmentation algorithms within this domain, all of which utilises Mask-RCNN for object segmentation. Proposed by He *et al.* [3], Mask-RCNN is an instance segmentation algorithm that simultaneously performs detection and segmentation to each object within an image. The proposed approach is composed of two main stages and utilises two well-established detection and segmentation algorithms. The first stage uses F-RCNN detection strategy to perform object detection via the box regression and classification layers (Figure A.3b). The second stage is the *mask branch* that is a binary mask classifier that classifies each pixel within a bounding box as a target class (cat, dog, etc.) or background. Combining binary mask classifier with F-RCNN (Figure A.5) detection module yields state-of-the-art instance segmentation results.

A.4 Unsupervised CNN Architectures

Previous approaches introduced in Section A.2 are supervised learning techniques such that the input dataset contains ground-truth labels, and the model is trained to predict the labels. In some applications, however, the ground-truth labels are not available within the dataset, also known as unsupervised learning. This section introduces unsupervised learning approaches.

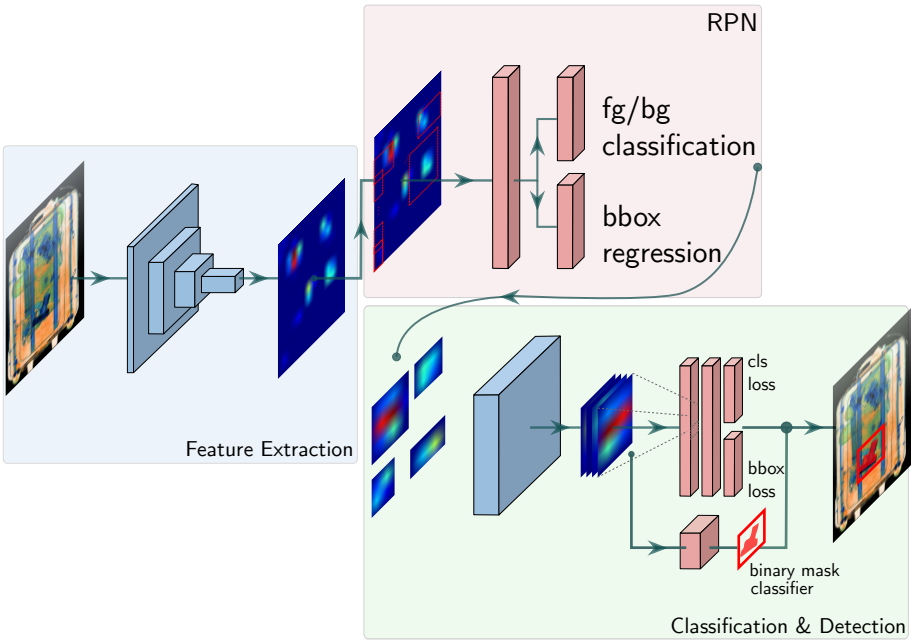


Figure A.5: Mask-RCNN pipeline. The architecture simultaneously performs detection and instance segmentation.

A.4.1 Autoencoders

An autoencoder is an unsupervised neural network that encodes an input x to a lower-dimensional latent space z, then reconstructs the input from the latent space via a decoder network (Figure A.6). The objective here is to learn a model that is

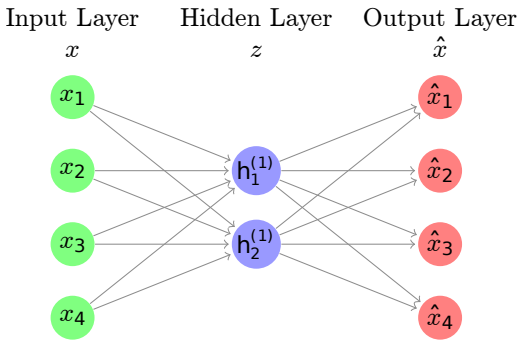


Figure A.6: An autoencoder pipeline. The input is reduced to a smaller dimension, which is subsequently reconstructed back to its original dimensionality.

capable of reconstructing the output $\hat{x}$ as close to input x as possible. Hence, the optimisation problem becomes

$$\mathcal{L} = \underset{\Theta}{\operatorname{argmin}} |x - \hat{x}|. \tag{A.4.6}$$

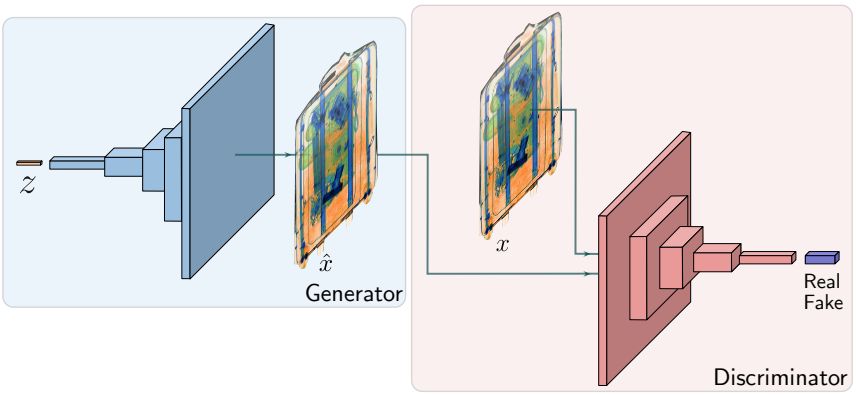


Figure A.7: A generative adversarial network. The generator network produces high dimensional output from a low-dimensional noise vector, while the discriminator network classifies the real and reconstructed images

A.4.2 Generative Adversarial Networks (GAN)

Initially proposed by Goodfellow *et al.* [164], GAN are unsupervised deep neural architectures that learn to capture any input data distribution by predicting features from an initially hidden representation z. The theory behind GAN are based on a competition between the two networks within a zero-sum game framework, as initially used in game theory. The first network, called *Generator* (G), aims to generate high dimensional output from a low-dimension latent space, which is commonly a random noise vector. The use of a decoder-alike network architecture upsamples the latent vector to a higher-dimensional feature map.

The second network, called *Discriminator* (D), measures the similarity between the original input (*real*) and the generated output (*fake*). The discriminator network usually adopts an encoder network architecture such that for a given high-dimensional feature, it predicts its class label. The objective during the training is that the generator aims to produce as realistic output as possible, while the discriminator tries to classify the two images as *real correctly* or *fake*. With optimisation based on a zero-sum game framework, each network strengthens its prediction capability until they reach an equilibrium. The task here, hence, is the minimisation and the maximisation of G and D, respectively. The overall loss function for this objective is as follows:

$$\min_{G} \max_{D} \mathop{\mathbb{E}}_{x \sim p_x} [\log D(x)] + \mathop{\mathbb{E}}_{z \sim p_z} [\log(1 - D(G(z))]. \tag{A.4.7}$$