International Journal of Innovation and Applied Studies
ISSN 2028-9324 Vol. 48 No. 2 Apr. 2026, pp. 485-497

http://www.ijias.issr-journals.org/

Design of a system capable of classifying suspicious plastic luggage with reasonable accuracy using pre-trained CNNs within the framework of X-ray image classification

Konan Trinité Boca[1], Konan Hyacinthe Kouassi[2], Allani Jules[1], and Asseu Olivier[1-2]

[1]INPHB, EDP-STI, Côte d'Ivoire

[2]ESATIC, LASTIC, Côte d'Ivoire

ABSTRACT: Accurately detecting threats such as plastic firearms presents a complex challenge in modern security systems due to the difficulty in distinguishing these objects from harmless ones when examined using X-ray scanners. This paper explores CNN architecture and image projection methods to compare systems capable of classifying plastic firearms with high accuracy. The results show that integrating data from three sources (a Stream of Commerce dataset, staged images, and synthetically produced images) was crucial for achieving satisfactory classification performance. We also reveal that to improve accuracy and generalization, it is important to expand the training dataset and explore more advanced neural networks, despite the limitations imposed by available computing power. Future work could include exploring the need for multiple views of the baggage examined and the use of more sophisticated imaging technologies, such as CT scanners, to further improve detection capabilities.

KEYWORDS: convolutional neural networks (CNNs), image processing techniques.

1 INTRODUCTION

Detecting and classifying dangerous objects, such as plastic firearms, which resemble harmless objects, is a crucial challenge for airport security. This article explores existing convolutional neural networks (CNNs) to improve the detection of suspicious objects (such as firearms) in X-ray images. Despite advances in imaging and machine learning, recognizing plastic objects remains difficult due to limitations in algorithms and datasets. This article aims to optimize detection by developing robust methods for data collection, improving the accuracy of pre-trained CNNs, and using image projection techniques based on Poisson differential equations. The results show promise for strengthening security systems. Challenges remain in adapting the methods to new threats and improving accuracy with larger datasets.

2 PRESENTATION OF THREE PRE-TRAINED MODELS

Various available CNN architectures (such as AlexNet, VGGNet, GoogleNet, ResNet, etc.) are capable of learning complex features from data and achieving peak performance on a variety of tasks. They are also effective at capturing the spatial and temporal features of videos. However, they are computationally intensive systems that require GPUs for efficient training. Here, we present three convolutional neural networks with different architectures (AlexNet, GoogleNet, and VGGNet). We will parameterize these networks (in Section 4) to learn the small features of X-ray images. Although these three networks are trained on the ImageNet natural image dataset [1] - [2] - [3], with a new, diverse dataset, we can modify all of their parameters to make them suitable for a new image style. This is possible by using a higher number of epochs and lower training rates. The initial parameters will be modified to accomplish this new task, and the selection of initial values will be of great importance for accelerating the convergence of the learning model. Numerous sources, such as [4] - [5], have noted that although these networks are trained on optical images, they have successfully classified metallic threats from X-ray baggage after fine-tuning with a smaller dataset. The strategy we use in this paper is to determine the dataset size that allows the network to modify the parameters and converge successfully. We will test our four dataset combinations on these three networks, as each network differs in how it learns, the time required for convergence, and its ability to classify complex images. A brief description of each network is summarized as follows:

2.1 THE ALEXNET NEURAL NETWORK

Alexnet is one of the first CNNs created (2012). Although it consists of only 8 layers (5 filters and 3 fully connected layers), it has many parameters to train (+60 million) (see Figure 01).

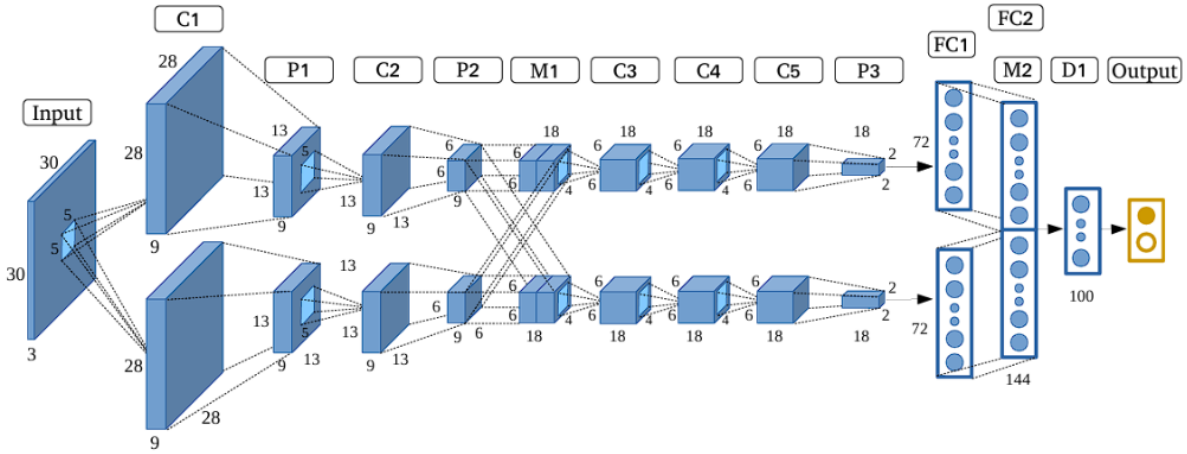

Fig. 1. AlexNet-type architecture (source: [6])

AlexNet has the following advantages: it is quick to train; quick to test; and has a simple architecture.

2.2 THE GOOGLENET NEURAL NETWORK

Googlenet is one of the latest models, created by Google researchers in 2014 [1]. It is based on inception layers and 1x1 convolution filters. It has fewer parameters to train (+4 million) compared to Alexnet. However, it is slower to train than Alexnet [2] (see Figure 02).

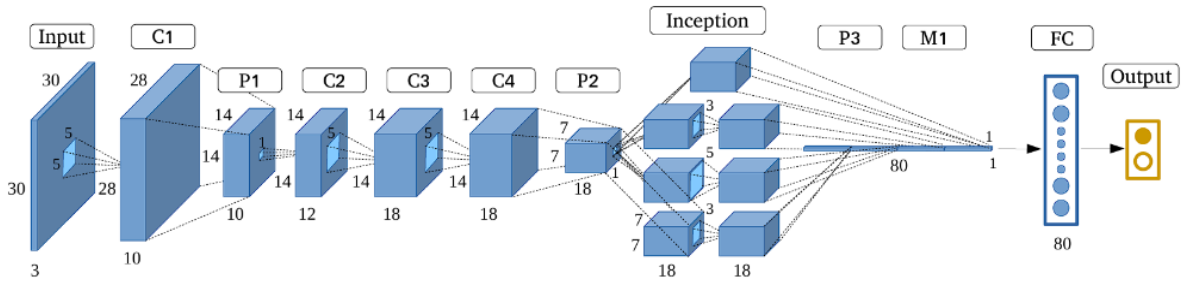

Fig. 2. VGG-16 Network Architecture (source: [6])

VGG-16 has the following advantages: VGG-16's error rate is lower than Alexnet's in the ILSVRC competition; it is deeper (22 layers); and it has a small memory size (20 MB).

2.3 THE VGG NEURAL NETWORK

VGG was created by researchers at the University of Oxford and was a finalist in the ILSVRC competition [3]. The network has a depth of 16 layers but contains approximately 130 million parameters, so any one of these parameters could be difficult to manage in our application.

VGG's error rate is 7.3%. The architecture demonstrates its simplicity by using only 3×3 convolutional layers stacked on top of each other with increasing depth. Furthermore, VGG was also a finalist in the same competition that GoogLeNet won. Although VGG used small convolution filters, its number of parameters and calculations were intensive compared to GoogLeNet (see Figure 03).

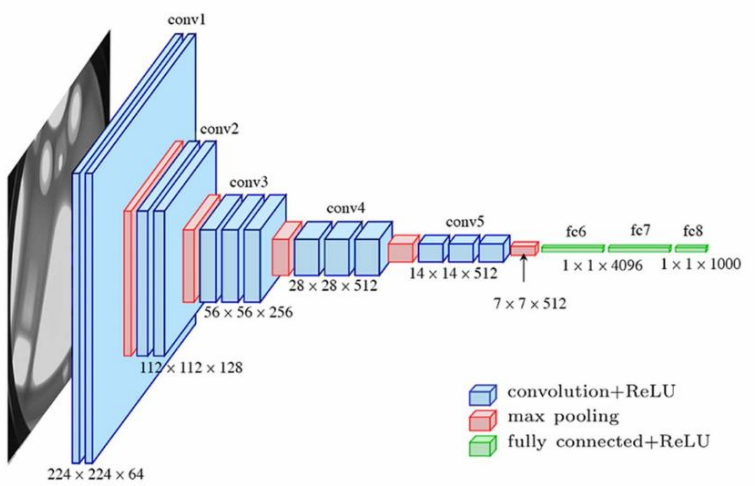

Fig. 3. VGG-16 network architecture (source: [7])

VGG-16 has the following advantages: it is a well-known image feature extractor; it has a simple architecture (similar to Alexnet)

3 EXPERIMENTAL SETUP

In this section, we describe the setup of our experiments. It is important to note that although over one hundred trials with different scenarios were conducted, only five representative trials will be discussed. Trials 1, 2, and 3 are based on the Alexnet framework with different datasets, while trials 4 and 5 are based on the Googlenet network. All trials are based on refined end-to-end learning of convolutional neural networks. The main variables are:

- Dataset (3 combinations of datasets are used)
- Network model (3 main networks are discussed)
- Use of image processing techniques

Although creating a network from scratch is simple, fine-tuning the parameters with a small dataset of no more than 1,500 images is extremely difficult. Therefore, we will draw inspiration from existing CNNs to create our own model: this is transfer learning.

For example, chest X-rays are sensitive to rotation and vertical flipping. However, in baggage screening, rotating and flipping the image does not change the direction it conveys. Another example would be increasing intensity. A change in intensity might not affect chest X-rays as much as it can affect baggage screening, due to the information embedded in the color format of the baggage.

3.1 TRANSFER OF LEARNING FROM A NATURAL IMAGE CLASSIFIER TO AN X-RAY CLASSIFIER

This scheme is proposed for classifying hidden plastic threats in X-ray luggage images. In the architectures mentioned in Section 2, each network uses a different concept and technique within its layers, which have their own effect on accuracy, speed, convergence rate, and memory size.

3.1.1 TESTING WITH RADIOGRAPHIC IMAGES OF LUGGAGE

Based on the advantages mentioned in Section 2, we decided to test their ability to classify radiographic images of luggage. After each training session, we will measure: 1) Model training accuracy; 2) Training loss; 3) Validation accuracy; 4) Test accuracy (to verify the validation result with a different set); 5) Confusion matrix to extract false positives and false negatives.

From the results, we will then select the network with the best performance and use it as a basis for designing our own network. We will not consider speed as an indicator of good performance since our main objective for this article is the network's ability to correctly classify 3D plastic threats, provided it is less than 1.5 seconds. However, if both networks provide similar accuracy, we will choose the network with the faster speed to test the images.

3.1.2 SELECTING HYPERPARAMETER OPTIONS

CNNs have many variables that can be adjusted to suit our scope. On the other hand, there are parameters that are established when the architecture is created from scratch, such as: 1) Zero Fill; 2) Strides; 3) Input Size; 4) Number of Layers; 5) Convolution Window Size; 6) Serial or Direct Acyclic Graph Network.

In this article, we will not modify the pre-established parameters because this would lead to very large and uncontrolled variables. However, we will focus on the variables that are included in the training options for the deep network framework. These variables include: 1) Network model type; 2) Number of epochs; 3) Batch size; 4) Optimization function; 5) Learning rate; 6) Increase.

We will try three network models as mentioned previously and experiment with the augmentation style, epoch time, batch size, and learning rate.

3.2 DATA PREPROCESSING

Analyzing and cleaning a dataset is one of the most important steps before applying a machine learning architecture. Raw data is generally not suitable for direct transmission to the CNN without a preprocessing step that includes cleaning. It is very common to have a dataset with numerous errors during the data collection phase due to:

1) X-ray machine errors caused by anomalies in the transmission and reception of photons
2) Human error (labeling error), such as mistakenly placing a normal negative class image in a positive threat class folder
3) TIP cases that fail and leave behind large visual landmarks
4) Unbalanced datasets (positive class being larger than negative class)
5) Highly complex images with occlusions of approximately 100%

Such errors are very common in many data collection scenarios, including baggage X-rays. Therefore, we must clean and analyze our datasets to ensure they contain only images suitable for training and classifying the network. Failure to do so would cause the network to diverge.

3.3 USE CASES

The cleaned dataset must be divided to account for different use cases. The four use cases discussed will provide insight into the dataset to choose for our intended model. We will only consider datasets obtained by combining staged threat images.

3.3.1 TRAINING AND VALIDATION CORPORA

In each of the four datasets, a full network will be run to study different behaviors, and we will attempt to interpret the representation and retrieval capabilities of our deep network.

DATASET 1: FOR TWO-CLASS CLASSIFICATION

This dataset contains both projections and staged images.

Class 1 = plastic threat, 371 images

Class 2 = normal luggage, 378 images

Total = 749 images (90% training, 10% validation dataset)

Note:

The Class 1 dataset will refer to standard light-background color images.

The Class 2 dataset will refer to images that have undergone a color transformation/complement of the original images.

DATASET 2: FOR TWO-CLASS CLASSIFICATION

This dataset contains both projections and staged images. We added additional images to this dataset that we considered "poorly projected images" to increase the total number of images in the dataset.

Class 1 = Plastic Threat, 406 images

Class 2 = Normal Luggage, 406 images

Total = 812 images (80% training, 10% validation, and 10% testing)

DATASET 3: FOR TWO-CLASS CLASSIFICATION

In this dataset, we want to determine if the network can generalize and learn to classify images extracted from machines with different color standards, such as black and white and dark-intensity background images.

Class 1 = Plastic Threat, 562 images with various machine color formats

Class 2 = Regular Luggage, 562 images with various machine color formats

Total = 1,124 images (80% training, 10% validation, and 10% testing)

DATASET 4: FOR THE MULTI-CLASSIFICATION TASK. THIS DATASET CONTAINS THE FOLLOWING CLASSES

Class 1 = Plastic Threat, 408 images

Class 2 = Regular Luggage, 406 images

Class 3 = Metallic Weapon, 350 images

Class 4 = Other (random mini-crops of a BW X-ray), 427 images

Total = 1,591 images – 80% training, 10% validation, 10% testing.

These four different datasets will be tested using graphs and confusion matrices to reflect the best results and reach a meaningful conclusion.

3.3.2 CORPUS AUGMENTATION

A medium-sized dataset (over 500 images) is insufficient to train the network to perform well during testing. Therefore, we will apply various forms of random augmentation to address this issue and ensure good generalization to a wide range of valid real-world scenarios. The variables are listed below with their expected values:

1) Random Rotations: from [-60 60]
2) Random Horizontal Translations: from [-3 3]
3) Random Vertical Translations: from [-3 3]
4) Random Horizontal Reflections
5) Random Vertical Reflections

4 RESULTS AND DISCUSSION

In this section, we train and test our pre-trained networks mentioned in section (2) on each of the different types of datasets mentioned in section (3.3.1). These datasets are therefore the main ingredient of our experiment.

We note that the choice of a high-quality, complete, and diverse dataset affects the final accuracy of the system. We also evaluate the networks with different tuning points to determine the critical "sweet spot" of our system solution. Each experimental parameter will be called a "trial," and we will train and test the models with different datasets accordingly.

4.1 SIMULATION RESULTS

Datasets 1, 2, 3, and 4 are tested:

4.1.1 ALEXNET PRE-TRAINED MODEL

In Trial 1, we will train the Alexnet model and then test it accordingly. We simulate the model with dataset 1 using the following parameters:

- Data Augmentation: Yes
- Training Rate: 0.001
- Batch Size: Standard 128
- Loss Function: SGDM
- Epoch: 60
- Data Partitioning: 90% Training, 10% Testing

Figure 4 shows the training performance of this model. Table 1 shows the results on the validation set in terms of the confusion matrix.

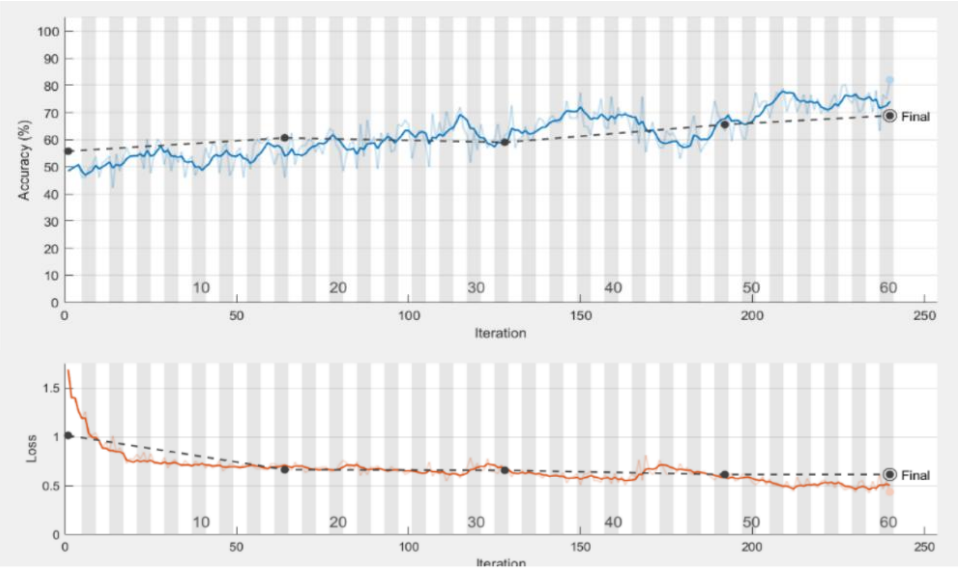

Fig. 4. Training graph of the Alexnet model based on dataset 1, trial 1

Table 1. Confusion matrix for trial 1, validation accuracy: 68.85%

		Predicted Class	
		Normal	**Threat**
True Class	Normal	28	2
	Threat	17	14

In trial 2, we increased the dataset size with 75 additional images in both classes and retrained the network. The results were slightly improved, as shown in Figure 5 and Table 2:

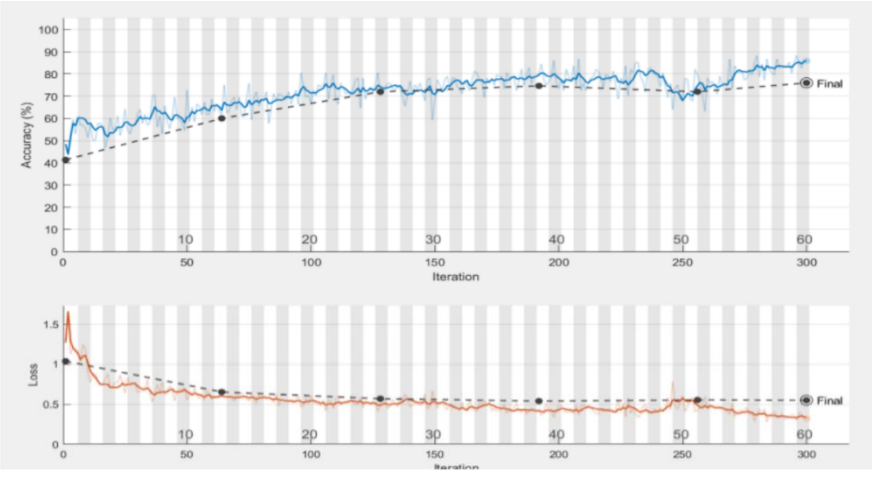

Fig. 5. Training plot of the Alexnet model based on dataset #2, trial 2

Table 2. Confusion matrix for trial 2, validation accuracy: 76%

		Predicted Class	
		Normal	Threat
True Class	Normal	21	16
	Threat	2	36

Accuracy was increased by increasing the dataset size, indicating that with a more diverse dataset, the network generalized better. Current results show that the network can learn, but it does not learn quickly. It requires a more complex learning architecture.

Before testing other, deeper networks, further research will be conducted on Trial 3 to see the effect of improving the visibility of plastics in images by using the image complement as an image preprocessing step.

In Trial 3, the results are shown in Figure 5.3 and Table 5.3:

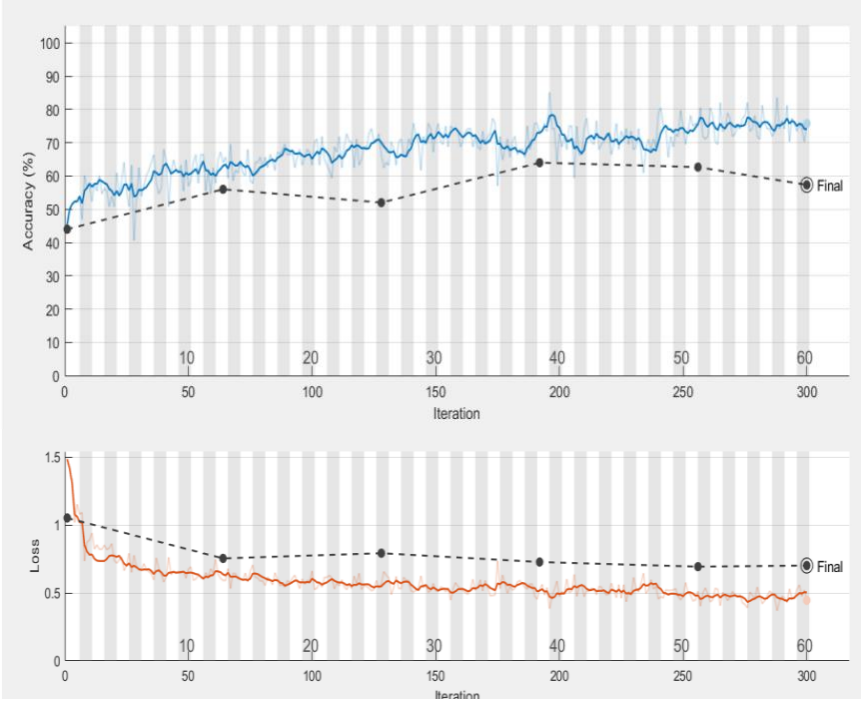

Fig. 6. Training graph of the Alexnet model based on dataset 2, trial 3

Table 3. Confusion matrix for trial 3, validation accuracy: 57.33%

		Predicted Class	
		Normal	Threat
True Class	Normal (color transformed)	26	11
	Threat (color transformed)	21	17

This method is far worse than having the dataset with its normal color space and no color transformations.

4.1.2 PRE-TRAINED VGG MODEL

VGG is a deeper network, suggesting it can learn more complex data. Therefore, we decided to train it with the same parameters as in the previous example. The system could only be successfully trained when the batch size was reduced to a very small number due to a GPU memory issue. The small batch size resulted in very low accuracy, and consequently, training failed. When the batch size was increased, an error pop-up appeared stating "GPU low on memory" due to the layer size and the way the network learned. Therefore, this model failed due to the GPU's inability to meet the memory requirements.

4.1.3 PRE-TRAINED GOOGLENET MODEL

GoogleNet has the most complex architecture among the previous models. Therefore, it is logical to assume that such a network would be capable of classifying more complex images. We present three simulations using this model, with training performed on datasets 2, 3, and 4.

Trial 4 uses dataset 2 with the following parameters:

1) Data augmentation: Yes
2) Training rate: 0.001
3) Batch size: Standard 128
4) Loss function: SGDM
5) Epoch: 90
6) Data partitioning: 80% training, 10% validation, 10% testing

The results are shown in Figure 7 and Table 4:

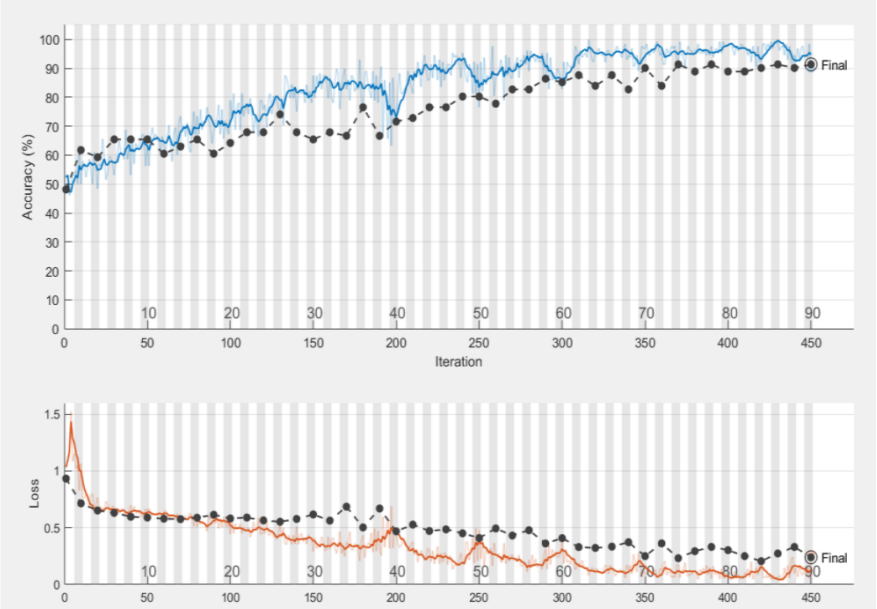

Fig. 7. Training graph of the Googlenet model based on dataset #2, trial 4

Table 4. Confusion matrix for trial 4, validation accuracy: 91.36%

		Predicted Class	
		Normal	**Threat**
True Class	Normal	38	2
	Threat	5	36

It is immediately apparent that the results far exceeded those of the Alexnet model. This model is therefore the most accurate result of our research for a two-class classification problem.

In trial 5, we used dataset 3, with an increased increment in this trial as follows: Random rotation: [-90 90] (to ensure sufficient variance is applied to the dataset).

The results are presented below in Figure 8, Tables 5 and 6:

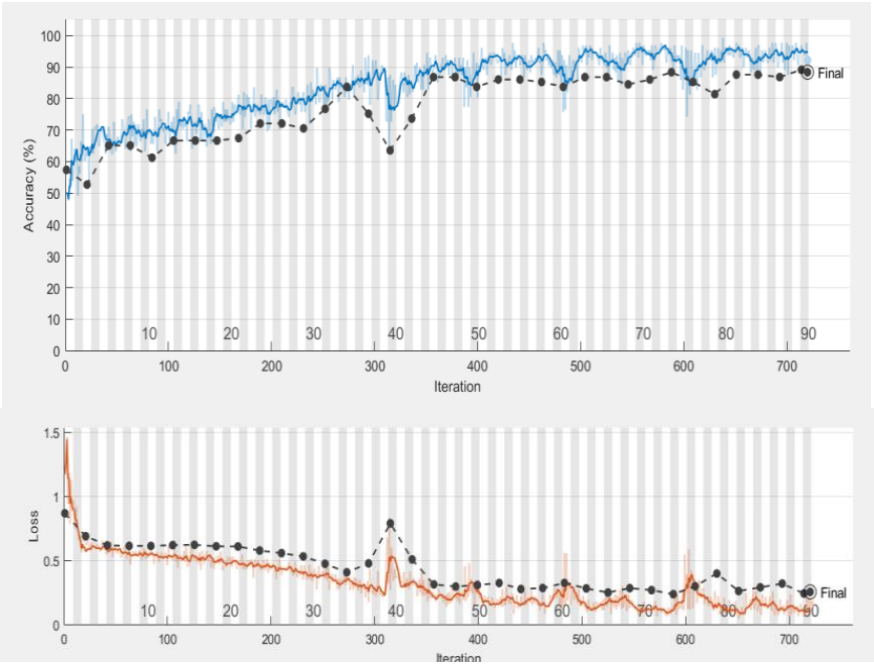

Fig. 8. Training graph of the Googlenet model based on dataset #3, trial 5

Table 5. Confusion matrix for trial 5, validation accuracy: 88.3%

		Predicted Class	
		Normal	**Threat**
True Class	Normal	56	9
	Threat	6	58

Table 6. Confusion matrix for trial 5, test precision: 89%

		Predicted Class	
		Normal	**Threat**
True Class	Normal	59	5
	Threat	9	56

These results indicate that models such as GoogleNet can be generic with images of different colors within the same dataset. Although the result is not as good as in the case of the application of dataset #2, it nevertheless shows that accuracies can fall within an acceptable range when sufficient information is included in the dataset.

Finally, Trial 6 investigates a 4-class classification problem. In this problem, we want to explore whether the network can classify both plastic and metal weapons. This was done by integrating our dataset with the publicly available GDXray dataset [2]. In this example, dataset #4 is used with the following parameters:

1) Data augmentation: Yes
2) Learning rate: 0.001
3) Batch size: Standard 128
4) Loss function: SGDM
5) Epoch: 90
6) Data partitioning: 80% training, 10% validation, 10% testing

The details of the four classes are:

- Gun: Metal gun
- Normal: Normal baggage class, SoC images
- Other: Normal baggage, according to the GDXray dataset
- Threat: Plastic gun threat image

The results are shown in Figures 9

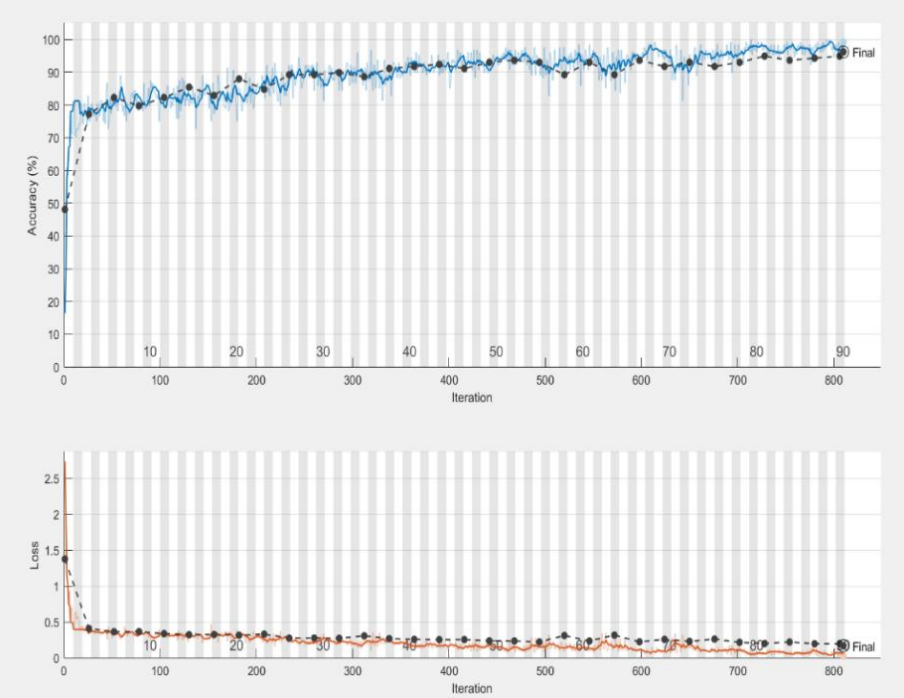

Fig. 9. Training graph of the GoogleNet model based on dataset #4, trial 6

This result is one of the most inspiring findings of this article. Not only did the network classify plastic guns as "difficult to detect" with high accuracy, but it also ranked the metal guns available to the academic community as one of the few datasets available worldwide.

4.2 PERFORMANCE EVALUATION

We summarize the results in Table 7 with a legend:

- A: Trial Number
- B: Dataset Used
- C: Number of Classes

Table 7. Comparison of the five trials discussed with metrics for each method

A	B	C	Variables	Accuracy	Precision	Recall
1	1	2	The dataset is simple, based on the Alexnet model.	68.85%	0.4516	0.875
2	2	2	The dataset is augmented with more images, still trained on the Alexnet model.	76%	0.9474	0.6923
3	2	2	We took the complement of the image dataset to convert the colors to a more plastic-sensitive intensity.	57.33%	0.4474	0.6071
4	2	2	Transition to a deeper network, Googlenet. Increase in the number of epochs to 120.	91.36%	0.8780	0.9474
5	3	2	Adding background images of different colors, such as black and white and dark backgrounds directly extracted from the machine.	88.3%	0.9	0.8657

Trials 4 and 5 clearly yielded the best results. Epoch numbers also increased in these two trials. However, this is not a critical factor since the training graph shows that the network produced very accurate results from epoch 60 onward. Surprising results can be observed in trial 5, in which we used a mixture of color images with unbalanced quantities, yet the network was still able to correctly classify many test images. Another important result is that it is not necessary to take the complement of the images before feeding them into the network.

4.3 GENERALIZATION

AI systems and CNNs are an excellent option for threat detection, especially when validated with a different set of images not used for training. However, using the same threat itself (a 3D-printed plastic gun), as in our case, we cannot distinguish whether our system memorizes all the features of this object or if it actually learns the main features and is capable of generalization. To prove that the system is a generalizable AI system, it must be tested with objects that are:

- To some extent from a different domain (not exactly from the same weapon family)
- Made from a different material

To achieve this, it was decided to use toys. The toy guns resemble the 3D-printed guns using X-ray imaging.

The toy guns were deliberately chosen so that they did not resemble the gun with which the system was trained. These two weapons are:

4.3.1 BLACK TOY PISTOL (ILLUSTRATED IN FIGURE 10)

Fig. 10. Image of a black toy gun with two handles

This gun looks different from the gun on which the system is trained. This gun has two handles and is not 3D printed. Therefore, it is reasonable to assume that the system will likely not be able to classify it correctly. It is worth noting that the following results are from testing (not training) 100 images containing both normal and threatening images (black toy gun) in Table 8:

Table 8. Confusion Matrix for proving generalization, test accuracy: 77%

		Predicted Class	
		Threat	Normal
True Class	Threat	30	20
	Normal	3	47

Achieving a 77% accuracy rate on a previously unknown threat is very good. First, this weapon was not used for system training. Second, the weapon differs from the one used for training in terms of appearance and construction. Third, this matrix showed very few false negatives, which is excellent for real-world applications. Overall, considering the points mentioned above, the system has proven that it can generalize its parameters to classify an object on which it had not been trained. Figure 11 shows how blurry the weapon's appearance is in the X-ray image.

Fig. 11. X-ray images of a black toy gun

4.3.2 GREEN TOY GUN

To study the system's generalizability in more detail, we decided to explore an extreme case: a science-fiction toy gun, illustrated in Figure 12.

Fig. 12. Image of a different front view of a green toy gun

This gun is completely different from the one on which the system was trained. The functionality of these two guns is also different. This green gun is more in line with science fiction guns, which are not very realistic. Therefore, it is quite normal to expect a poor result, as shown below in Table 9. Figure 15 shows what this gun looks like in the X-ray.

Table 9. Confusion Matrix for proving generalization, accuracy test: 60.3%

		Predicted Class	
		Threat	**Normal**
True Class	Threat	20	41
	Normal	3	47

Although the 60% accuracy is rather low, it is nonetheless significant in terms of the value it represents. The figures in Table 9 indicate that the system is learning (capable of learning), but it incorrectly classifies most threats as normal. Evidence that the system is capable of partially detecting a threat is found in the results of normal images. Note that the system correctly classifies normal images with a very small error. If the system were random, it would be impossible to behave as shown in Table 9. However, because the tested object is far from the object on which we trained the system, it does not classify the threat with very high accuracy.

Fig. 13. X-ray images of a green toy gun

With these results, our system's ability to generalize to new threats has been proven. It is believed that by incorporating these two toy guns into our original training dataset, the system will improve further and be able to detect much more unusual firearms.

The key lies in variability and knowledge extraction from a large set of random materials. The accuracy results of this experiment can be increased by increasing the number of images in the training dataset. However, a fixed number of datasets must always be used due to time and cost constraints.

Furthermore, it is believed that by using more advanced networks than the GoogleNet, such as InceptionV3 or newer ones, the level of accuracy in testing complex images can be increased. A low-end GPU (Titan 1070A) was used, which made using the InceptionV3 model impossible due to memory limitations. Finally, it's also important to mention the computational information for the two networks used, to understand how network depth and speed can affect convergence:

- GoogLeNet: This network is deep in size (22 layers) but small in memory (22 MB). It is suitable for standard PCs with a mid-range GPU.
- AlexNet: This network contains 9 layers and over 70 million parameters, and is enormous in memory (200 MB). We also trained it on the Titan 1070A GPU.

5 CONCLUSION AND FUTURE WORK

In this thesis, various convolutional neural networks (CNNs), datasets, and techniques were used to find the optimal combination, resulting in a system capable of classifying suspicious luggage (containing plastic firearms) with reasonable accuracy.

CNNs can recognize objects almost to the point of disappearance in an image. However, beyond this point, they generally struggle to perform a correct classification.

CNNs can always be improved to achieve better results. We believe that the contributions made in this paper can serve as a basis for building our own CNN for classifying X-ray images.

Therefore, for future work, we propose acquiring more than one view of the examined luggage and then applying both views in the learning algorithm for improved results. This can be achieved through the use of more complex X-ray devices such as CT scanners and dual-view X-ray machines.

REFERENCES

[1] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich «Going deeper with convolutions,» IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, 2015, pp. 1-9.

[2] A. Krizhevsky, I. Sutskever, and G. Hinton. «ImageNet Classification with Deep Convolutional Neural Networks». Neural Information Processing Systems. vol. 25, Jan 2012.

[3] K. Simonyan and A. Zisserman, «Very Deep Convolutional Networks for Large-Scale Image Recognition». ArXiv 1409.1556, Sep. 2014.

[4] 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, pp. 682-692, Apr. 2017.

[5] K.J. Liang, G. Heilmann, C. Gregory, S.O. Diallo, D. Carlson, G.P. Spell, J.B. Sigman, K. Roe, L. Carin, «Automatic threat recognition of prohibited items at aviation checkpoint with x-ray imaging: a deep learning approach» Proc. SPIE 10632, 2018.

[6] 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, Nov. 2015.

[7] Gaudenz Boesch. GoogLeNet Explained: The Inception Model that Won ImageNet, May 7, 2024. https://viso.ai/deep-learning/googlenet-explained-the-inception-model-that-won-imagenet/.