An Optimized Mask R-CNN with Bag-of-Visual Words and Fast+Surf Algorithm in Sharp Object Instance Segmentation for X-ray Security

Edgardo Jr S. Abong [1], Karelle Teyle A. Janducayan [1],
Jomer Mae M. Lima [1], Meljohn V. Aborde [1]

[1] *University of Mindanao, Matina, Davao City, Philippines*

Abstract **– Automated security X-ray analysis is highly desired for efficiently inspecting sharp objects. The research formulated an optimized approach for sharp object detection using a Mask R-CNN architecture. The dataset used during the training phase consists of 238 balanced raw images extracted from GitHub named OPIXray. The researchers utilized recent advances in computer vision algorithms, including the Bag-of-Words and Fast+Surf feature extraction techniques, to improve the accuracy and reliability of object deletion. The research demonstrated that the optimized versions of the classification and object detection models have significantly improved accuracy for most categories, with a 5% improvement for the clear category and a 3% improvement for both the scissor and straight knife detection.**

Keywords **– Mask R-CNN, Bag-of-Visual-Words, Fast-Surf, x-ray scanning, detection.**

DOI: 10.18421/TEM132-09
https://doi.org/10.18421/TEM132-09

Corresponding author: Meljohn V. Aborde,
College of Computing Education, Faculty of Computer Science, University of Mindanao, Davao City Philippines
Email: mjaborde@umindanao.edu.ph

Received: 01 October 2023.
Revised: 10 January 2024.
Accepted: 19 January 2024.
Published: 28 May 2024.

1. Introduction

Object detection through X-ray image processing has been more helpful and gained more popularity in both past and present times. As a result, object detection is used for various purposes, from medical applications to stopping terrorist attacks. However, the most well-known use is for security purposes, typically seen in airport security, where thousands of bags are scanned daily on a conveyor belt. With the emergence of this kind of technology, checking bags has always been challenging while reducing time constraints.

The Explosive Detection System for Cabin Baggage Screening (EDSCB) performs far better detecting explosives than manual screening by highly tenured experts [1]. EDSCB uses an automated decision-making system and is still being used. There are many ways to approach such threat detection in X-ray images; however, it has been proven far better than manual inspection. The study by [2] highlighted how manual inspection takes up much time while providing slow service. This process might cause delays and even poor performance.

X-ray imaging techniques for detecting threats in baggage can revolutionize how baggage checking is done [3]. It allows inspection without any contact with the baggage owner and dramatically increases the speed while accommodating more people quickly. The Dual Convolutional Neural Network (Dual CNN) architecture achieved the modern standard on object detection for airport security in terms of its state-of-the-art performance and effectiveness rate. It consists of two stages in detecting possible threats from X-ray footage. Stage 1 identifies every object found within the baggage from the X-ray image using Mask R-CNN.

Furthermore, stage 2 classifies the object, whether it is a threat or not, using seminal CNN object classification architectures (SqueezeNet, VGG-16, ResNet).

The data shows that the first stage has 76.86% true positive (TP) with 66% accuracy in classifying whether they are a threat or not in the second stage. Although with superb statistical results, the method has a 10% false positive (FP) chance, which highly affects its integrity [3]. In hindsight, Mask R-CNN has a simple concept: when an object is detected, that object is then outlined with a mask. When such an object is masked, it shows finer details of that object. Additionally, going deeper into the Mask R-CNN, it has two stages. The first stage is the Regional Proposal Network (RPN), which proposes a candidate object bounding box, and the second stage is parallel to predicting both box offset and class [18].

A recent study showed that the classical Bag-of-Visual-Words (BoVW) model, when used to split an object into different regions before analyzing whether it is a threat or benign, can significantly affect the speed and reliability of this technology [4]. This method can be inserted after the Object Detection Stage, as shown in Figure 1, to make the Classification Stage more accurate and effortless. The same study also demonstrated that the integration of Features from Accelerated Segment Test - Speeded Up Robust Features (FAST-SURF) improves the effectiveness of object splitting by a significant margin, resulting in a 94% accuracy, 83% true positive and a minimal 3.3% false-positive chance, compared to the original Dual CNN setup. While the Bag-of-Visual-Words (BoVW) model exhibits significant power when integrated with the FAST-SURF algorithm, its standalone implementation without additional optimization proves inefficient. As a classical model, it may need more advanced features in newer models, thus hindering its performance and limiting its applicability in contemporary scenarios.

Another study [5] provided evidence of how the SURF algorithm enhances the effectiveness of this stage by allowing improved object classification even in situations where the object appears different due to visual changes. The SURF algorithm can achieve this by using color invariant transformations, information entropy theory, and a set of constraint conditions to improve feature point identification and matching. In line with the study, a study from 2017 suggested using an automated object segmentation and clustering architecture to detect high-risk threat objects in the UK [6]. The study used dual-view single/dual-energy 2D X-ray imagery and a triple-layered processing scheme based on the atomic number of the contents of the luggage. It combined radiology, image processing, and computer vision concepts [6]. The study supports the research as it aims to improve the current method of detecting high-risk objects and enhancing the results [6].

Another study [7] compared several algorithms, such as Convolutional Neural Networks (CNN), Stack Autoencoders, Shallow Neural Networks, and Random Forest, to see which one could better identify steel barrel bores as threat objects on a 22k double view x-ray scan dataset. The evaluation performance was measured using the Receiver Operating Characteristic (ROC) curve (AUC), FPR@90%TPR (False Positive Rate of 90% True Positive Rate), and F1-score. The study's results [7] showed that CNN outperformed all the other algorithms on the three performance evaluations and the dataset. Therefore, the research will simulate a similar comparison, assessing the old and improved methods of detecting threat objects.

Another relevant study introduced the OPIXray dataset, which includes fully annotated bounding box samples and annotation boxes, serving as a benchmark for X-ray image detection tasks. However, it should be noted that the OPIXray dataset lacks annotations for masking and requires additional preprocessing and annotation efforts in that aspect. In addition, despite heavy occlusion in X-ray imaging, objects retain their shape and appearance, and different materials exhibit distinct colors and textures. The De-occlusion Attention Module (DOAM) leverages these observations to exploit the diverse appearance information of prohibited items and generate attention maps, refining feature maps for general detectors [8].

The Dual CNN architecture has shown the exceptional performance of all the available algorithms. One of its significant strengths is its ability to provide a convenient and automated system for detecting harmful objects without requiring direct human interaction. Additionally, the Dual CNN architecture exhibits remarkable speed, allowing it to swiftly detect and identify multiple objects within a fraction of a second. However, it is essential to acknowledge specific weaknesses as well. One notable drawback is the relatively low accuracy in classifying objects, achieving only a 66% accuracy rate. Moreover, the architecture demonstrates a concerning 10% false positive rate, which poses a challenge in real-world applications where minimizing false alarms is crucial. Addressing these weaknesses will be a primary focus of the research, aiming to enhance the accuracy and reduce false positives in the Dual CNN architecture.

The main objective of this research is to Optimize Region-based Convolutional Neural Networks (R-CNN) architecture object detection method by implementing both Bag-of-Visual-Words (BoVW) and FAST-SURF for object matching algorithm to reduce False-Positive results and attain higher accuracy in sharp object detection on security X-ray images.

2. Methodology

The current algorithm or way used for scanning revolves around two stages; within these two stages, the first stage has an effectiveness rate of 97.9% mean average precision on object detection [3]; the first stage uses Mask R-CNN for detecting objects. The second stage boasts a true positive of around 76.86%, with 66% accuracy in classifying if such an object is a threat or benign [3]. The current way of scanning objects and the algorithms used are considered superb, but it can still be improved by adding another stage to maximize the results. This study focuses on detecting sharp objects using the Mask R-CNN Algorithm with the addition of the two algorithms, the BoVW and the FAST+SURF algorithm.

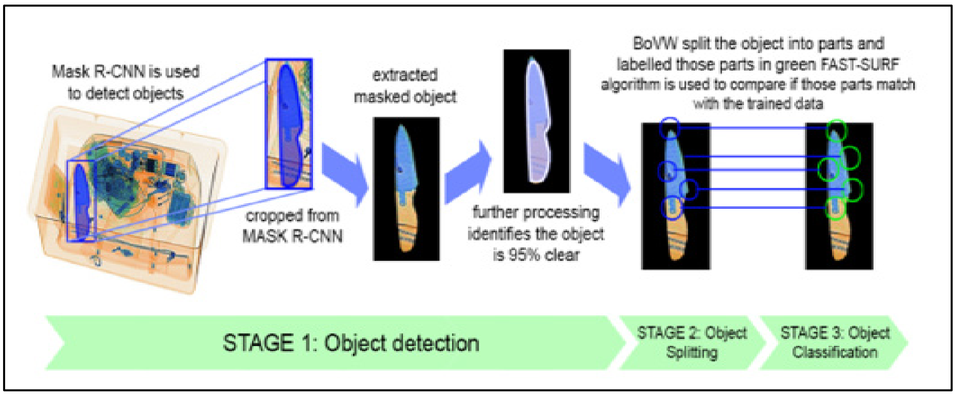
Mask R-CNN is used to detect objects
cropped from MASK R-CNN
extracted masked object
further processing identifies the object is 95% clear
BoVW split the object into parts and labelled those parts in green FAST-SURF algorithm is used to compare if those parts match with the trained data
STAGE 1: Object detection
STAGE 2: Object Splitting
STAGE 3: Object Classification

Figure 1. Optimized Mask R-CNN process diagram

First, given a scenario: A passenger passed through the security entrance and dropped his baggage on the X-ray machine for scanning. The X-ray machine will scan the luggage and send an X-ray image to the system for analysis. The X-ray image will undergo three stages to ensure the reliability of the result. The three stages are the object detection stage, object splitting stage, and object classification stage. Each of the stages has its functions. This study's main point is to recognize each object inside the luggage to see if it is a threat or benign. Therefore, the first stage will be the object detection stage, which will recognize the objects found on the luggage. The second stage is the object splitting stage; after the objects are identified, it is time to acknowledge the parts of each object, known as splitting the object information. Moreover, the third stage, the object classification stage, is where the objects will be recognized as benign or a threat.

2.1. Conceptual Framework

The following sections discuss the concept of the study. It analyses the different activities happening on the object detection stage, object splitting stage, and the object classification stage.

2.1.1. Object Detection Stage

The object detection stage is the first stage in which the raw X-ray image will undergo. In this stage, the Mask R-CNN algorithm is used to identify objects from the input image. Next, it processes the image through Region of Interest (ROI) Pooling - a masking process that carefully analyzes the edges of each object to be mapped [3]. Moreover, the image is analyzed by scanning for high pixel differences to map the boundary of a particular object. After careful analysis, it will return an array of detected objects used in the second stage, region splitting.

2.1.2. Object Splitting Stage

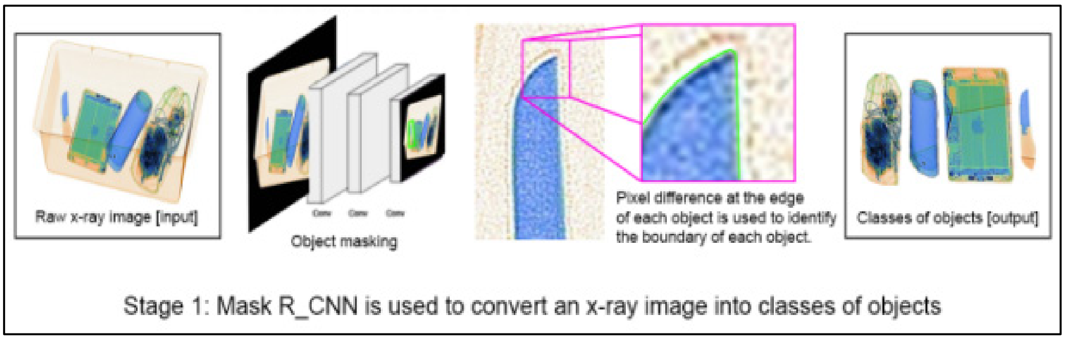
Raw x-ray image [input]
Conv Conv Conv
Object masking
Pixel difference at the edge of each object is used to identify the boundary of each object.
Classes of objects [output]
Stage 1: Mask R_CNN is used to convert an x-ray image into classes of objects

Figure 2. Object detection

In region splitting shown in Figure 2, each object detected in the previous stage will undergo specific part identification using the BoVW model. One object will be divided into a particular number of identifiable parts depending on how big the object is or how many features are identifiable [4]. An array of divided regions will then be forwarded to the last stage.

2.1.3. Object Classification Stage

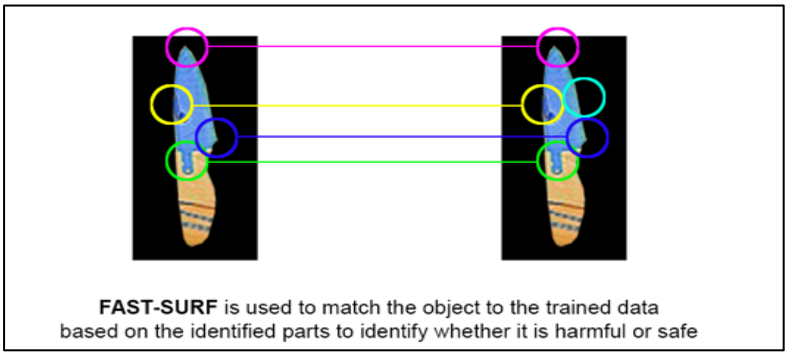
FAST-SURF is used to match the object to the trained data
based on the identified parts to identify whether it is harmful or safe

Figure 3. Object classification

Lastly is the object classification stage shown in Figure 3. In this part, the FAST+SURF object matching algorithm will be used to classify whether a particular object is suspicious or benign by comparing each region to the acceptable models from the dataset. The FAST algorithm will set feature points based on pixel comparison by plotting a point on a pixel and analyzing its surrounding pixels. The SURF algorithm will then process the resulting data to add descriptors [9]. After completing the three stages, the data is analyzed more accurately, and true positive results are given.

2.2. Mask R-CNN

Before forming the said Mask R-CNN, CNN consisted of several convolutional and pooling layers, which end in one or more fully affiliated layers. Therefore, each convolutional layer consists of convolutional, non-linear activation, and pooling, three general steps. First, a feature map is generated after each layer of the convolutional process; this is then passed down to the next layer.

Developed for object localization, object instance segmentation, and semantic segmentation, the model called Mask Regional Convolutional Neural Network (Mask R-CNN). In-depth, the model Mask R-CNN consists of two stages. The first stage scans for initial feature maps and creates regions of interest (RoI). In the second stage, there will be a process known as RoI-pooling; that process uses the nearest neighbor approach, which is applied to each RoI to down-sample the feature map [15].

The term Dual CNN was derived after two different CNN approaches were used in two stages to process an X-ray image. The first stage is the object identification stage. This stage uses the Mask R-CNN algorithm to detect objects from the X-ray image. After objects are detected, they will go to the object classification stage. In this stage, the proposed study will use various CNN object classification architectures to analyze and identify whether these identified objects are safe or harmful.

This architecture focuses only on the image and shape of an object, paying no attention to little details that might be a factor in identifying whether this object is safe or not. Considering this, the x-ray data provided in study [3] reflected how this architecture risks up to 10% false positive (FP) result.

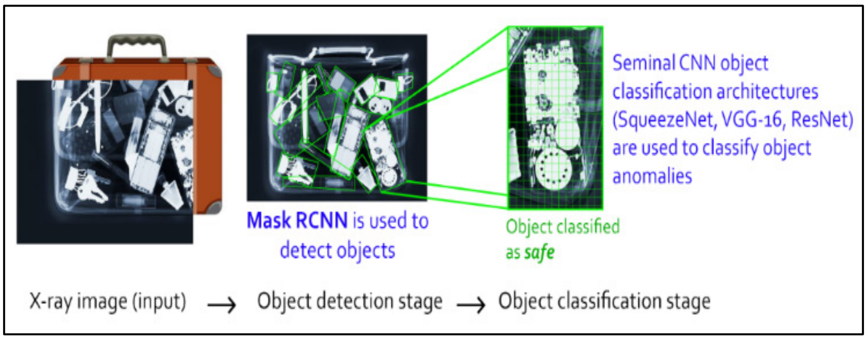
Seminal CNN object classification architectures (SqueezeNet, VGG-16, ResNet) are used to classify object anomalies
Mask RCNN is used to detect objects
Object classified as *safe*
X-ray image (input) ⟶ Object detection stage ⟶ Object classification stage

Figure 4. Old dual CNN process diagram

A more concrete description of this data shows that 1 out of 10 objects classified for anomaly detection shows a false positive result. This interpretation means that either the algorithm marked a safe object as harmful or a dangerous object as safe. The latter is a considerable risk that might cause harmful objects to be tagged safely and pass the establishment's safety protocol.

2.3. Bags-of-Visual-Words (BoVW)

The Bag of Visual Words (BoVW) is well-known and frequently used in image classification. Its concept is modified from information retrieval and NLP's Bag of Words (BoVW) [11].

The Bag-of-Visual-Words (BoVW) model can be adequate for image classification, object recognition, and image retrieval. Within the BoVW model, it extracts features from any set of images into visual words within the visual dictionary [10]. The origin of BoVW is for text information retrieval and text classification. However, regarding image classification, BoVW interprets an image as a collection of local features [4]; those features are descriptor vectors. The BoVW can also be used in baggage security screening, which can boost the output if an object within that baggage is benign or a threat.

During the X-ray Scanning, there will be an image of what the things are composed inside the luggage or handbags. The result of the captured image lets the algorithm work on how the image will be split into parts. The Bag of Visual Words in this study is to identify the features or parts of the image or an object detected; after that, it is to divide the image that has been classified into pieces that are identifiable into classes.

2.4. Features from Accelerated Segment Test + Speeded Up Robust Features (FAST+SURF) Algorithm

Improved FAST feature point combined with SURF descriptor matching algorithm is proposed, which realizes the real-time matching of the target. The experimental results are compared with SIFT, SURF, and FAST+SURF algorithms [9].

Rosten and Drummond [12] proposed the features from accelerated segment test (FAST) algorithm to identify the interest points in a particular object or image. The Interest point of an image, called the Pixel, has a well-defined position sturdy enough to be detected. Moreover, the interest points have extensive local information content that is ideally repeatable between different image results. Also, interest point detection has many applications: image matching, object recognition, tracking, and more.

The SURF (speed up robust features) algorithm was based on the Hessian matrix, feature detector, and multi-scale space theory; additionally, the Hessian matrix has been considered to have good accuracy and performance [13]. Scale invariant feature transform (SIFT) was the first before the SURF; the SURF was the development of SIFT, but before SURF was put into the light, the SIFT algorithm was considered weak between robustness and computational time [14].

Features from accelerated segment test + speeded up robust features (FAST-SURF): The effectiveness of this method is a giant leap from the result of the original Dual CNN setup with 94% accuracy, 83% true positive, and the minimal 3.3% false-positive result. Using this algorithm in this study would compare the identified parts results during the object splitting stage using the BoVW algorithm. After that, it is to use the FAST + SURF since this algorithm is for object matching algorithm, which helps to identify the object whether it is benign or suspicious. Therefore, using these two algorithms might give a high chance of increasing the true positive result and decreasing the false positive result on the original DUAL CNN result.

2.5. Optimized Mask R-CNN with the Integration of Bags-of-Visual-Words and FAST-SURF Algorithm

The researchers have meticulously studied the possible ways to make the Mask R-CNN approach more accurate with a lesser false-positive chance while improving the true positive statistics.

The selected Bags-of-Visual-Words (BoVW) model will be integrated before the classification stage to simplify object classification. The bags-of-visual-words model will split the objects into parts or features and label them accordingly so that the classification stage will classify every identifiable feature individually by comparing it to the trained data instead of classifying it as an entire object.

Another model called the features from accelerated segment test - speeded up robust features (FAST-SURF) algorithm will also be incorporated to significantly increase the speed and accuracy of the object classification stage, especially since it will classify objects by splitting features.

The bag-of-visual-words (BoVW) and features from accelerated segment test - Speeded up robust features (FAST-SURF) are well-known models for object detection. This study aims to integrate the models into the MASK R-CNN and overcome the flaws in the object splitting and classification stage. Furthermore, aside from overcoming the deficiencies of the mask R-CNN, it also seeks to improve the false positive (FP) chance.

The experiment results will be obtained by comparing the optimized mask R-CNN algorithm and the baseline algorithm, which is the dual CNN algorithm, to see how integrating the BoVW and FAST-SURF algorithms improves the former.

To get the results, the experiment will run both the baseline algorithm and the mask R-CNN with bag-of-visual-words (BoVW) and FAST-SURF; the aims are to further reduce the false positive (FP) rate and to overcome the object splitting and classification flaws. To proceed with the experiment, the researchers intend to use Python programming language as the primary language for implementing these machine-learning models. Python's different libraries will be utilized to simulate the old and improved algorithms carefully. Furthermore, the simulation will run a hefty amount of publicly available images to see the differences in the result, more specifically, to identify whether there is an improvement.

2.6. OPIXray

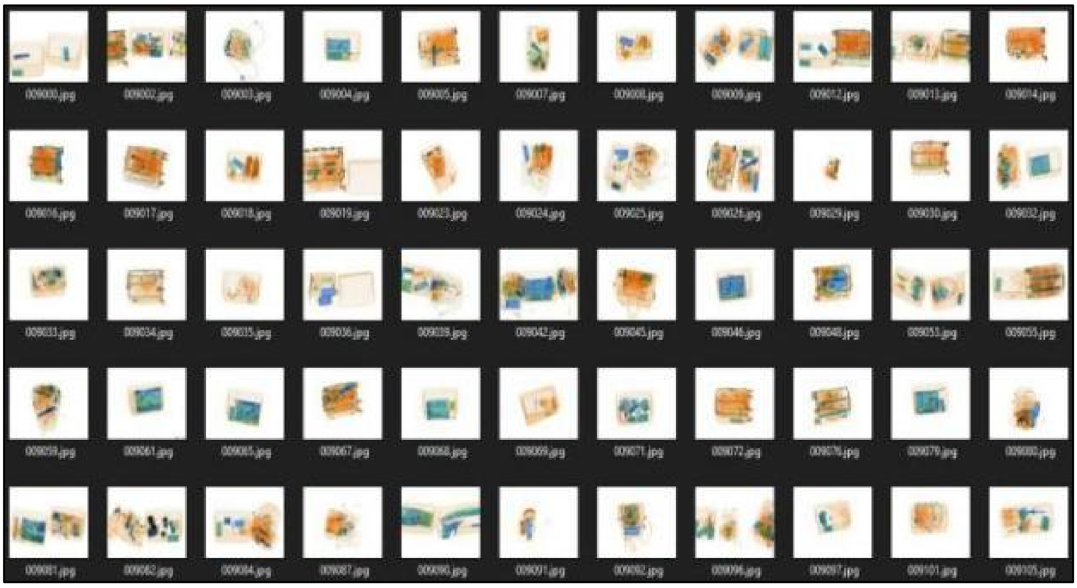

Figure 5. OPIXray dataset

The OPIXray was officially requested and gathered from the State Key Laboratory of Software Development Environment (SKLDE) at Bei Hang University. The OPIXray will be the dataset used within the simulation, using 238 images for the training and 102 for testing. Straight knives and scissors are within the dataset in many forms and angles, as seen in Figure 5.

2.7. Data Cleaning and Preprocessing

The researchers employed a combination of annotation tools and manual data selection to achieve the highest quality and efficiency of the dataset. The dataset underwent a thorough cleaning process to remove irrelevant or corrupted samples, ensuring that only relevant and high-quality data remained. Furthermore, an optimal number of images was carefully chosen for each class to achieve a balanced representation, specifically for the straight knife and scissor categories. These selected images were then divided into training and testing sets following a 70/30 ratio, ensuring a suitable distribution for evaluating the models' performance.

The resulting dataset exhibits a refined composition that can effectively support the subsequent model training and evaluation processes by employing these meticulous data cleaning and preprocessing techniques.

2.7.1. LabelMe

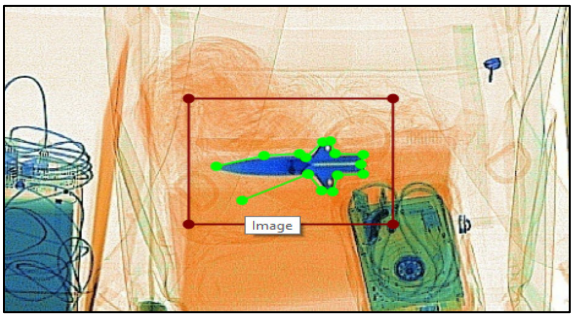

Figure 6. LabelMe data annotation

LabelMe is a polygonal image annotation software using Python [16]. Since the simulation focuses on Mask R-CNN, the original dataset OPIXray needed an annotation.

On the OPIXray dataset, the only annotation it has is a bounding box which can be seen in Figure 6 as an example. The bounding box is correlated to FAST R-CNN; because of this, using LabelMe to add the masking or outline annotation to the dataset manually is a must for the Mask R-CNN to work correctly.

2.8. ResNet

ResNet will be the learning model in the mask R-CNN; ResNet is considered the best and most popular regarding image recognition. The ResNet for image recognition came from a paper titled *Deep Residual Learning for Image Recognition.* [19]. Additionally, ResNet comes with pre-trained models and existing configures that do not need to be done from scratch. With the introduction of ResNet, the problem of training deep networks has been lessened [17].

The figure below is called a 'Skip Connection,' a direct connection that skips some layers within the model, as seen in Figure 7.

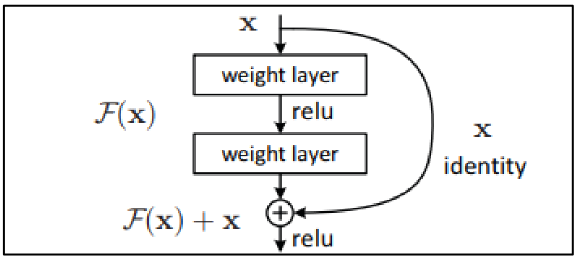
x
weight layer
$\mathcal{F}(\mathbf{x})$
relu
weight layer
x
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu

Figure 7. Skip connection [17]

Inspired by VGG-19, the architecture has a 34-layer plain network in which the shortcut or skip connections are added. In addition, the architecture is converted into residual networks with skip connections added [17], [19], as seen in Figure 8.

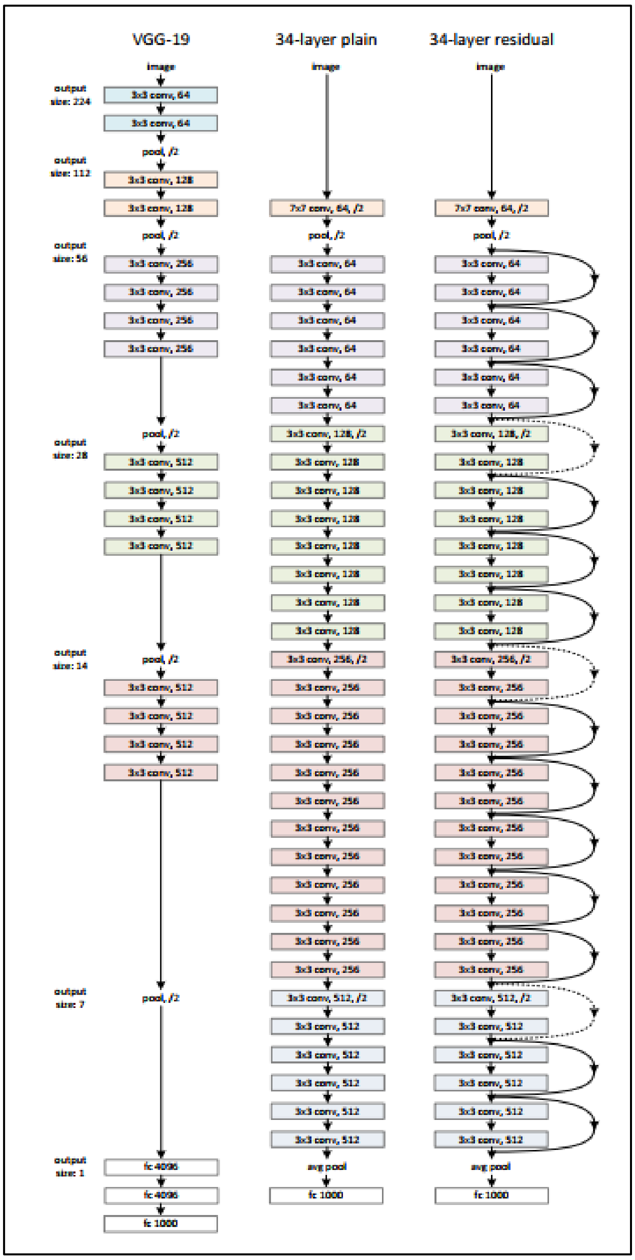

Figure 8. The Architecture of ResNet [17]

2.9. Experiment Using the Baseline FAST R-CNN and Optimized MASK R-CNN

Table 1. Confusion matrix of the baseline FAST R-CNN for the object detection model in 5000 epochs with 70:30 dataset split ratio

	Prediction		
Actual		**Straight Knife**	**Scissor**
	Straight Knife	45	7
	Scissor	6	44

Table 2. Confusion matrix of the optimized MASK R-CNN for the object detection model in 5000 epochs with 70:30 dataset split ratio

	Prediction		
Actual		**Straight Knife**	**Scissor**
	Straight Knife	46	4
	Scissor	5	47

Table 3. Confusion matrix of the baseline FAST R-CNN for the classification model in 5000 epochs with 70:30 dataset split ratio

	Prediction		
Actual		**Clear**	**Unclear**
	Clear	47	5
	Unclear	4	46

Table 4. Confusion matrix of the optimized MASK R-CNN + FAST+SURF + BoVW as validator for the classification model in 5000 epochs with 70:30 dataset split ratio

	Prediction		
Actual		**Clear**	**Unclear**
	Clear	50	5
	Unclear	1	46

Table 5. Results of the baseline and optimized MASK R-CNN with 70:30 dataset split ratio in 5000 epochs

		Object Detection Model	Classifi cation Model
Precision	**Baseline**	86.54%	90.38%
	Optimized	92.00%	90.91%
Recall	**Baseline**	88.24%	92.16%
	Optimized	90.20%	98.04%
F1-Score	**Baseline**	87.38%	91.26%
	Optimized	91.09%	94.34%

Table 5 compares the baseline and optimized algorithms regarding precision, recall, and F1-score for the two-object detection and classification models. The baseline algorithm, FAST R-CNN, achieves a precision of 86.54%, recall of 88.24%, and an F1-score of 87.38% in object detection. This model accurately identifies objects with a balanced trade-off between precision and recall. In contrast, the optimized algorithm, MASK R-CNN, outperforms FAST R-CNN in object detection. It achieves a precision of 92.00%, a recall of 90.20%, and an F1-score of 91.09%. The MASK R-CNN model exhibits improved accuracy, particularly in capturing true positive objects, indicating its effectiveness in object detection. Moving on to the classification model, FAST R-CNN performs well with a precision of 90.38%, a recall of 92.16%, and an F1-score of 91.26%. It demonstrates high accuracy in correctly classifying objects within the given classes.

The combined model, MASK R-CNN + BOVW + FAST-SURF, showcases promising object detection and classification results. It achieves a precision of 90.91%, a recall of 98.04%, and an F1-score of 94.34%. This model demonstrates impressive performance, particularly regarding the recall, indicating its ability to capture a high proportion of true positive objects.

Overall, the results indicate that the optimized model, MASK R-CNN, outperforms the baseline algorithm, FAST R-CNN, in object detection and classification tasks. It exhibits higher precision and comparable or higher recall and F1-score. The combined model, MASK R-CNN + BOVW + FAST-SURF, shows great potential for accurate object detection and classification, especially with its high recall rate.

3. Results

This section provides results and discusses the baseline and proposed model.

3.1. Training Results

The results of the test conducted are discussed; these include training and testing. The dataset will be 238 on training and 102 on testing.

3.1.1. Object Detection Training Results

Upon running the training phase, the results are generated with them; during training, it is well said that 5000 epochs would yield more significant results. Shown in Table 6 are the training results of object detection training; further inspection of the results, the training section of the objection detection with 238 images shows the average precision (AP) of straight knives to be 88% and scissors to be 90%.

It is additionally running a testing phase in object detection after the training has the following results below with 102 images. Straight knife has an AP of 48%, and scissor has an AP of 20%.

Table 6. Object detection training and testing average precision results

	Straight Knife	**Scissor**
Training	88.34%	90.17%
Testing	48.73%	20.91%

3.1.2. Classification Training Results

In Table 7 the classification training results under 238 images are presented. The result shows that the clear category has an AP of 98%, and the unclear has an AP of 96%. This signifies that the classifications in the 238 images have a 98% clear view and are guaranteed to be straight knives or scissors.

In contrast, 96% are unclear because of being obstructed, formed differently, or prospectively oriented.

After generating the results of the training, the testing is subsequent. The results of the classification testing were 102 images. The clear category shows a boastful result of 95% clear, which means the sharp objects are well noticeable and detected. In contrast to 90% unclear, the sharp objects could be better oriented, possibly a scissor or straight knife.

Table 7. Classification training and testing average precision

	Clear	**Unclear**
Training	98.22%	96.48%
Testing	95.86%	90.44%

4. Discussion

This section provides detailed discussion of the collected results after testing the model.

4.1. Learning Curve

In the section dedicated to the loss graph output, the researchers examined the performance of the two models throughout the training process. The loss graph output visually represents the models' loss values, where smaller values indicate better performance. The accompanying figure displays the loss trends for both the training and validation datasets. The blue line represents the loss in the training data, while the orange line illustrates the loss in the validation data. Additionally, the graph allows us to observe the distance between the loss values of the training and validation datasets during training, providing insights into the relationship between the two. Analyzing the loss graphs aids in understanding how effectively the models learn and generalize from the training data to minimize the overall loss.

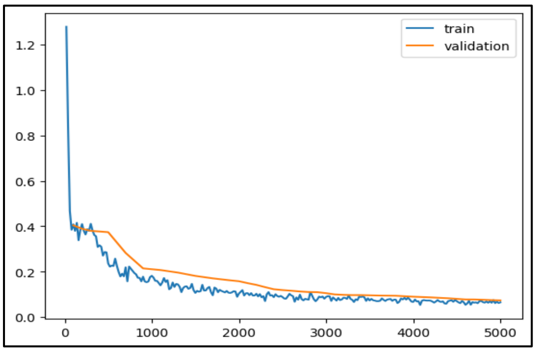

Figure 9. FAST R-CNN object detection model loss graph

Figure 9 shows above the FAST R-CNN baseline model for the object detection, 0 until 2000 epochs, shows that the graph is underfitting; however, 3000 to 4000 epochs show that the validation line seems stable, but visualizing those 5000 to 10000 epochs, the graph will be in good fit.

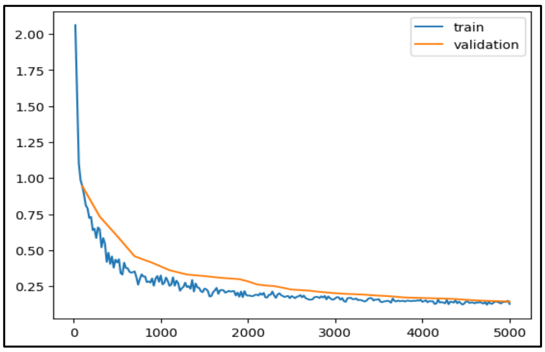

Figure 10. MASK R-CNN object detection model loss graph

Figure 10 shows the optimized MASK R-CNN model for object detection; 0 to 3000 epochs show the graph underfitting. Approaching 4000 to 5000 epochs, the graph shows the results are already a good fit.

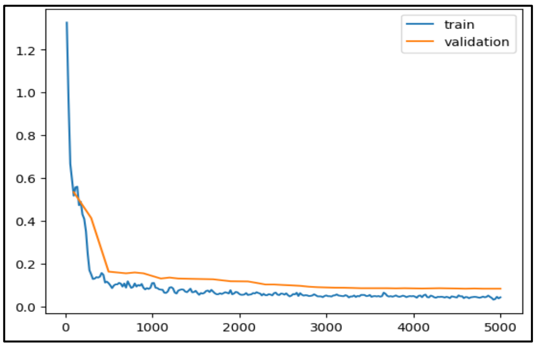

Figure 11. FAST R-CNN classification model loss graph

Figure 11 shows the FAST R-CNN baseline model for classification; 0 to 5000 epochs are still under fit. Visualizing 5000 epochs to 10000 epochs, the graph will still be underfitting. This might be reached until 20000 epochs to be considered so that the graph will show the results as a good fit.

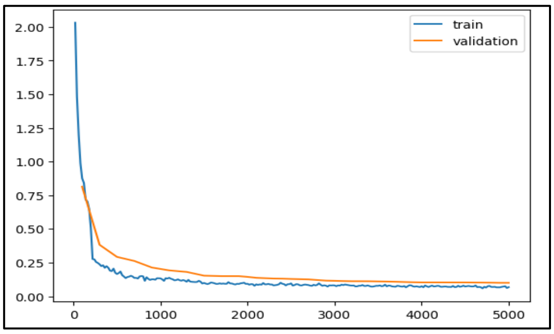

Figure 12. MASK R-CNN + FAST-SURF + BoVW as validator classification model loss graph

Figure 12 shows the optimized MASK R-CNN + FAST-SURF + BoVW as a validator for classification; 0 to 5000 epochs, the graph shows that it is still under fit, visualizing that 5000 until 10000 epochs, the result might be considered already as a good fit.

4.2. Accuracy Output

In this section, the researchers present the performance evaluation of the two models during the training phase. The accuracy metric assesses the models' predictive capability by measuring the proportion of correct predictions over the total data in the split dataset. The accompanying figure showcases the accuracy trends observed in the training and validation datasets. The blue line illustrates the model's accuracy on the training dataset, while the orange line represents the accuracy on the validation dataset. This analysis provides valuable insights into the models' ability to predict outcomes and their generalization capability to unseen data accurately.

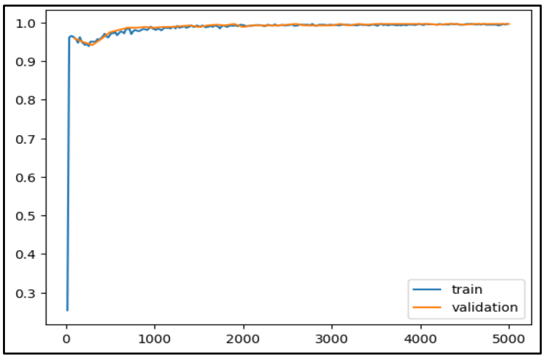

Figure 13. FAST R-CNN object detection model accuracy graph

The accuracy of the object detection fast RCNN baseline model has an outstandingly good fit from 1500 to 5000 epochs as shown in Figure 13. With the accuracy having a good fit, the learning curve is initially unstable.

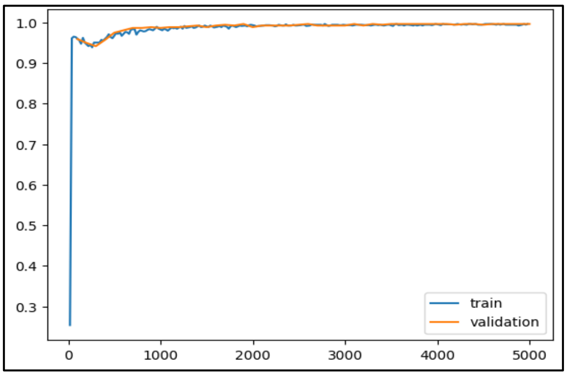

Figure 14. MASK R-CNN object detection model accuracy graph

Figure 14 shows the accuracy of the object detection of the mask R-CNN. While the fast R-CNN did have a stable or good fitting start, the mask R-CNN still has a good fitting past 1000 epochs, making fast R-CNN and mask R-CNN have a slight difference. Mask R-CNN performs better in object detection because of the smooth learning curve.

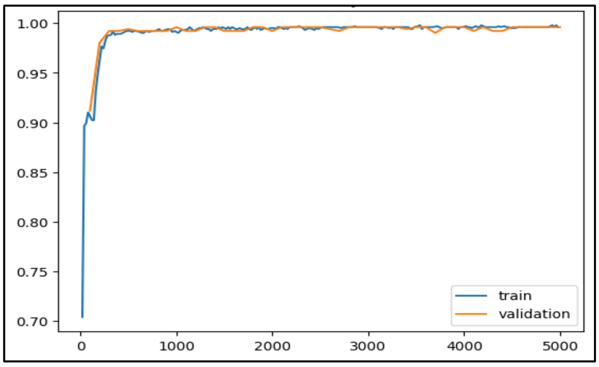

Figure 15. FAST R-CNN classification model accuracy graph

The accuracy of the Fast R-CNN in classification has some noticeable fluctuations; with the classification of Fast R-CNN, the validation line between 3700 epochs to 4500 epochs, the fluctuations are considerable. Nonetheless, the accuracy of the fast R-CNN in classification is a good fit this is shown in Figure 15.

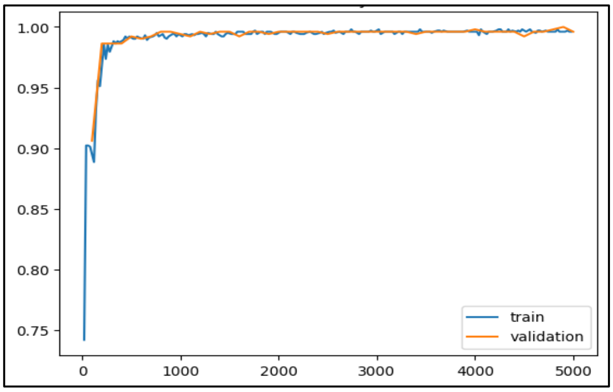

Figure 16. MASK R-CNN + FAST-SURF + BoVW as Validator Classification Model Accuracy Graph

From the beginning, the Mask R-CNN accuracy has no significant fluctuations, just like the fast R-CNN. The Mask R-CNN is a good fitting making it reliable for classifying sharp objects. From 1000 epochs to 5000 epochs, the line is considered together, as shown in Figure 16.

4.3. Object Detection Model Comparison Table Output

Table 8. Object detection model comparison table results in 1000 epochs

Category	FAST R-CNN	Optimized Mask-RCNN
Straight Knife	61.49%	72.36%
Scissor	69.46%	80.12%

Table 8 shows the results of 1000 epochs of the baseline and optimized model in object detection. For the category of the straight knife, the optimized model performs better than the baseline, with a gap of 10.87%. Same as the scissor, the baseline model performs better than the baseline, which has a gap of 10.66%.

Table 9. Object detection model comparison table results in 2000 epochs

Category	FAST R-CNN	Optimized Mask-RCNN
Straight Knife	80.05%	77.63%
Scissor	81.61%	83.22%

Table 9 shows the 2000 epochs of the object detection model of the baseline and optimized. For the category of the straight knife, this time, the baseline model performs better than the optimized one with a gap of 2.42%. In the scissors category, the optimized model performs better than the baseline, with a gap of 1.61%.

Table 10. Object detection model comparison table results in 3000 epochs

Category	FAST R-CNN	Optimized Mark-RCNN
Straight Knife	82.67%	86.46%
Scissor	79.28%	90.40%

Table 10 shows the 3000 epochs of the object detection model of the baseline and optimized. For the category of the straight Knife, the optimized performed better than the baseline, was a gap of 3.82%. In the scissors category, the optimized model performs better than the baseline at 11.12%.

Table 11. Object detection model comparison table results in 4000 epochs

Category	FAST R-CNN	Optimized Mask-RCNN
Straight Knife	82.45%	87.47%
Scissor	84.91%	89.84%

Table 11 shows the 4000 epochs of the object detection model. For the straight knife category, the optimized performs better than the baseline, with a gap of 5.02%. In the scissors category, the optimized achieves better than the baseline, with a gap of 4.92%.

Table 12. Object detection model comparison table results in 5000 epochs

	FAST R-CNN	Optimized Mask-RCNN
Straight Knife	87.99%	90.02%
Scissor	87.16%	91.71%

Table 12 shows the comparison of the 5000 epochs of the object detection model. For the straight knife category, the optimized performs better than the baseline, with a gap of 2.03%. While in the scissors category, the optimized achieves better than the baseline with an interval of 4.55%.

4.4. Classification Model Comparison Table Output

Table 13. Classification model comparison table results in 1000 epochs

Category	FAST R-CNN	Optimized Mask R-CNN + FAST-SURF + BoWV
Clear	86.45%	78.28%
Unclear	81.14%	86.87%

Table 13 shows the 1000 epochs for the classification model. In the clear category, the baseline performs better than the optimized, with a gap of 8.17%. While the unclear category still, the optimized achieves better with an interval of 5.73%.

Table 14. Classification model comparison table results in 2000 Epochs

Category	FAST R-CNN	Optimized Mask R-CNN + FAST-SURF + BoWV
Clear	86.63%	95.27%
Unclear	85.94%	90.16%

Table 14 shows the 2000 epochs of the classification model. In the clear category, the optimized performs better than the baseline, with a gap of 8.64%. While in the unclear category, the optimized achieves better than the optimized with an interval of 4.22%.

Table 15. Classification model comparison table results in 3000 epochs

Category	FAST R-CNN	Optimized Mask R-CNN + FAST-SURF + BoWV
Clear	90.22%	92.41%
Unclear	89.94%	88.91%

Table 15 shows the 3000 epochs of the classification model results. For the clear category, the optimized performs better than the baseline, with a gap of 2.19%. While for the unclear category, the baseline serves better than the baseline with an interval of 1.03%.

Table 16. Classification model comparison table results in 4000 epochs

Category	FAST R-CNN	Optimized Mask R-CNN + FAST-SURF + BoWV
Clear	93.51%	89.72%
Unclear	88.63%	88.13%

Table 16 shows the 4000 epochs for the classification model. For the clear category, the baseline performs better than the optimized, with a gap of 3.79%. While in the unclear category, the baseline performs better than the baseline with a gap of 0.5%.

Table 17. Classification model comparison table results in 5000 epochs

Category	FAST R-CNN	Optimized Mask R-CNN + FAST-SURF + BoWV
Clear	92.57%	97.86%
Unclear	91.48%	89.36%

Table 17 shows the 5000 epochs for the classification model. For the clear category, the optimized performs better than the baseline, with a gap of 5.29%. For the unclear category, the baseline serves better than the baseline, with a gap of 2.12%.

5. Conclusion

The research demonstrates that the optimized versions of the classification and object detection models have achieved significant improvements in accuracy for most categories, fulfilling our specific objectives. Specifically, the FAST-SURF model achieved a 5% improvement for the clear category, and the MASK R-CNN with the BoVW model as a validator achieved a 3% improvement for both the scissor and straight knife detection.

The researchers achieved the objective by dividing an X-ray image into parts using the BoVW model, labeling objects as either clear or unclear, adding BoVW and FAST-SURF object matching algorithms to reduce the false-positive possibilities of the scanned images from an X-ray, and simulating the algorithms for Dual Convolutional Neural Network and Optimized Mask R-CNN using the same X-ray images. These results suggest that optimizing models using appropriate techniques can significantly improve accuracy, which is a crucial factor for the success of many computer vision applications. Future research could explore the effectiveness of other optimization techniques and their impact on model accuracy. Overall, this study has achieved its objectives of improving the accuracy of sharp object detection on security X-ray images.

References:

[1]. Hättenschwiler, N., Sterchi, Y., Mendes, M., & Schwaninger, A. (2018). Automation in airport security X-ray screening of cabin baggage: Examining benefits and possible implementations of automated explosives detection. *Applied ergonomics*, *72*, 58-68.

[2]. Upreti, A., & Rajat, B. (2021). *Automated Threat Detection In X-Ray Imagery For Advanced Security Applications* [Doctoral dissertation, University of Alberta, Canada].

[3]. Gaus, Y. F. A., Bhowmik, N., Akçay, S., Guillén-Garcia, P. M., Barker, J. W., & Breckon, T. P. (2019). Evaluation of a dual convolutional neural network architecture for object-wise anomaly detection in cluttered X-ray security imagery. In *2019 international joint conference on neural networks (IJCNN)*, 1-8. IEEE.

[4]. Kundegorski, M.E., Akçay, S., Devereux, M., Mouton, A., Breckon, T.P. (2016) On using Feature Descriptors as Visual Words for Object Detection within X-ray Baggage Security Screening. *7th International Conference on Imaging for Crime Detection and Prevention.*

[5]. Chen, S. J., Zheng, S. Z., Xu, Z. G., Guo, C. C., & Ma, X. L. (2018). AN IMPROVED IMAGE MATCHING METHOD BASED ON SURF ALGORITHM. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences, 42*(3).

[6]. Kechagias-Stamatis, O., Aouf, N., Nam, D., & Belloni, C. (2017). Automatic X-ray image segmentation and clustering for threat detection. In *Target and Background Signatures III, 10432*, 226-234. SPIE.

[7]. Petrozziello, A., & Jordanov, I. (2019). Automated deep learning for threat detection in luggage from X-ray images. In *International Symposium on Experimental Algorithms* , 505-512. Springer, Cham.

[8]. Wei, Y., Tao, R., Wu, Z., Ma, Y., Zhang, L., Liu, X. (2020). Occluded Prohibited Items Detection: an X-ray Security Inspection Benchmark and De-occlusion Attention Module. In *Proceedings of the 28th ACM international conference on multimedia*, 138-146.

[9]. Li, A., Jiang, W., Yuan, W., Dai, D., Zhang, S., & Wei, Z. (2017). An improved FAST+ SURF fast matching algorithm. *Procedia Computer Science, 107,* 306-312.

[10]. Xu, Y., Yu, X., Wang, T., & Xu, Z. (2020). Pooling region learning of visual words for image classification using bag-of-visual-words model. *PLoS One, 15*(6).
Doi: http://dx.doi.org/10.1371/journal.pone.0234144

[11]. Davida, B. (2018). *Bag of Visual Words in a Nutshell.*Medium. Retrieved from:
https://towardsdatascience.com/bag-of-visual-words-in-a-nutshell-9ceea97ce0fb
[accessed: 10 September 2023].

[12]. Viswanathan, D. G. (2009). Features from accelerated segment test (fast). In *Proceedings of the 10th workshop on image analysis for multimedia interactive services, London, UK*, 6-8.

[13]. Ali, F., Khan, S. U., Mahmudi, M. Z., & Ullah, R. (2016). A comparison of FAST, SURF, Eigen, Harris, and MSER features. *International Journal of Computer Engineering and Information Technology*, *8*(6), 100.

[14]. Setiawan, A., Yunmar, R. A., & Tantriawan, H. (2020). Comparison of speeded-up robust feature (SURF) and oriented FAST and rotated BRIEF (ORB) methods in identifying museum objects using low light intensity images. *IOP Conference Series. Earth and Environmental Science,537*(1). Doi: 10.1088/1755-1315/537/1/012025

[15]. Ahmed, B., Gulliver, T. A., & alZahir, S. (2020). Image splicing detection using mask-RCNN. *Signal, image and video processing, 14,* 1035-1042.

[16]. Wada, K. Labelme et al. (2021). Image Polygonal Annotation with Python [Computer software]. *Zenodo*. Doi: 10.5281/zenodo.5711226

[17]. Shakhadri, S. A. G. (2021). *Build ResNet from Scratch With Python !* Analytics Vidhya. Retrieved from: https://www.analyticsvidhya.com/blog/2021/06/build-resnet-from-scratch-with-python/ [accessed: 12 September 2023].

[18]. He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, 2961-2969.

[19]. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770-778.