Identifying threat objects using faster region-based convolutional neural networks (faster r-cnn)

Reagan Galvez [a,1,*], Elmer Pamisa Dadios [b,2]

[a] Electronics Engineering Department, Bulacan State University, City of Malolos, Bulacan 3000, Philippines
[b] Manufacturing Engineering and Management Department, De La Salle University, Taft Avenue, Manila 1004, Philippines
[1] reagan.galvez@bulsu.edu.ph; [2] elmer.dadios@dlsu.edu.ph
* corresponding author

ARTICLE INFO

Article history
Selected paper from The 2021 4th International Symposium on Advanced Intelligent Informatics (SAIN'21), Yogyakarta (Virtually), October 13-14, 2021, http://sain.ijain.org/2021/. Peer-reviewed by SAIN'21 Scientific Committee and Editorial Team of IJAIN journal

Received September 13, 2021
Revised September 27, 2022
Accepted November 30, 2022
Available online November 30, 2022

Keywords
Computer vision
Convolutional neural networks
Faster r-cnn
Threat object detection
X-ray imaging

ABSTRACT

Automated detection of threat objects in a security X-ray image is vital to prevent unwanted incidents in busy places like airports, train stations, and malls. The manual method of threat object detection is time-consuming and tedious. Also, the person on duty can overlook the threat objects due to limited time in checking every person's belongings. As a solution, this paper presents a faster region-based convolutional neural network (Faster R-CNN) object detector to automatically identify threat objects in an X-ray image using the IEDXray dataset. The dataset was composed of scanned X-ray images of improvised explosive device (IED) replicas without the main charge. This paper extensively evaluates the Faster R-CNN architecture in threat object detection to determine which configuration can be used to improve the detection performance. Our findings showed that the proposed method could identify three classes of threat objects in X-ray images. In addition, the mean average precision (mAP) of the threat object detector could be improved by increasing the input image's image resolution but sacrificing the detector's speed. The threat object detector achieved 77.59% mAP and recorded an inference time of 208.96 ms by resizing the input image to 900 × 1536 resolution. Results also showed that increasing the bounding box proposals did not significantly improve the detection performance. The mAP using 150 bounding box proposals only achieved 75.65% mAP, and increasing the bounding box proposal twice reduced the mAP to 72.22%.

1. Introduction

Terrorist attacks in many countries result in the injury and deaths of civilians and even military personnel [1]. In the Philippines, this problem is also dominant due to the terrorist attacks that happened recently [2] caused by the use of an improvised explosive device (IED). IED is a homemade explosive device used by perpetrators designed to harm people. Generally, IED contains a power source, switch, initiator, wires, and main charge. The power source, commonly a 9 volts battery, provides power to the initiator (electric or non-electric) to start the detonation of the main charge. The arming or firing of the IED is controlled by the switch.

In the Global Terrorism Index 2022, the Philippines was listed in the top 20 countries most impacted by terrorism [3]. As a safety measure, tightened security in public transport systems such as airport terminals, train stations, and also in commercial establishments is strictly implemented. Pieces of baggage are scanned using an X-ray machine to identify the objects inside and look for threats like explosives and bladed weapons. Although this process is valid, the possibility of missed detection is high

https://doi.org/10.26555/ijain.v8i3.952

during rush hour because of the limited time to scan thousands of baggage and identify threat objects [4]. As a solution, this paper used Faster Region-based Convolutional Neural Network (Faster R-CNN) to identify threat objects (e.g., battery, mortar, wires) in an X-ray image to aid the operator in deciding whether a piece of baggage poses a threat or not. Faster R-CNN [5] is a deep learning-based object detector from the family of a region-based convolutional neural network that introduces Region Proposal Networks (RPN). This network accepts a feature map and then outputs object proposals (bounding box) with corresponding objectness scores.

To date, several studies in the computer vision field explored Faster R-CNN in many different applications such as vehicle detection [6], disease detection [7], [8], face detection [9], [10], ship detection [11], [12], metal object detection [13], radar images [14], defect detection [15], [16], object detection on medical images [17], [18], and autonomous driving [19]. Although many researchers successfully implemented Faster R-CNN in object detection, there are few studies [20] that explored this detector for X-ray images due to limited data available and complicated procedures in collecting X-ray images. Some researchers used a different approach [21], like improved Mask R-CNN [22], X-ray proposal and discriminative Networks [23], and multi-view branch-and-bound search algorithm [24] for object detection in X-ray images. Researchers in [25] and [26] were able to implement a deep learning-based object detector for identifying threat objects such as IEDs. However, a detailed evaluation is still needed to know the right configuration and trade-offs.

The contributions of this paper are as follows: (a) extensive evaluation of Faster R-CNN architecture in threat object detection, (b) investigation of how the bounding box proposals and image resolution affects the performance of the treat object detector, (c) experiments on how to improve the performance of the threat object detector in terms of mean average precision (mAP) and speed.

2. Method

The overview of the Faster R-CNN architecture for identifying threat objects is shown in Fig. 1.

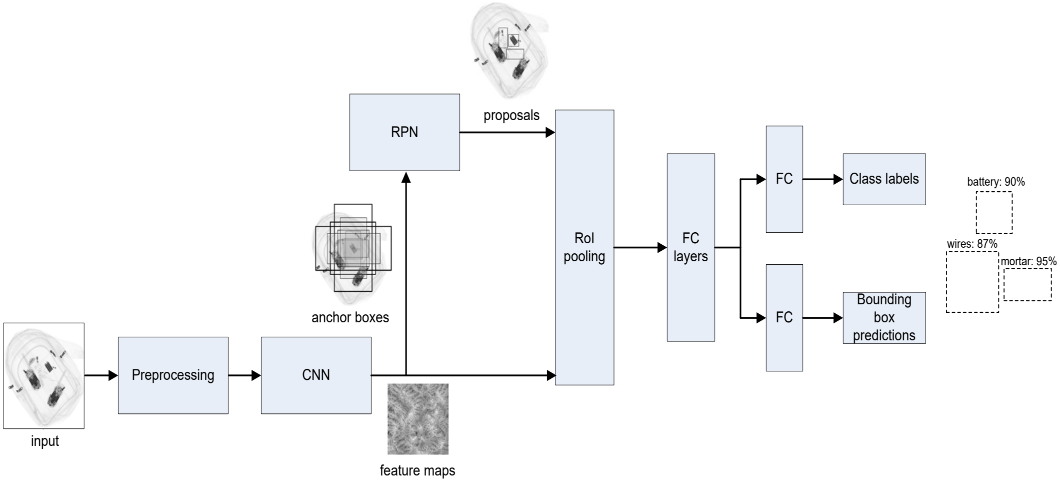

Fig. 1. Faster R-CNN architecture

The input is an X-ray image with corresponding class labels and bounding boxes. X-ray images are fed to the preprocessing stage, such as resizing and augmentation before feature extraction. Data augmentation performs random geometric transformations to the image to increase the training data. Features are extracted using CNN via transfer learning using ResNet-101 [27] as a base network. The RPN module accepts anchor boxes and looks for possible objects in the image. The anchor boxes serve as a reference at multiple scales (e.g., 64 × 64, 128 × 128, and 256 × 256) and aspect ratios (e.g., 1:1, 2:1, 1:2). Each sliding window contains nine anchor boxes centered at every position. Then, the RPN module determines its objectness score and proposed regions where the objects are possibly located. The objectness score measures the probability that an anchor is an object. The output of the RPN module is bounding box proposals, each having an objectness score. The region of interest (ROI) pooling module accepts the top N proposals from the RPN module and extracts fixed-sized windows of ROI features from the feature maps. The N proposals were varied from 10 to 450 to determine the effect on the detection performance. The ROI pooling module resizes the feature map into 14 × 14 × D, where D is the depth of the feature map. When max pooling is applied with a stride of 2, the result is a 7 × 7 × D feature vector that will be fed to two fully connected (FC) layers and then finally passed to two fully connected layers that yield the class label and bounding box. Class label C has four dimensions (3 classes + 1 background) such as the battery, mortar, and wires, while the bounding boxes are twelve (4 coordinates ×3 classes).

2.1. Dataset

Dataset collection was done using a dual-view X-ray machine. In order to capture the X-ray images projected to the computer monitor, a video recorder was used. The images were collected by extracting one out of five frames (20%) in a given video file to ensure that the extracted images were not similar to the previous image. As an example, in a 60-second video with a frame rate of 30 frames per second (fps), the extracted images will be 360 images. Once extracted, the images were manually selected based on the clarity and quality of the image. Finally, the images were labeled according to classes using LabelImg [28]. The dataset was called IEDXray [25], as shown in Fig. 2, which is composed of X-ray images of IED replicas without the main charge. The left part of the figure shows the one-channel histogram (grayscale) of the sample X-ray image. The histogram shows that the pixel intensities of the image were concentrated approximately between 200 to 255 (white pixels). This dataset contains the basic circuitry of an IED without explosive material. Six IED types were scanned in the X-ray machine.

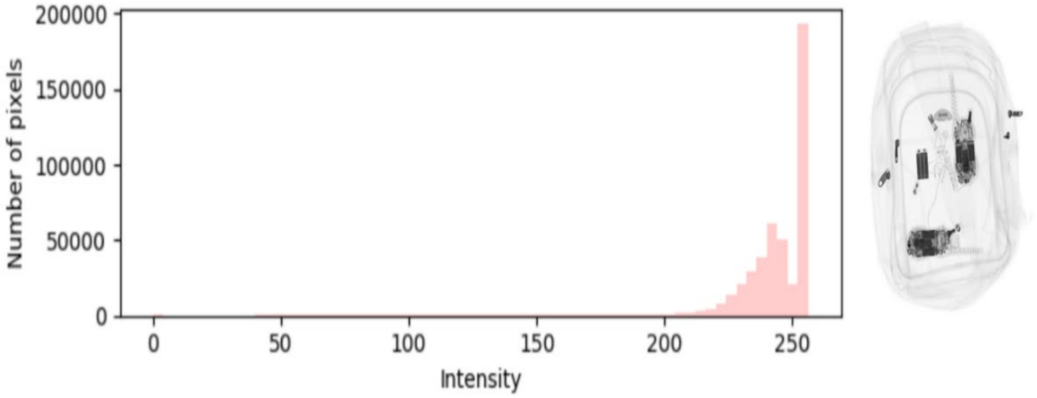

Fig. 2. IEDXray dataset.

2.2. Training and Evaluation

Faster R-CNN was trained using stochastic gradient descent (SGD) with momentum. Momentum is a method used to improve convergence speed and reduce oscillation [29]. Several hyperparameter values were tried during the experiment using the manual search method. The highest mAP was achieved using the following hyperparameter values: learning rate = 0.0003, momentum = 0.9, batch size = 1. Regularization was also added to the model to increase the mAP by augmenting the data passed into the network for training. Data augmentation was used as an implicit regularization [30]. Each experiment was trained for 20,000 steps. The IEDXray dataset was divided into train and test data. Train and test data consist of 1,209 and 134 images, respectively. Then, the evaluation metric used to measure the performance of Faster R-CNN in threat object detection was based on the PASCAL VOC metric [31], which uses the equations (1), (2), and (3) to compute the mean average precision (mAP).

Intersection over Union (IoU) was calculated by dividing the area of intersection between ground-truth XG and predicted bounding box XP to the area of union shown in (1). To be considered as correct detection or true positive (TP), the score should have an IoU > 0.5 [31]; otherwise, it is a false positive (FP). A false negative (FN) is recorded for undetected ground truths.

$$IOU = \frac{A(X_G \cap X_P)}{A(X_G \cup X_P)} \tag{1}$$

Then, the precision P and recall R values were calculated to compute the average precision AP. P in (2) measures the percentage of correct positive predictions, and R in (3) measures the ability of the model to find all ground-truth bounding boxes. Where TP, FP, and FN are true positive, false positive, and false negative, respectively.

$$P = \frac{TP}{TP+FP} \tag{2}$$

$$R = \frac{TP}{TP+FN} \tag{3}$$

Given that the average precision *AP* is the precision *P* averaged across all recall *R* values between 0 and 1, the mAP in (4) can be computed by averaging the *AP* of all class *C* (3 classes). The classes were battery, mortar, and wires.

$$mAP = \frac{1}{C} \sum_{i=1}^{C} AP_i \tag{4}$$

2.3. Hardware and Software Setup

All of the experiments were conducted on a desktop computer with Intel Core i7-9700K 3.6 GHz 8-Core Processor, 16GB RAM, using Ubuntu 18.04 LTS with NVIDIA RTX 2070 8GB graphics processing unit in a Tensorflow framework

3. Results and Discussion

In this research, two important parameters of the Faster R-CNN were investigated, such as the number of bounding box proposals generated by the RPN and the image resolution of the input image using the IEDXray dataset that was discussed in the previous section.

3.1. Bounding Box Proposals

In the experiment, the number of bounding box proposals varied between 10 and 450 to explore the trade-off. Table 1 illustrates the mAP and evaluation time (per image) of Faster R-CNN on the different number of bounding box proposals. The mean average precision (mAP) was calculated from the last training step (20,000), while the evaluation time was measured by averaging the time it takes to evaluate the test data. The notation (e.g., $AP_{battery}$) is the average precision of each class. It can be seen from the table that changing the number of bounding box proposals in each training results in different values of mAP. The highest value was achieved using 150 bounding box proposals (75.65%), with a small difference when using 75 bounding box proposals (75.10%). What is interesting about the data is that using 75 bounding box proposals reduces the evaluation time by 29.85 ms (22.22%) and still has a comparable mAP as 150 bounding box proposals. On the other hand, increasing the number of bounding box proposals from 150 to 450 recorded a 1.16% decrease in mAP. Therefore, increasing the number of bounding box proposals does not always improve the mAP of the object detector.

Table 1. Faster R-CNN performance on the different number of bounding box proposals

bounding box proposal	mAP	APbattery	APmortar	APwires	time(ms)
10	0.6733	0.6923	0.9885	0.3391	89.55
75	0.7510	0.7381	0.9874	0.5274	104.48
100	0.7359	0.7034	0.9862	0.5180	126.87
150	**0.7565**	0.7292	0.9862	0.5540	134.33
300	0.7222	0.6843	0.9828	0.4994	171.64
450	0.7449	0.7374	0.9828	0.5146	216.42

The precision and recall in each class using 150 bounding box proposals are shown in Table 2. It can be seen that the Faster R-CNN detected the mortar with high precision (96.67%) and high recall (100%). While the wires were not accurately detected with 87.41% precision and 65.10% recall.

Table 2. Precision and recall (150 bounding box proposals)

class	precision (%)	recall (%)
battery	97.09	63.29
wires	87.41	65.10
mortar	96.67	100

The performance of Faster R-CNN in each bounding box proposal during the evaluation is shown in Fig. 3. It can be seen that the mAP using 10 bounding box proposals significantly reduces the performance of the object detector.

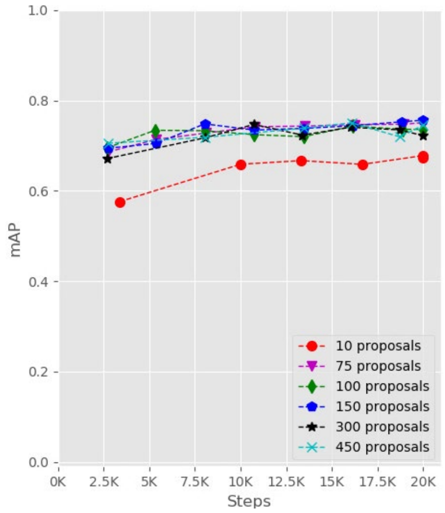

Fig. 3. mAP plot on different bounding box proposals.

The inference time in each bounding box proposal was also evaluated. The comparison of mAP versus time on the different number of bounding box proposals is presented in Fig. 4. Using 450 bounding box proposals gives the slowest inference time, while 10 bounding box proposals are the fastest but give the lowest mAP. The graph indicates that it is recommended to use 75 bounding box proposals to get the best trade-off between speed and mAP.

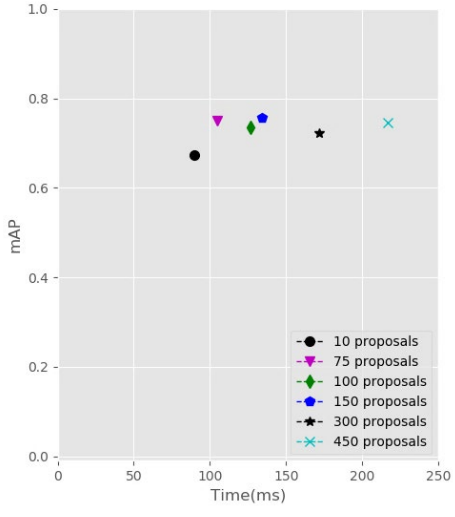

Fig. 4. mAP vs. time on different bounding box proposals.

3.2. Image Resolutions

In this experiment, the input image resolutions were varied between 150 × 256 and 900 × 1536. Then, the Faster R-CNN was trained using these image resolutions. Table 3 shows the performance of Faster R-CNN on different image resolutions. The aspect ratio of all resolutions was fixed (75/128), while the number of proposals was 300. It can be seen from the table that as the image resolution gets bigger, the mAP increases. The highest mAP was achieved using 900 × 1536 resolution (77.59%) in exchange for lower speed. Increasing the resolution by a factor of 2 (from 150 × 256 to 300 × 512) increases the mAP by 16.42% while increasing it to a factor of 4 (from 150 × 256 to 600 × 1024) increases the mAP by 27.09%. In addition, there is no change in evaluation time if 150 × 256 or 300 × 512 resolution is used, but the mAP in 300 × 512 is higher than 150 × 256. The table clearly shows that image resolution can significantly impact the mAP of the object detector.

Table 3. Faster R-CNN performance on different image resolutions

resolution	mAP	AP$_{battery}$	AP$_{mortar}$	AP$_{wires}$	time(ms)
150 × 256	0.4513	0.2739	0.9469	0.1332	149.25
300 × 512	0.6155	0.5196	0.9711	0.3558	149.25
450 × 768	0.6984	0.6745	0.9805	0.4402	156.72
600 × 1024	0.7222	0.6843	0.9828	0.4994	171.64
750 × 1280	0.7563	0.7527	0.9828	0.5335	186.57
900 × 1536	**0.7759**	0.7739	0.9740	0.5799	208.96

The precision and recall in each class using 900 × 1536 resolution are shown in Table 4. It can be seen that the Faster R-CNN detected the mortar with high precision (93.55%) and high recall (100%). While the wires were not accurately detected with 77.84% precision and 75% recall.

Table 4. Precision and recall (900 × 1536 Resolution)

class	precision (%)	recall (%)
battery	95.65	69.62
wires	77.84	75
mortar	93.55	100

The mAP plot on different image resolutions is shown in Fig. 5. Interestingly, the image size was observed to affect the performance of the object detector. Increasing the image size also increases the mAP of the object detector.

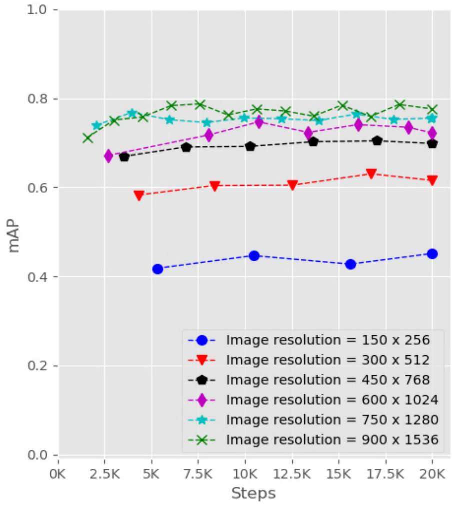

Fig. 5. mAP plot on different image resolutions.

Same with the bounding box proposal experiment, the inference time in different image resolutions was also examined. The comparison of mAP versus time on different image resolutions is presented in Fig. 6. The increased mAP can be achieved by sacrificing the speed of the object detector. Every 150 pixels increase in the shorter edge, and 256 pixels increase in the other edge of the input image increases the mAP while the evaluation speed slows down.

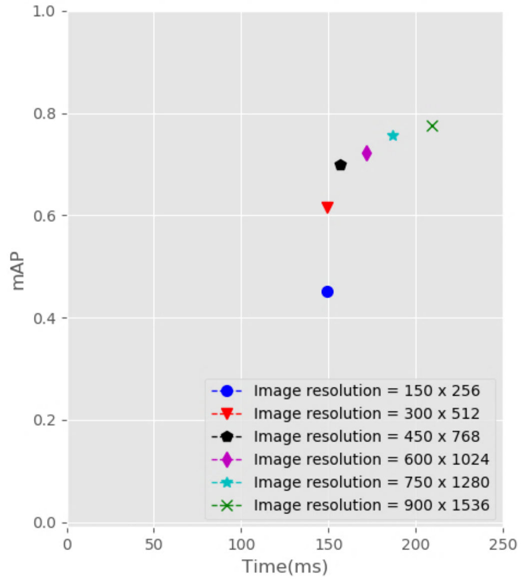

Fig. 6. mAP vs. time on different image resolutions.

After training and evaluating the Faster R-CNN, the trained model was tested in an X-ray image to verify its detection performance. A python script was developed that accepts an input image, performs inference, and outputs the bounding box coordinates and corresponding class labels of the threat objects. The detection output using Faster R-CNN is shown in Fig. 7. The class label and class score of the detected objects are shown in the upper portion of the bounding box coordinates. The model was able to detect three classes of IED components, such as battery, mortar, and wires.

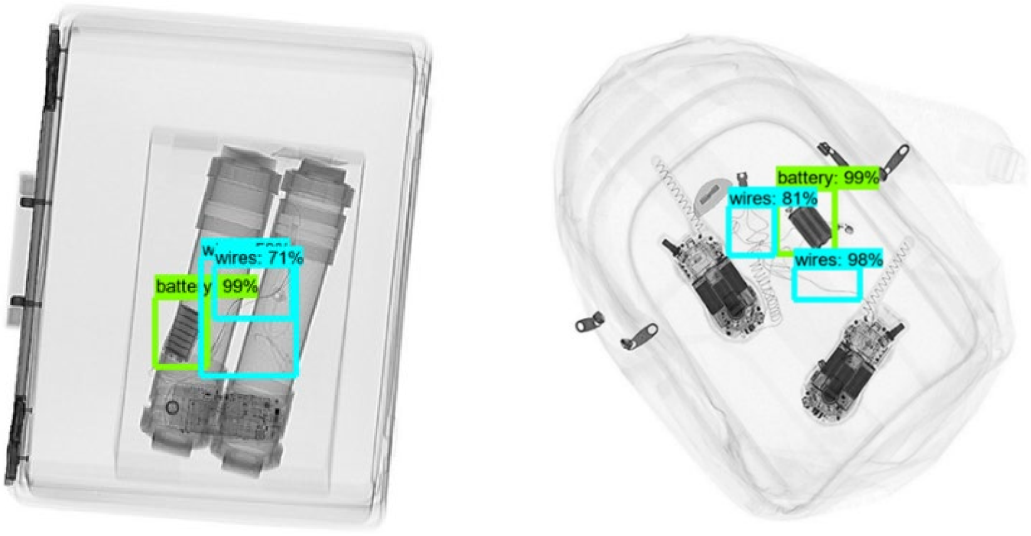

Fig. 7. Detection output in the X-ray images

4. Conclusion

This study extensively evaluated Faster R-CNN in identifying threat objects in an X-ray image dataset. Different experiments were conducted to increase the performance of the threat object detector by changing the number of bounding box proposals and the image resolution of the input image. These experiments confirmed that increasing the number of bounding box proposals may lower the mean average precision (mAP) and slows the detection time. The research has also shown that increasing the input image's size positively impacts the mAP by sacrificing speed. It is recommended to identify the best trade-off between the mAP and speed when using Faster R-CNN by balancing the bounding box proposals and the image size. Overall, the experiment result shows that the proposed method can reliably identify the threat object in an X-ray image.

More X-ray images can be added to the training data to improve this study further. The data is recommended to have other objects aside from the IED components. This may increase the generalizability of the IED detector model and prevent several false positives and negatives. If acquiring additional data is impossible, another option is to generate synthetic X-ray images using another machine learning framework like generative adversarial networks (GANs) and variational autoencoders (VAEs).

Acknowledgment

The authors thank Bulacan State University, De la Salle University, and the Engineering Research and Development for Technology (ERDT), Department of Science and Technology (DOST) for their financial support while doing this research.

Declarations

Author contribution. Reagan Galvez performed the manuscript revision, data acquisition, training, and evaluation. Elmer Dadios provided consultations to improve the content of the paper.
Funding statement. The Philippine Council for Industry, Energy, and Emerging Technology Research and Development (PCIEERD) funded the research under Project No. 05464.
Conflict of interest. The authors declare no conflict of interest.
Additional information. No additional information is available for this paper.

References

[1] C. Schmeitz, D. Barten, K. Van Barneveld, H. De Cauwer, L. Mortelmans, F. Van Osch, J. Wijnands, E. C. Tan, and A. Boin, "Terrorist Attacks Against Emergency Medical Services: Secondary Attacks are an Emerging Risk," *Prehos. Disast. Med.*, vol. 37, no. 2, pp. 185-191, 2022, doi: 10.1017/S1049023X22000140.

[2] S. Buigut, B. Kapar, and U. Braendle, "Effect of regional terrorism events on Malaysian tourism demand," *Tour. and Hospit. Res.*, vol. 22, no 3., pp. 271–283.

[3] Institute for Economics & Peace, "Global terrorism index 2018: measuring the impact of terrorism." 2022, Accessed : Dec, 20, 2022. [Online]. Available : http://visionofhumanity.org/reports/

[4] V. Riffo, S. Flores, and D. Mery, "Threat Objects Detection in X-ray Images Using an Active Vision Approach," *J. Nondestruct. Eval.*, vol. 36, no. 3, p. 44, Sep. 2017, doi: 10.1007/s10921-017-0419-3.

[5] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, doi: 10.1109/TPAMI.2016.2577031.

[6] H. Ji, Z. Gao, T. Mei, and Y. Li, "Improved faster r-cnn with multiscale feature fusion and homography augmentation for vehicle detection in remote sensing images," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 11, pp. 1761–1765, 2019, doi: 10.1109/LGRS.2019.2909541.

[7] G. Zhou, W. Zhang, A. Chen, M. He, and X. Ma, "Rapid detection of rice disease based on FCM-KM and faster r-cnn fusion," *IEEE Access*, vol. 7, pp. 143190–143206, 2019, doi: 10.1109/ACCESS.2019.2943454.

[8] F. Deng, W. Mao, Z. Zeng, H. Zeng, and B. Wei, "Multiple diseases and pests detection based on federated learning and improved faster R-CNN," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–11, 2022, doi: 10.1109/TIM.2022.3201937.

[9] W. Wu, Y. Yin, X. Wang, and D. Xu, “Face detection with different scales based on Faster R-CNN,” *IEEE Trans. Cybern.*, vol. 49, no. 11, pp. 4017–4028, Nov. 2019, doi: 10.1109/TCYB.2018.2859482.

[10] P. J. Lu and J.-H. Chuang, “Fusion of multi-intensity image for deep learning-based human and face detection,” *IEEE Access*, vol. 10, pp. 8816–8823, 2022, doi: 10.1109/ACCESS.2022.3143536.

[11] Z. Lin, K. Ji, X. Leng, and G. Kuang, “Squeeze and excitation rank Faster R-CNN for ship detection in SAR images,” *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 5, pp. 751–755, May 2019, doi: 10.1109/LGRS.2018.2882551.

[12] Y. Li, S. Zhang, and W.-Q. Wang, “A lightweight faster R-CNN for ship detection in SAR images,” *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022, doi: 10.1109/LGRS.2020.3038901.

[13] R. Gao *et al.*, “Small foreign metal objects detection in X-Ray images of clothing products using faster R-CNN and feature pyramid network,” *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–11, 2021, doi: 10.1109/TIM.2021.3077666.

[14] R. Gonzales-Martinez, J. Machacuay, P. Rotta, and C. Chinguel, “Hyperparameters tuning of faster R-CNN deep learning transfer for persistent object detection in radar images,” *IEEE Lat. Am. Trans.*, vol. 20, no. 4, pp. 677–685, Apr. 2022, doi: 10.1109/TLA.2022.9675474.

[15] Y. Zhang, Z. Zhang, K. Fu, and X. Luo, “Adaptive defect detection for 3-D printed lattice structures based on improved faster R-CNN,” *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–9, 2022, doi: 10.1109/TIM.2022.3200362.

[16] F. Selamet, S. Cakar, and M. Kotan, “Automatic detection and classification of defective areas on metal parts by using adaptive fusion of faster R-CNN and shape from shading,” *IEEE Access*, vol. 10, pp. 126030–126038, 2022, doi: 10.1109/ACCESS.2022.3224037.

[17] Y. Liu, Z. Ma, X. Liu, S. Ma, and K. Ren, “Privacy-preserving object detection for medical images with faster R-CNN,” *IEEE Trans. Inf. Forensics Secur.*, vol. 17, pp. 69–84, 2022, doi: 10.1109/TIFS.2019.2946476.

[18] Z. Qian *et al.*, “A new approach to polyp detection by pre-processing of images and enhanced faster R-CNN,” *IEEE Sens. J.*, vol. 21, no. 10, pp. 11374–11381, May 2021, doi: 10.1109/JSEN.2020.3036005.

[19] G. Wang, J. Guo, Y. Chen, Y. Li, and Q. Xu, “A PSO and BFO-based learning strategy applied to Faster R-CNN for object detection in autonomous driving,” *IEEE Access*, vol. 7, pp. 18840–18859, 2019, doi: 10.1109/ACCESS.2019.2897283.

[20] S. Akcay, 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 Trans. Inf. Forensics Secur.*, vol. 13, no. 9, pp. 2203–2215, Sep. 2018, doi: 10.1109/TIFS.2018.2812196.

[21] D. Mery, D. Saavedra, and M. Prasad, “X-Ray baggage inspection with computer vision: a survey,” *IEEE Access*, vol. 8, pp. 145620–145633, 2020, doi: 10.1109/ACCESS.2020.3015014.

[22] J. Zhang, X. Song, J. Feng, and J. Fei, “X-Ray image recognition based on improved Mask R-CNN algorithm,” *Math. Probl. Eng.*, vol. 2021, pp. 1–14, Sep. 2021, doi: 10.1155/2021/6544325.

[23] B. Gu, R. Ge, Y. Chen, L. Luo, and G. Coatrieux, “Automatic and robust object detection in X-Ray baggage inspection using deep convolutional neural networks,” *IEEE Trans. Ind. Electron.*, vol. 68, no. 10, pp. 10248–10257, Oct. 2021, doi: 10.1109/TIE.2020.3026285.

[24] M. Baştan, “Multi-view object detection in dual-energy X-ray images,” *Mach. Vis. Appl.*, vol. 26, no. 7–8, pp. 1045–1060, Nov. 2015, doi: 10.1007/s00138-015-0706-x.

[25] R. L. Galvez, E. P. Dadios, A. A. Bandala, and R. R. P. Vicerra, “Object detection in x-ray images using transfer learning with data augmentation,” *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 9, no. 6, p. 2147, Dec. 2019, doi: 10.18517/ijaseit.9.6.9960.

[26] R. L. Galvez and E. P. Dadios, “Threat object detection and analysis for explosive ordnance disposal robot,” *Glob. J. Eng. Technol. Adv.*, vol. 11, no. 1, pp. 078–087, Apr. 2022, doi: 10.30574/gjeta.2022.11.1.0074.

[27] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[28] Tzutalin, “LabelImg,” *Github*. 2015, [Online]. Available : https://github.com/tzutalin/labelImg

[29] N. Qian, “On the momentum term in gradient descent learning algorithms,” *Neural Networks*, vol. 12, no. 1, pp. 145–151, 1999, doi: 10.1016/s0893-6080(98)00116-6.

[30] A. Hernandez-Garcia and P. König, “Data augmentation instead of explicit regularization,” *CoRR*, vol. abs/1806.0, 2018, [Online]. Available : http://arxiv.org/abs/1806.03852.

[31] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, “The pascal visual object classes (VOC) challenge,” *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010, doi: 10.1007/s11263-009-0275-4.