International Journal on Advanced Science Engineering Information Technology

Vol.9 (2019) No. 6
ISSN: 2088-5334

Object Detection in X-ray Images Using Transfer Learning with Data Augmentation

Reagan L. Galvez[#1], Elmer P. Dadios[#2], Argel A. Bandala[#3], Ryan Rhay P. Vicerra[#4]

[#] *Gokongwei College of Engineering, De La Salle University, Manila, 1004, Philippines*
E-mail: [1]reagan_galvez@dlsu.edu.ph; [2]elmer.dadios@dlsu.edu.ph; [3]argel.bandala@dlsu.edu.ph; [4]ryan.vicerra @dlsu.edu.ph

***Abstract*— Object detection in X-ray images is an interesting problem in the field of machine vision. The reason is that images from an X-ray machine are usually obstructed with other objects and to itself, therefore object classification and localization is a challenging task. Furthermore, obtaining X-ray data is difficult due to an insufficient dataset available compared with photographic images from a digital camera. It is vital to easily detect objects in an X-ray image because it can be used as decision support in the detection of threat items such as improvised explosive devices (IED's) in airports, train stations, and public places. Detection of IED components accurately requires an expert and can be achieved through extensive training. Also, manual inspection is tedious, and the probability of missed detection increases due to several pieces of baggage are scanned in a short period of time. As a solution, this paper used different object detection techniques (Faster R-CNN, SSD, R-FCN) and feature extractors (ResNet, MobileNet, Inception, Inception-ResNet) based on convolutional neural networks (CNN) in a novel IEDXray dataset in the detection of IED components. The IEDXray dataset is an X-ray image of IED replicas without the explosive material. Transfer learning with data augmentation was performed due to limited X-ray data available to train the whole network from scratch. Evaluation results showed that individual detection achieved 99.08% average precision (AP) in mortar detection and 77.29% mAP in three IED components.**

***Keywords*— convolutional neural networks; data augmentation; object detection; transfer learning; X-ray image.**

I. INTRODUCTION

According to the data retrieved from the Global Terrorism Database (GTB) [1], there were 10,900 terrorist attacks around the world in 2017 that killed more than 26,400 individuals. In the Philippines, 692 terrorist incidents were recorded with 496 deaths and 674 injuries. Armed assault and bombing have the highest number of incidents, which are 257 and 168, respectively. With this data, the Philippines was listed in the top 4 for the most number of terrorist incidents in 2017. To prevent terrorist incidents, many establishments, train stations, and airport terminals implemented tight security measures. Most train stations and airport terminals have an X-ray machine at the entrance to scan the bags of every passenger. The task of the operator is to look for threat objects like firearms, knife, and explosives.

Improvised explosive device (IED) is commonly used by perpetrators in many countries to harm people. The main reason is that it is simple to construct, and most of its components can be acquired easily. IED is placed usually inside the bag, box, or any material that will conceal them. One way to identify IED is to scan the unknown object using an X-ray machine to see all the items inside. Then the operator, based on their knowledge decides whether the unidentified object is an IED or not. However, this process is tiresome and time-consuming. Furthermore, the possibility of missed detection increases over time due to exhausted personnel. As a solution, this paper used different object detection models based on convolutional neural networks (CNN) to detect the components of an IED in X-ray images.

Some researchers used classic object detection algorithms such as histograms of oriented gradient (HOG) [2] to extract features in human detection and mean-shift based blob analysis and tracking to identify and track vehicles approaching an intersection in real-time [3]. In [4] used fuzzy logic to determine the class and color of the vehicle. However, due to the rapid development and robust performance of object detection algorithms based on CNN, this is now widely used.

Detection of objects in an X-ray image is a challenging problem in computer vision. Some reasons are due to occlusion of the target object to other objects, self-occlusion, background clutter, and viewpoint variation due to object rotation. Another challenge is that certain datasets of X-ray images are not publicly available, and it is difficult to generate because the X-ray equipment is expensive compared to the digital camera.

There are few studies about the detection of an object in X-ray images. In [5] used deep convolutional neural networks to detect multiple objects such as gun, laptop, knife,

camera, gun component, and ceramic knife. The architectures used for the detection of these objects are Sliding Window based CNN (SW-CNN) [6], Faster R-CNN, Region-based Fully Convolutional Networks (R-FCN) and, You Only Look Once (YOLO) V2 [7]. In [8] presented a multi-view branch-and-bound algorithm for multi-view object detection such as laptop, handgun, and glass bottle using standard local features in a bag of visual words (BoW) framework with linear structural Support Vector Machine (SVM). Other researchers introduced an active vision approach [9] and the Adapted Implicit Shape Model (AISM) [10] to detect threat objects in X-ray images in GDXray [11] database. Furthermore, [12] used an attention mechanism based on CNN to identify the prohibited objects in airport X-ray images. Based on the aforementioned studies, the authors focused mostly on the detection of familiar objects such as laptops, cameras, and glass bottle, which does not pose any threats. This paper introduced the IEDXray dataset that can be used for the training in the detection of IED components.

The contributions of this paper are: (a) the presentation of a novel IEDXray dataset, which is composed of X-ray images of essential IED components. (b) evaluation using different CNN-based object detection models and feature extractors for the detection of IED components in IEDXray dataset. To the best knowledge of the authors, this is the first paper that uses CNN in IED components detection in X-ray images.

II. Materials and Method

A. Transfer Learning

Transfer learning is a technique of reusing a knowledge trained for a specific task (source domain) and apply it to another related or different task (target domain). This method is essential when the available data is limited like X-ray images. Transfer learning is commonly used for deep learning, but it can also be used in another context like reinforcement learning. Preliminary experiments were conducted in [13], [14] to examine the effect of transfer learning in object detection and classification, respectively. The researchers in [15] defined transfer learning using the following notations:

A domain (D) contains two components in (1), the feature space (X) and marginal probability distribution $P(X)$ where $X = \{x_1, \ldots, x_n\} \in X$.

$$D = \{X, P(X)\} \tag{1}$$

A task (T) contains two components in (2), the label space (Y) and objective function $f(\cdot)$

$$T = \{Y, f(\cdot)\} \tag{2}$$

Transfer learning aims to improve the learning of the target objective function $f_T(\cdot)$ in target domain D_T using knowledge in the source domain D_S and source learning task T_S, where T_T is a target learning task and $D_S \neq D_T$, or $T_S \neq T_T$.

Fig. 1 shows a simple transfer learning pipeline. The source domain (left) is a detection model trained on a large database like Microsoft Common Objects in Context (MS COCO) dataset [16]. The convolutional features from its early layer contain generic features (i.e., edges, shapes, and textures) that can be extracted and transferred to the target domain. On the right part of the figure, the target domain is the new detection model to detect IED components in an X-ray image. The convolutional features extracted from the source domain are frozen to avoid updating the weights. Then, the top layers of the target domain are trained with random weights.

B. Object Detection Models

1) Faster R-CNN: Faster R-CNN [17] is composed of two parts, the deep fully convolutional network and Fast R-CNN [18] detector. The task of the deep fully convolutional network is to propose regions while a Fast R-CNN detector makes use of the proposed regions. Faster R-CNN highlights the introduction of Region Proposal Network (RPN) solving the slow proposal computation in Fast R-CNN. The RPN accepts any size of an input image and outputs a set of rectangular object proposals with an objectness score.

2) Single Shot Multibox Detector (SSD): The SSD [19] is a single deep neural network object detector that uses multiple-scale feature maps and default boxes for detection. It eliminates bounding box proposals and feature resampling stage, as a result, increases the speed of detection compared with Faster R-CNN and YOLO [20]. This improvement became possible by using a small convolutional filter applied to feature maps to predict object categories, offsets in bounding box locations (default boxes) and separate filters for different aspect ratio detection. SSD allows the use of low-resolution images (300×300) and can achieve real-time processing speed.

3) Region-based Fully Convolutional Networks (R-FCN): The R-FCN [21] proposed a position-sensitive score maps by using a set of specialized convolutional layers to solve the translation invariance in image classification and translation variance in object detection. These score maps represent one relative position of one object class. R-FCN used Residual Networks (ResNets) that has 100 convolutional layers as a backbone but removes the average pooling layer and fully connected layer. The convolutional layers were used to compute feature maps. The model claims 2.5-20 times faster than Faster R-CNN and still maintained accurate results.

C. Feature Extractors

There are four feature extractors used to extract low-level features in the input image such as Inception-v2, ResNet, Inception-ResNet-v2, and MobileNet.

1) Inception-v2: Inception-v2 [22] is an upgrade from previous version of inception called GoogleNet (Inception-v1) [23]. In this version, batch normalization is introduced to prevent internal covariate shift that slows down the training due to low learning rates. Batch normalization can be done by normalizing the inputs in every layer before feeding to the activation function. Batch normalization allows the use of higher learning rates and thus, increases the training speed.

Furthermore, batch normalization acts as a regularizer similar to dropout that reduces overfitting.

2) ResNet: Although previous deep neural network architectures such as AlexNet [24] and VGGNet [25] perform well in an image recognition task, most of it cannot be trained deeper due to decreased in accuracy and increased computational cost. In [26], introduced a deep residual learning framework called residual networks (ResNet) that addresses the degradation problem in which as the depth of the network increases, the accuracy gets saturated and degrades rapidly. ResNet works by adding skip connections forming a residual block. Skip connections allow ResNet to train a deeper neural network (152 layers) without the loss in performance compared with a traditional network.

3) Inception-ResNet-v2: Inception-ResNet-v2 [27] is a hybrid model based on the inception module which uses residual connections (shortcut connection) from ResNet [26] that allows faster training of deep neural network and improves accuracy that roughly matches the computational cost of Inception-v4. This version is more accurate than Inception-ResNet-v1 in the ImageNet [28] dataset.

4) MobileNet: MobileNet [29] uses a depthwise separable convolution that enables a lightweight deep convolutional neural network and can be deployed in mobile applications. Depthwise separable convolution is a depthwise convolution followed by a pointwise convolution or simply 1 × 1 convolution. Two hyperparameters are introduced in MobileNet such as width multiplier and resolution multiplier to tune the model easily. This architecture is slightly less accurate than VGG-16 [23] but 32 times smaller in terms of the number of parameters.

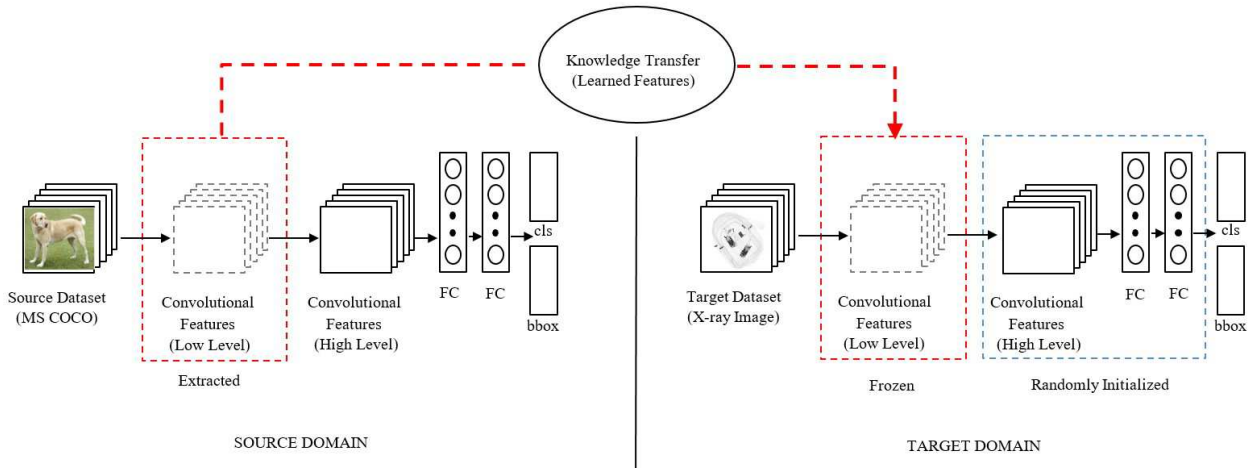


Fig. 1 Transfer learning pipeline

III. Results and Discussion

A. X-ray Image Acquisition

Philippine Bomb Data Center (PBDC) and Explosive Ordnance Disposal and Canine (EOD/K9) Group of the Philippine National Police (PNP) provided the IED replicas that were used in the experiment. The IED replicas were placed several times in a HI-SCAN 6040-2is dual-view X-ray machine [30] shown in Fig. 2. This X-ray machine can penetrate up to 35mm steel. The operator manually captures and saves the images of each IED replicas projected on the LCD screen. Also, to gather more X-ray images, there is a video recorder in front of the LCD screen that captures every projected X-ray images.

B. IEDXray Dataset

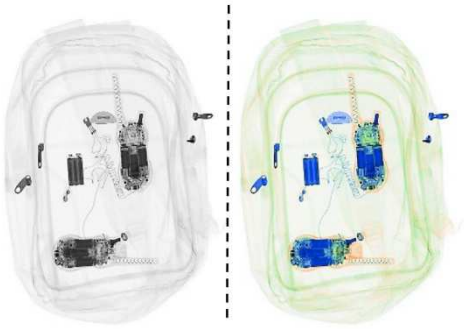
The IEDXray dataset is a collection of X-ray images with various IED replicas without explosive material. IED replicas have complete mechanisms of common IEDs, like a power source, wires, initiator, switch and container. Table 1 shows the number of images in the IEDXray dataset.

Fig. 2 HI-SCAN 6040-2is dual-view X-ray machine

TABLE I
Number of Images in IEDXray Dataset

Description	Number
training images	1209
test images	134
Total	**1343**

In colored images, the color varies depending on the type of material scanned in an X-ray machine. Orange represents organic substances; green represents indeterminate, while blue represents inorganic/metallic objects. Grayscale images were derived from colored images using grayscale image converter. Fig. 3 shows the sample IEDXray dataset in grayscale and colored images.



(a) (b)

Fig. 3 IEDXray dataset

C. Dataset Annotation and Training

After the image acquisition, dataset annotation was performed manually by drawing a bounding box or ground truth in each image and labeled according to its class. This paper focused on three IED components, such as the battery, mortar, and wires. Then, 90% of the data was used to train the IED detector, and 10% for testing to measure the detection performance of each model in an unseen set of images. This train-test split was chosen to maximize images used for training due to the small number of data. Table 2 shows the number of labels (ground truth) used in training and testing. Wires have the highest number of ground truths because of many segments of wire was labeled per image.

TABLE II
NUMBER OF LABELS IN IEDXRAY DATASET

Class	Training	Testing
battery	1159	158
mortar	529	29
wires	1872	192
Total	**3560**	**379**

During the training, data augmentation was performed to compensate for the lack of available datasets. All object detection models and feature extractors were trained and evaluated with the use of data augmentation and without data augmentation. Fig. 4 shows the system setup. The augmentation used were horizontal flip, vertical flip, image scale, image rotation, and image crop. In a horizontal flip, images are randomly flipped horizontally 50% of the time. While vertical flip randomly flipped images vertically 50% of the time. Image scale randomly enlarges or shrinks images and keeps the aspect ratio. The minimum scale ratio used was 0.7, and maximum scale ratio was 2.1. Image rotation randomly rotates the image and detections by 90 degrees counter-clockwise 50% of the time. Lastly, the image crop randomly crops the image. The object detection model and feature extractor combinations used in the experiments were based on the configuration described in [31].

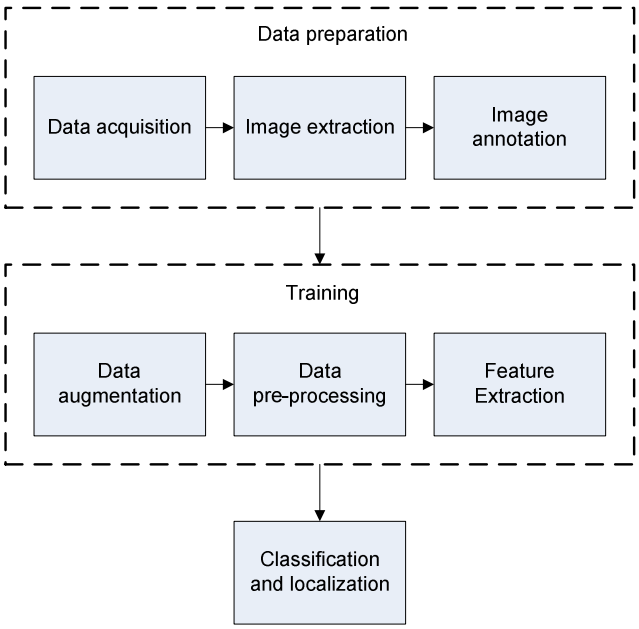


Fig. 4 System setup

D. Evaluation Metrics

The classification and detection performance was measured using the PASCAL VOC 2010 metric [32]. The average precision (AP) was computed by interpolation in all data points in the precision (Pr) × recall (Re) curve and by getting the area under the curve (AUC). The mean average precision (mAP) in (3) is the average AP of all classes (C).

$$mAP = \frac{1}{C}\sum_{i=1}^{C} AP_i \tag{3}$$

Pr and Re was calculated using (4) and (5), respectively. Pr shows the predictive power of the model or percentage of correct positive predictions while Re shows the hit rate (true positive rate) or percentage of true positives detected among all relevant ground truths.

$$Pr = \frac{TP}{TP + FP} \tag{4}$$

$$Re = \frac{TP}{TP + FN} \tag{5}$$

The detection is considered correct if the intersection over union (IoU) is greater than 0.5 (IoU > 0.50). The IoU in (6) was calculated by dividing the area A of intersection (ground truth X_G ∩ predicted X_p) to the area of union (ground truth X_G ∪ predicted X_p).

$$IoU = \frac{\text{area of overlap}}{\text{area of union}} = \frac{A(X_G \cap X_P)}{A(X_G \cup X_P)} \tag{6}$$

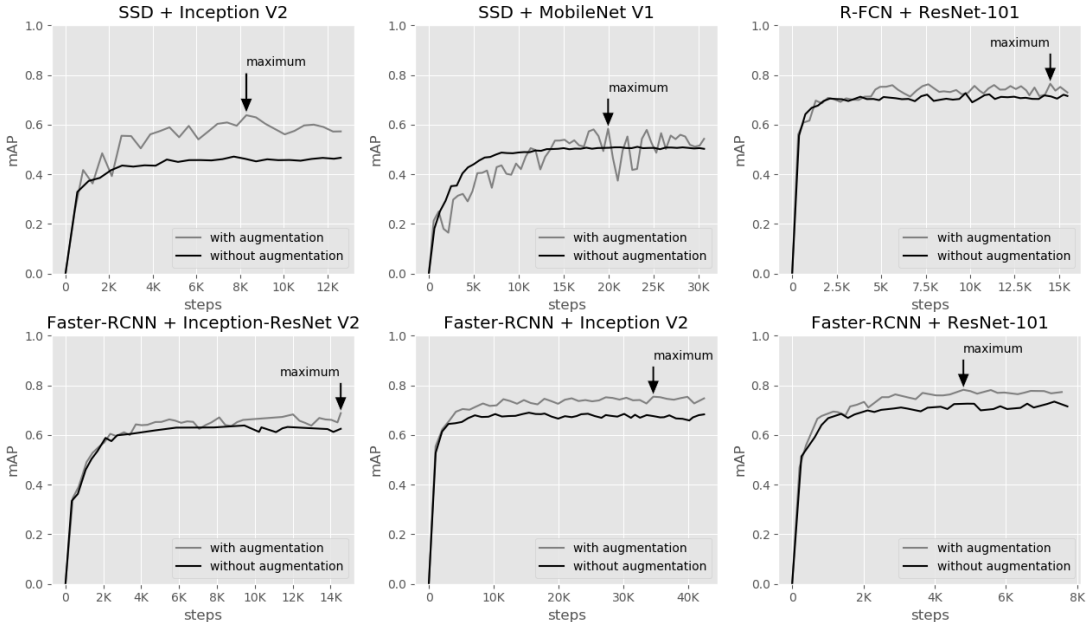


Fig. 5 mAP monitoring of different object detection models and feature extractor combinations

During the training, $mAP's$ of each object detection model and feature extractor combinations are monitored every five minutes until the value stops improving. Fig. 5 shows the evaluation progress. The black arrow indicates the maximum value of the mAP recorded in whole training. Some object detection models and feature extractors took a longer time to train than the others due to the complexity of the model. In all combinations, the maximum mAP recorded was 78.19% using Faster R-CNN + ResNet-101 with data augmentation.

Table 3 shows the detection performance in each object detection models without using data augmentation with different feature extractor combinations. The subscript (g) (i.e., $battery_g$) in the table means that the test data is a grayscale image while in another table (Table 5), subscript (c) means that the test data is a colored image. The mAP_g comparison was based on last saved checkpoint while mAP_{max} was based on the highest mAP recorded during training. In using grayscale test images, the highest mAP_g was recorded in RFCN + ResNet-101 with 71.57% while the lowest mAP_g was recorded using SSD + Inception V2 with 46.67%. The huge gap in the performance of Faster R-CNN, while Faster R-CNN and R-FCN are excellent when accurate detection is desired to identify objects.

On the other hand, Table 4 shows the detection performance using data augmentation. The highest mAP_g was recorded in Faster R-CNN + ResNet-101 with 77.29% while the lowest mAP_g was recorded using SSD + MobileNet V1 with 54.31%. The results indicate an increase (at least 8.04%) in detection performance after using data augmentation. Fig. 6 shows sample detection using Faster R-CNN + ResNet-101. The red bounding box indicates the ground truth. Fig. 6b shows false detection in the test image. The IED detector failed to detect the IED component as wire.

When it comes to the performance in each class, mortar has the highest AP which is 99.08% using R-FCN + ResNet101 (with and without data augmentation) while wire has the lowest which is 20.93% using SSD + MobileNet V1 (with data augmentation). Based on the AP results, it indicates that data augmentation can improve or worsen the individual detection performance depending on the model chosen.

TABLE III
DETECTION PERFORMANCE WITHOUT DATA AUGMENTATION

Object detection model	Feature extractor	mAP_g	mAP_{max}	$battery_g$	$mortar_g$	$wires_g$
SSD	Inception V2	0.4667	0.4715	0.3071	0.8588	0.2342
SSD	MobileNet V1	0.5027	0.5107	0.2952	0.9811	0.2318
R-FCN	ResNet-101	0.7157	0.7268	0.6624	0.9908	0.4939
Faster R-CNN	Inception-ResNet	0.6248	0.6381	0.5527	0.9766	0.3452
Faster R-CNN	Inception V2	0.6829	0.6897	0.6091	0.9885	0.4510
Faster R-CNN	ResNet-101	0.7151	0.7343	0.6730	0.9885	0.4836

R-FCN compared with SSD models is the tradeoff between accuracy and speed. SSD is ideal in real-time applications,

TABLE IV
DETECTION PERFORMANCE WITH DATA AUGMENTATION

Object detection model	Feature extractor	mAP_g	mAP_{max}	$battery_g$	$mortar_g$	$wires_g$
SSD	Inception V2	0.5729	0.6382	0.4434	0.9392	0.3360
SSD	MobileNet V1	0.5431	0.5837	0.4836	0.9363	0.2093
R-FCN	ResNet-101	0.7297	0.7649	0.6603	0.9908	0.5381
Faster R-CNN	Inception-ResNet	0.6880	0.6880	0.6326	0.9802	0.4512
Faster R-CNN	Inception V2	0.7474	0.7541	0.7077	0.9851	0.5493
Faster R-CNN	ResNet-101	0.7729	0.7819	0.7765	0.9897	0.5524

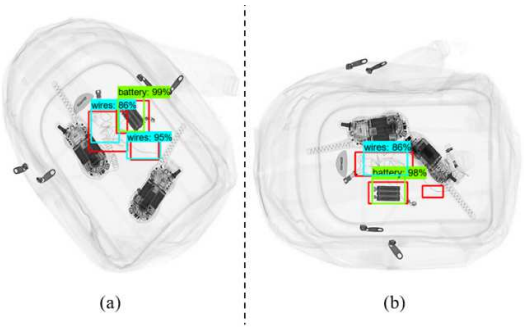


Fig. 6. Sample detection using Faster R-CNN + ResNet-101

Meanwhile, the mortar was accurately detected because its shape and size do not vary in the training data compared with wires that have different shape and sizes. Furthermore, the ground truth used for wires were not consistent. Fig. 7 shows the sample detections of all object detection models and feature extractors that were evaluated in the study.

In another experiment, colored images were evaluated using the trained grayscale IED detector with data augmentation to know the effect of color in detection performance.

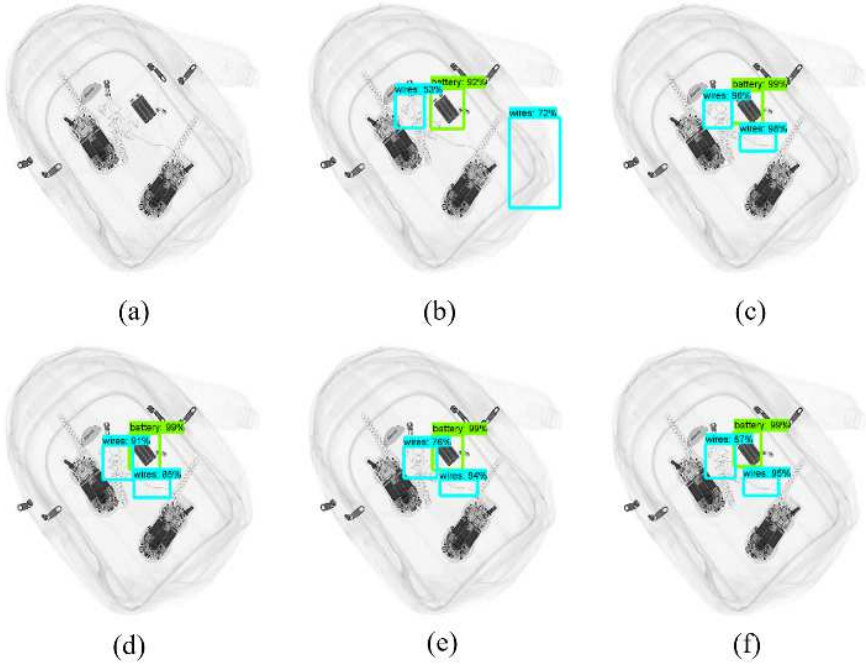


Fig. 7 Sample detections using data augmentation of (a) SSD + Inception V2, (b) SSD + MobileNet V1, (c) R-FCN + ResNet-101, (d) Faster R-CNN + Inception-ResNet V2, (e) Faster R-CNN + Inception V2, (f) Faster R-CNN + ResNet-101

Table 5 shows the performance of the grayscale IED detector in a colored test set. The results show that the *mAP* tend to decrease in all object detection and feature extractor combinations. The maximum decrease in *mAP* was recorded using SSD + MobileNet V1 (-12.49%). Therefore, the data trained on grayscale images should be tested only in grayscale images to maximize its performance because features that can be extracted in grayscale images are entirely different in colored images.

TABLE V
DETECTION PERFORMANCE WITH DATA AUGMENTATION
(COLORED TEST SET)

Object detection model	Feature extractor	$mAP_{(c)}$
SSD	Inception V2	0.5300
SSD	MobileNet V1	0.4753
R-FCN	ResNet-101	0.6935
Faster R-CNN	Inception-ResNet	0.6252
Faster R-CNN	Inception V2	0.7026
Faster R-CNN	ResNet-101	0.7316

IV. Conclusion

This paper introduced a novel IEDXray dataset, which is composed of X-ray images of IED replicas. The IEDXray dataset was used as training data for the detection of IED components in X-ray images. State-of-the-art CNN-based object detection models and feature extractor combinations were evaluated and compared its performance using the IEDXray dataset. Experiment results showed that Faster R-CNN + ResNet-101 achieves the highest value with 77.29% mAP in the test set using transfer learning with data augmentation. While in individual detection performance, R-FCN + ResNet-101 makes 99.08% AP in the test set. In another experiment using a colored test set, the detection performance decreased by a maximum of 12.49% using SSD + MobileNet V1.

For future works, an additional class of components and data will be included in order to train the IED components detector from scratch.

Acknowledgment

The authors would like to thank the Department of Science and Technology-Science Education Institute (DOST-SEI) and Bulacan State University for the support given in the completion of this research.

References

[1] National Consortium for the Study of Terrorism and Responses to Terrorism (START), "Global Terrorism Database [Data file]." 2018.

[2] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, pp. 886–893.

[3] R. A. Bedruz, E. Sybingco, A. Bandala, A. R. Quiros, A. C. Uy, and E. Dadios, "Real-time vehicle detection and tracking using a mean-shift based blob analysis and tracking approach," in *2017IEEE 9th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*, 2017, pp. 1–5.

[4] A. C. P. Uy *et al.*, "Automated vehicle class and color profiling system based on fuzzy logic," in *2017 5th International Conference on Information and Communication Technology (ICoIC7)*, 2017, pp. 1–6.

[5] 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, 2018.

[6] T. Franzel, U. Schmidt, and S. Roth, "Object detection in multi-view X-ray images," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2012, vol. 7476 LNCS, pp. 144–154.

[7] J. Redmon and A. Farhadi, "{YOLO9000:} Better, Faster, Stronger," *CoRR*, vol. abs/1612.0, 2016.

[8] M. Bacstan, "Multi-view object detection in dual-energy x-ray images," *Mach. Vis. Appl.*, vol. 26, no. 7–8, pp. 1045–1060, 2015.

[9] 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, 2017.

[10] V. Riffo and D. Mery, "Automated detection of threat objects using adapted implicit shape model," *IEEE Trans. Syst. Man, Cybern. Syst.*, vol. 46, no. 4, pp. 472–482, 2015.

[11] D. Mery *et al.*, "GDXray: The database of X-ray images for nondestructive testing," *J. Nondestruct. Eval.*, vol. 34, no. 4, p. 42, 2015.

[12] M. Xu, H. Zhang, and J. Yang, "Prohibited Item Detection in Airport X-Ray Security Images via Attention Mechanism Based CNN," in *Chinese Conference on Pattern Recognition and Computer Vision (PRCV)*, 2018, pp. 429–439.

[13] R. L. Galvez, A. A. Bandala, E. P. Dadios, R. R. P. Vicerra, and J. M. Z. Maningo, "Object detection using convolutional neural networks," in *TENCON 2018-2018 IEEE Region 10 Conference*, 2018, pp. 2023–2027.

[14] R. L. Galvez, E. P. Dadios, A. A. Bandala, and R. R. P. Vicerra, "Threat object classification in X-ray images using transfer learning," in *2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management, HNICEM 2018*, 2019.

[15] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, 2009.

[16] T.-Y. Lin *et al.*, "Microsoft coco: Common objects in context," in *European conference on computer vision*, 2014, pp. 740–755.

[17] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[18] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[19] W. Liu *et al.*, "Ssd: Single shot multibox detector," in *European conference on computer vision*, 2016, pp. 21–37.

[20] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[21] J. Dai, Y. Li, K. He, and J. Sun, "R-fcn: Object detection via region-based fully convolutional networks," in *Advances in neural information processing systems*, 2016, pp. 379–387.

[22] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv Prepr. arXiv1502.03167*, 2015.

[23] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[24] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[25] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv Prepr. arXiv1409.1556*, 2014.

[26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[27] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[28] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, 2009, pp. 248–255.

[29] A. G. Howard *et al.*, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv Prepr. arXiv1704.04861*, 2017.

[30] Smiths Detection, "HI-SCAN 6040-2is dual-view X-ray machine," 2019. [Online]. Available: https://www.smithsdetection.com/products/hi-scan-6040-2is-hr/.

[31] J. Huang *et al.*, "Speed/accuracy trade-offs for modern convolutional object detectors," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7310–7311.

[32] 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, 2010.