International Journal on Advanced Science Engineering Information Technology

Vol.13 (2023) No. 6
ISSN: 2088-5334

Telecommunication Fiber Box Detection Using YOLO in Urban Environment

Azib-Jazman Azmawi [a], Wan-Noorshahida Mohd-Isa [a,*], Abdul Aziz Abdul Rahman [b]

[a] *Faculty of Computing and Informatics, Multimedia University, Persiaran Multimedia, Cyberjaya, 63100, Malaysia*
[b] *Telekom Research & Development Sdn. Bhd., TM Innovation Center, Cyberjaya, 63100, Malaysia*
*Corresponding author: *wan.noorshahida.isa@mmu.edu.my*

***Abstract*— The Fiber Distribution Panel (FDP) box is an essential piece of internet access hardware because it provides users with high-speed data networking and functions as a cable organizer to reduce wire clutter. After installing the FDP, an inspection must be performed to ensure that all necessary components are present. However, This examination is still done manually; the technician snaps a picture of the panel and sends it to its supervisor for verification, which is time-consuming and often prone to errors. In addition to images captured in low-light and complex environments, it makes it more difficult for humans to identify the components with just a naked eye. On this matter, a much more efficient method to assess the FDP installation work is very much needed. Therefore, using computer vision approaches, we utilize a deep learning algorithm to perform object detection and automate the assessment of FDP installation components based on visual data. One of the deep learning models established in the literature is the You Only Look Once (YOLO) model, a one-stage deep learning object detection algorithm that employs a fully conventional approach to generate highly accurate real-time predictions. This paper uses YOLOv5s to identify the fiber box and its relevant components, even in urban environments. Experimentations show that YOLO successfully identified the installation parts with a mean average precision score of 86% at a 0.5 confidence level, even with limited data.**

***Keywords*—Fiber distribution panel; computer vision; deep learning; optics network; complex background.**

Manuscript received 7 Jan. 2023; revised 17 May 2023; accepted 28 Oct. 2023. Date of publication 31 Dec. 2023.

I. INTRODUCTION

The telecommunication sector is one of the fastest-growing industries due to its high demands of providing wireless communication and acting as Internet Service Providers (ISPs) for the public [1]. Amidst the recent pandemic, there was a surge of demand for constantly available internet connection as more people work from home and rely on video conferencing to hold meetings, leading to upticks in revenues for the telecom industries despite the temporary closure of businesses [2]. According to the Department of Statistics Malaysia (DOSM), 96% of Malaysian households had internet access in 2022, an increase of 1.1% from the previous year [3]. With more people relying on technology for information, the industry must be more vigilant and proactive to meet people's demands [4]. This requires its employees to set up the necessary equipment for network connectivity at the speed of light. One of the many important pieces of hardware in internet installation is the Fiber Distribution Panel (FDP) box.

FDP boxes are high-density fiber distribution boxes that serve as direct splicing, branching, straight-through, splitting, and fiber termination devices while shielding them from environmental risks such as moisture or ultraviolet light (i.e., UV rays). These panels contain fiber optic cables and ports that provide high-speed data networking to their end-users and are usually mounted either on a utility pole for outside installation or on walls for inside installation. A typical fiber box can provide internet services to up to eight households but can be extended by installing additional ports. With the number of fiber cables brought into the network increasing exponentially every day, the complexity of the fiber optic cabling system also increases. Therefore, network technicians frequently utilize cable organizers or fiber panels to reduce wire clutter, particularly when dealing with fiber cable transfers, additions, and alterations. This allows them to manage huge fiber lines and connection points within a standard panel or container. As a result, cable management and maintenance were simplified and became more convenient for the contractors. Fig. 1 shows a sample of an indoor FDP box installed in urban areas.

Fig. 1 A sample of indoor Fiber Distribution Panel box installation. Some parts are blurred out due to confidentiality concerns.

A newly installed FDP must then be evaluated for the quality of fiber optic networks. Network specialists must verify that the components are present and in the correct arrangement following the company's blueprint. An old installation usually comprises an FDP box, a trunk, a copper box, and a fiber box, while recent network setups only include the FDP and the trunk. Before the epidemic sweeps the world, two people install an FDP panel: a network technician and a supervisor. The technician will set up the fiber box at the designated pole or building, and the inspector will inspect and verify that the installation is done properly, following the guidelines on the spot. However, since the strict implementation of workplace Standard Operating Procedure (SOP) due to the COVID-19 pandemic, the movement of employees from telecommunications firms has been restricted, allowing just one worker to be on site for FDP installation. Since then, only the wiremen go on-site and place the fiber panel, photograph it, and send it to their supervisor via e-mail or messaging application for review and approval. This makes the simple task of checking the FDP box overly complicated.

An emerging challenge would be communication between the two different parties. Late replies or approval from the supervisor make it difficult for contractors to know whether the installation is done according to its standards in a timely manner. By the time the supervisor replied, the technician might have left the site. If they disapprove, it may take some time for the network specialist to come back to rearrange the FDP box since they have their own schedule to work with. In cases where the contractor may have forgotten to update or take a picture after the installation, the review approval from the supervisor will be delayed. The current system is still being performed manually, so human errors can negatively affect the component's durability. An inspector or contractor can accidentally overlook missing components in the FDP setup, which does not comply with the company's requirements.

Moreover, images captured in low-light or occluded environments make it difficult for humans to identify the objects with just a naked eye. When done carelessly, the verification process can jeopardize data integrity, resulting in underperformance. This proves the current inspection and verification method is inefficient, costly, and resource-consuming process.

Frequently, these engineering requirements were overlooked by humans during the installation. Thus, this paper aims to investigate possible methods for detecting the FDP box by using computer vision approaches to assist in overcoming the challenges above. In this work, we consider utilizing a deep neural network to detect and recognize the components based on visual data. With computer vision, we hope to overcome these human errors and provide quality data integrity in telecommunication while minimizing the overall loss of value caused by external factors.

A. Literature Review

Current object detection algorithms can be separated into two groups: models that predict object location and classification separately (two-stage), like Faster Region-based Convolutional Neural Network (Faster R-CNN), and models that perform the localization and classification at the same time (one-stage), like You Only Look Once (YOLO) [5]. The two-stage algorithms identify the expected location of objects, called regions, using a region proposal method and detect objects in those regions with a Convolution Neural Network (CNN), resulting in a more precise approach [6]. In contrast, one-stage detectors use a fully convolutional method to predict the bounding boxes and objects without proposing regions. Thus, it is a much faster approach as compared to two-stage detectors. In this section, we describe the development of the YOLO models and literature on object detection for assessment purposes.

B. YOLO

You Only Look Once (YOLO) is one of the famous object detection algorithms due to its accurate and real-time prediction of classes and bounding boxes on images [7]. Introduced by Redmon et al. in 2015, the original YOLO architecture has 24 convolutional layers and two fully connected layers at the end of the model with a Leaky Rectified Linear Unit (LReLU) activation function across all convolutional and dense layers except for the last layer which has a linear activation function. The method divides the input image into grids of cells where each cell predicts the bounding box coordinates of the object as well as the likelihood of the object belonging to different classes. YOLO uses some predefined class score threshold and non-max suppression to discard less relevant as well as overlapping bounding boxes of the same object [8]. The first development of YOLO was able to outdo state-of-the-art object detection models in terms of its performance and F1 scores across different datasets [9].

Over the years, multiple versions of YOLO have been released. In YOLOv2, the creators used a Darknet-19 backbone framework (19 convolution and five max-pooling layers), replaced dense layers with fully convolutional architecture, and introduced batch normalization to improve convergence while preventing overfitting [10]. The backbone was once again changed in the third version to Darknet-53, which replaces all max-pooling layers with stride convolutions consisting of 53 convolutional layers. An essential feature in YOLOv3 is the ability to perform multi-scale predictions, thus improving the detection of small objects. In 2020, YOLOv4 was released with the introduction of mosaic data augmentation along with a new classification and loss function. YOLOv5, developed in PyTorch instead of Darknet, was announced a couple of months later with improved performance and was more lightweight compared to previous variants [11]. At the end of 2022, YOLOv6 was implemented using the EfficientRep backbone, which uses higher parallelism and introduces a quantization scheme using RepOtimizer to achieve faster detection. The launch of YOLOv7 surpassed all known object detectors in terms of performance, speed, and accuracy, as it uses an extended efficient layer aggregation network (E-ELAN) to enhance the model learning without destroying its gradient path. The latest YOLO version 8 was released in January 2023 with a smaller number of box predictions and a faster NMS process [12]. However, its architecture is still under exploration, as no official papers were released at the time this paper was written.

After much deliberation, this project decided to utilize the YOLOv5 variant as it has been shown to give better precision and speed performance compared to the other versions [13]. Moreover, due to hardware limitations, we will apply YOLOv5 as it is faster when a system with normal GPU trains and tests the object detector, especially on a custom dataset. Ultralytics released four structural models of YOLOv5: YOLOv5s, YOLOv5m, YOLOv5l and YOLOv5x. The model size ranges from small to large, and the detection accuracy ranges from low to high [14]. When testing on the public COCO dataset, YOLOv5s has a faster detection speed, and YOLOv5x has a higher detection accuracy as compared to YOLOv3 and YOLOv4 [15]. Therefore, we will be implementing YOLOv5s in this project.

C. Safety Assessment Using Computer Vision

Automatic safety assessment based on object detection in machine vision has garnered increasing scientific interest. Several researchers have developed systems to digitalize safety control duties for a substantially higher quality of a product and safety assurance of its workers [16], [17]. At the time this study was conducted, there were no published articles that identified the telecommunication fiber box components using deep learning. Furthermore, most systems are highly implemented in the construction sector and not from a telecommunication firm's point of view. However, we can still get insights from their methods as past papers have used object detection to improve manual safety assessment tasks.

An early study done by [18] curated real-time guardrail detection by using transfer learning knowledge and Convolutional Neural Networks (CNN) to detect the presence of guardrails in construction sites. In need of a large data set, they developed a 3D model of a metal guardrail according to a real guardrail dimension and added noise and random colors to simulate different guardrail specifications. The synthetic guardrail was put on images at different camera heights, distances, and angles to cover different viewpoints in the dataset. The authors utilized VGG-16 as its base model while integrating the last few layers with a Multilayer Perceptron (MLP) or SVM. When there is only a single guardrail in the image, MLP returned a high accuracy score of 97% as compared to SVM, which only achieves 78%. Although the MLP and SVM model's accuracy dropped when multiple guardrails were present, they still have a decent detection accuracy of 86% and 72%, respectively. The authors emphasize that the models faced difficulties finding the guardrail in poor lighting conditions and when other objects that look structurally similar to a guardrail exist in the image.

A study in [19] covers a similar problem in the construction industry but towards detecting the safety harness to reduce fall accidents and enhance worker safety. The Region Proposal Network in Faster R-CNN was set with three distinct scales, aspect ratios, and nine anchors at each location to detect human bodies of varying sizes. Then, the box coordinates of the detected worker outputted by Faster R-CNN were used as input for their five convolutional, three fully connected, and one SoftMax layer CNN, to identify whether or not the construction worker is wearing the harness. At a 0.8 threshold, their human detection model receives a precision of 99% and recall of 95%, while their safety harness detection model achieves 80% and 98%, respectively. Their approach was able to overcome differing perspectives and illumination in images, as almost all workers were able to be detected correctly. However, CNN faces a slight issue with detecting the harness due to the presence of shadows and identical colors between the worker's clothes and their harness. This is also due to their limited data and small sample size for training, especially in the cases above, making it a challenge for their model to recognize these safety harnesses.

To conclude, research on using object detection to automate assessment tasks has been made in recent years to ensure safety equipment is present during construction work [20]. In this paper, we proposed a method for FDP box detection to assist inspectors in identifying the components in complex backgrounds. With deep learning models, we can quickly identify the telecom fiber box installation parts to free it from potentially hazardous materials that could physically damage the components, resulting in network underperformance. By integrating object detection technology, the condition of the FDP, along with the other components, can be improved and safeguard the overall reliability of the network infrastructure. Moreover, the papers above focused on the construction sector, not the telecommunication sector. Studies on implementing object detection for FDP installation in telecom industries still remain very scarce. This shows a significant study gap on this matter, highlighting the need for this project.

II. MATERIAL AND METHOD

The method of assessing FDP installation using computer vision requires several steps that must be taken prior to completing the required task. Our proposed framework for automated assessment on FDP installations using deep learning is shown in Fig 2. The research starts with gathering our data by capturing images of FDP boxes around urban areas. After the images are annotated and processed, the data will be used to train the YOLO model for FDP box recognition, and the results will be evaluated using the specified metrics.

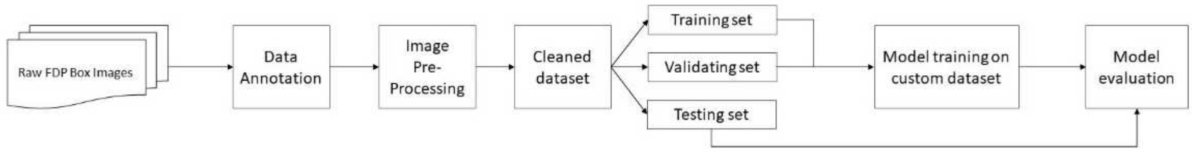

Fig. 2 The process framework of our FDP installation object detection using deep learning.

A. Data Collection

In many computer vision problems, a comprehensive dataset on the object of interest is needed to perform object recognition successfully. Due to works on deep learning-based object recognition for FDP panels being scarce, there is no known existing image database of FDP box images with their associated components at the moment this research was conducted. Therefore, data collection on these images is much needed in this study to train the models.

A dataset containing 370 images was captured from 104 unique FDP boxes, which consist of four classes, including FDP box, trunk, fiber box, and copper box. As FDP installations have indoor and outdoor arrangements, we decided to put our focus on indoor or wall-mounted FDP boxes that can be constantly seen in urban areas. Images were captured using a smartphone camera at approximately 50 cm apart from the FDP box with a height of 170 cm and a 90-degree angle throughout the data collection. All images have a dimension of 2252 by 4000 pixels, and each FDP box was captured from three different perspectives (left-view, front-view, and right-view) to increase the dataset's robustness. This will then generalize our system to identify the FDP box and its components even from multiple angles.

B. Data Annotation

The next step was to label every single picture in our dataset before incorporating them with the YOLO model. Data labeling is one of the computer vision pipeline's most important and exhaustive parts. Poor annotation can often lead to poor model performance. Therefore, we consult an expert in the field to assist us in identifying the different variations of FDP as well as the common components associated with the FDP box. Important pieces when it comes to FDP installation are the FDP box, green trunking, copper box, and fiber box. Fig 3 depicts a sample of the aforementioned components. Then, we annotate the box and its related components using an open-source image annotation tool to draw rectangle boxes on our object of interest. We chose labeling as it allows us to save the classes in the form of XML files in YOLO format, making it easier for our model to integrate, and no external conversion is needed.

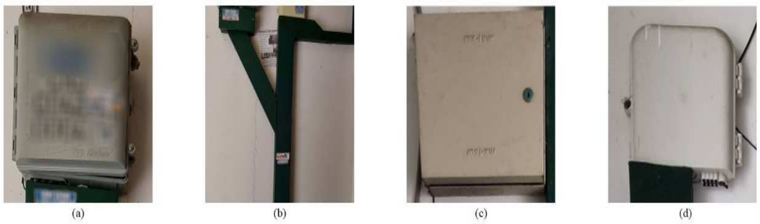

Fig. 3 Sample illustration of (a) FDP box, (b) green trunking, (c) copper box and (d) fiber box. Some parts are blurred out due to confidentiality concerns.

The images are divided into training, validating, and testing sets with an 80:10:10, ratio, respectively. The training dataset was used for training the deep learning model, while the validation dataset was used to evaluate our model. Finally, the model was tested on a completely new set of images from the testing dataset.

C. Image Pre-processing

Image pre-processing was done for formatting the images to ensure the quality and uniformity of images before feeding them to the models. It plays a crucial part in future procedures as it may decrease model training time and increase overall detection accuracy [21], [22]. Many model architectures, including the ones used in this research, have specific input sizes. For example, the YOLOv5 network only accepts images in the size of 640 by 640 pixels. However, over-resizing often makes images lose certain information in the process [23]. Although larger image size leads to better results, it may take longer to train the models as they contain a high amount of information. Therefore, it is important to make sure the input image should be altered carefully without affecting the quality of the images.

D. YOLOv5

The YOLOv5 object detector is the first YOLO variant that was implemented using PyTorch. The structure of YOLOv5 comprises four main components: the input, backbone, neck, and head layer, as shown in Fig 4. The input network of YOLOv5 includes mosaic data enhancement, auto image resizing, and an adaptive anchor box calculation process. Mosaic data enhancement was a feature introduced in YOLOv4 that enriches the background and small objects by combining four training images to improve the detection of small objects. Adaptive image scaling adds the least number of black borders to the original image with different widths and heights to have a uniform standard size before inputting it into the model. Once more information is filled in, as in the image is added with black borders, the inference speed increases. The adaptive anchor box calculation compares the output predicted boxes with the ground truth boxes based on the initial anchor boxes, calculates the gap, and updates it in reverse. This process is iterated continuously until parameters obtain the most suitable anchor box value [24].

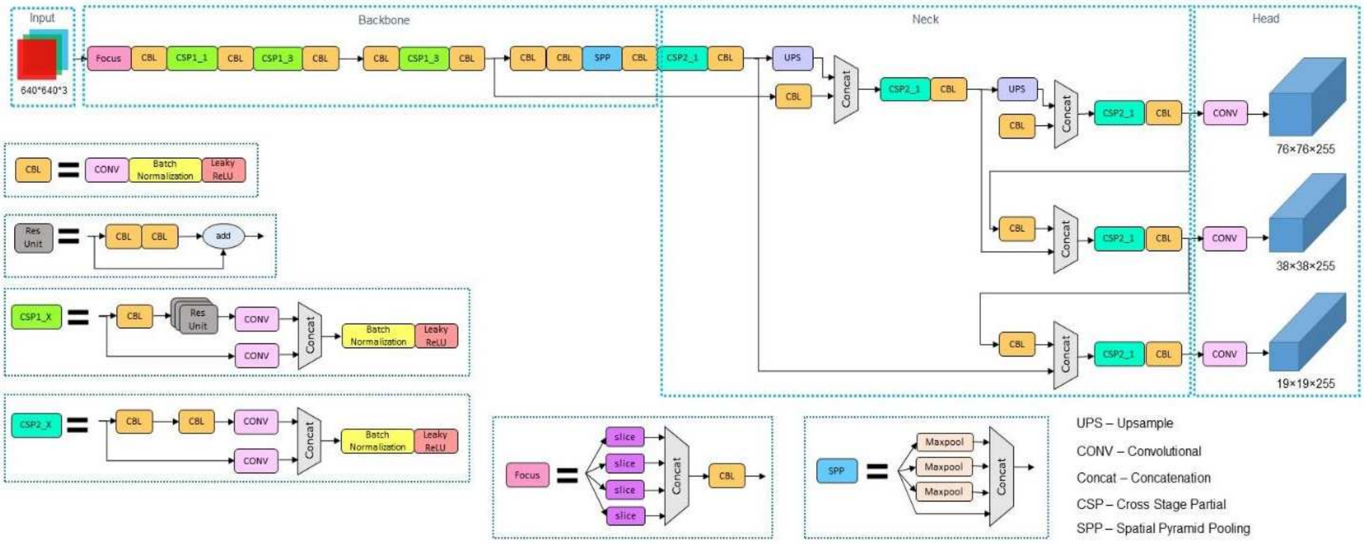

Fig. 4 Our proposed YOLOv5s architecture.

The backbone comprises a focus element, a few convolution modules, Cross Stage Partial (CSP) structures, and Spatial Pyramid Pooling (SPP). The focus structure splits the input image in each R, G, and B channel into four slices, where each slice is half the size of the input size. The four slices of each channel will then be concatenated, resulting in a feature map with 12 channels. After passing through one convolution operation, the final feature map contains 32 channels without losing any information. The input will then go through a Convolutional, Batch, and LeakyReLU (CBL) algorithm, which performs two-dimensional convolution, Batch Normalization (BN) regularization, and Sigmoid Linear Unit (SiLU) activation function. The main component in the backbone is the CSP module. The idea of CSP was adapted from YOLOv4 and was incorporated with Darknet in YOLOv5, called CSPDarknet, for feature extraction. The use of CSP, denoted as CSP1_X, was inspired by CSPNet [25], which brings a boost to the backbone by solving the repeated gradient information problem and increases the learning capability of CNN while reducing computational and model size [26][27]. The input of CSP goes through two convolutional layers and combines with the original value to perform residual feature transfer without increasing the output depth. Spatial Pyramid Pooling (SPP) fuses the features from max-pooling layers with 5×5, 9×9, and 13×13 kernel sizes to improve feature extraction abilities.

The neck of YOLOv5 utilizes a joint of Path Aggregation Network (PAN) [28] and Feature Pyramid Network (FPN) to boost the feature fusion capability of the network [29]. PAN improves the information flow within the model by conveying strong localization features from lower feature maps into higher feature maps. In contrast, FPN strengthens the object localization features using a top-down approach. The CSP structure, CSP2_X, is also applied for feature fusion enhancement in the neck network. Thus, further increasing the location accuracy of the object.

Lastly, the head of the network is made up of the YOLO layer where the three generated feature maps of various sizes pass through three detection layers for multi-scale object detection and prediction. Each layer outputs a corresponding vector containing the objectness score, an object's class probability, and its bounding box position. Finally, the object's predicted bounding box and category will be generated on the original image.

The loss functions used in YOLOv5 include localization, classification, and confidence loss. The localization loss function calculates how far the predicted bounding box is to an object's ground truth bounding box so that the parameters can be automatically adjusted and correct the position of the prediction frame to be as close as possible to the real value. This measure can be done using the concept of Intersection-Over-Union (IoU). However, using IoU itself as a loss function has its disadvantages. When no intersection exists between the predicted and the real bounding boxes, as in IoU is equal to 0, it prevents the model from calculating the gradient; hence, parameters cannot be optimized, making learning and training impossible to perform [27]. To overcome this issue, they introduce the idea of Generalized IoU (GIoU) [30]. IoU and GIoU are calculated as shown in Equations (1) and (2), respectively. A and B represent the central points of the predicted and ground truth bounding boxes, respectively, while C is the area of the smallest box that can encapsulate both the predicted and ground truth boxes. IoU can then be described as the intersection area of the predicted and ground truth box divided by its area of union. In that sense, YOLOv5, by default, uses the GIoU loss function, which can be described in Equation (3).

Confidence loss and classification loss are calculated during the training phase by using Binary Cross-Entropy (BCE) function, as it reduces computational complexity. While confidence loss measures the confidence that the detected object is an actual object, classification loss measures the probability of an object belonging to a class. A score value near 1 indicates that the model performs better in determining the FDP box and other items with higher confidence.

$$IoU = \frac{|A \cap B|}{|A \cup B|} \tag{1}$$

$$GIoU = IoU - \frac{|C-(A \cup B)|}{|C|} \tag{2}$$

$$L\_GIoU = 1 - GIoU \tag{3}$$

III. Results and Discussion

To validate the performance of the proposed approach, precision, recall, precision-recall curve (PR curve), and mean average precision (mAP) are used as measurement indicators [31]. Precision is the ratio of correctly classified positive samples to the total number of positively classified samples, while recall is the ratio of the correctly classified positive samples to all positive samples. Precision and recall are defined in Equation (4) and (5), respectively.

$$Precision\ (P) = \frac{TP}{TP+FP} \tag{4}$$

$$Recall\ (R) = \frac{TP}{TP+FN} \tag{5}$$

where TP is the number of correctly detected objects, FP is the number of incorrectly detected objects, FN is the number of undetected objects the model misses, and TN is the number of correctly undetected objects.

The PR curve plots the recall value on the horizontal axis and the precision value on the vertical axis. PR curve tells whether the model can detect all the objects in a single image correctly. The mean average precision is the average of all average precisions across all classes. YOLO calculates two mAP types: mAP@0.5 and mAP@0.5:0.95 defined in Equation (6) and (7), respectively.

$$Average\ Precision\ (AP) = \int_0^1 P(R)d(R) \tag{6}$$

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{7}$$

where P(R) is the precision value at a specified recall, d(R) is the difference between the current and next recalls while N is the total number of classes in the dataset. The mAP@0.5 calculates the average of average precision (AP) of all classes when the IoU threshold is 0.5, and the mAP@0.5:0.95 is the average mAP at numerous thresholds from 0.5, slowly incrementing by 0.05 until it reaches 0.95.

Three experiments were conducted in this research. In every experiment, a different dataset size was used, i.e., the total number of images in the experiment was different. Therefore, the number of images in our training, validating, and testing sets varies for every experiment while still maintaining the 80:10:10 ratio. The results of our experiment were saved, and the training time was recorded in Table 1.

TABLE I
EVALUATION METRICS FOR OUR EXPERIMENTS

Experiment	1	2	3
Dataset size	74	270	370
Precision (%)	50.33	77.85	**84.89**
Recall (%)	64.61	84.34	**87.29**
mAP@0.5 (%)	54.82	81.11	**86.12**
mAP@0.5:0.95 (%)	27.39	55.42	**59.72**
Total training time (mins)	30	105	140

*bold indicates the highest score achieved for each measure

Fig 5 depicts the performance metrics of YOLOv5s across every experiment. Experiment 3 retrieves the highest scores on all measures, followed by Experiments 2 and 1. Our initial test with 74 images shows that our model performed admirably, with 50.33% precision and 64.61% recall. Experiment 2 improves the model's precision by nearly 30% and recall by 20%. The values further increased during Experiment 3 where precision and recall obtained scores of 84.89% and 87.29%, respectively.

Furthermore, in each of our experiments, the recall value was always greater than the precision. This indicates that the model produces more detections but is not always classified correctly. The mAP@0.5 was likewise elevated. The model retrieves 54.82% in our first experiment, rising to 81.11% and 86.12% in Experiments 2 and 3, respectively. When analyzing its mAP values at different thresholds, its peak was in Experiment 3 (59.72%), and its lowest was in Experiment 1 (27.39%). This means that the model is able to detect objects with a precision of at least 59% and a recall of at least 50% across a range of IoU thresholds. We also observe that the total training time increases as more data is fed into training the model. Experiment 3 took almost five times longer to complete the YOLOv5s training than Experiment 1 on a custom dataset. We discovered that for every 74 images in the dataset, model training will take approximately 30 minutes to complete.

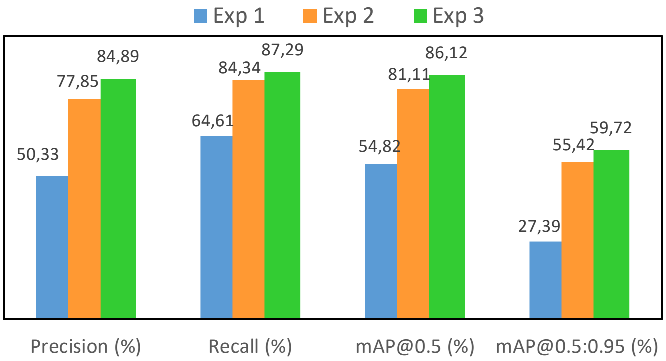

Fig. 5 Performance of YOLOv5s model in each experiment.

Fig 6 illustrates the precision-confidence graph for each component in every experiment. Experiment 1 shows that the precision steadily increases for the FDP class but fluctuates for the green trunk and copper box. For Experiments 2 and 3, it can be seen that the model could detect the green trunking much more accurately than the other parts. This is possibly due to the trunking being the biggest item among all the others in terms of space area, making it easier for our model to recognize the trunk. When the precision is at 1.00 for all classes, the confidence value increases. With 74 images, all classes only attain a precision of 1.00 at 85% confidence, but this rises to 91% with a larger dataset. Then, the confidence slightly increased by 2% when a dataset with additional photos was used.

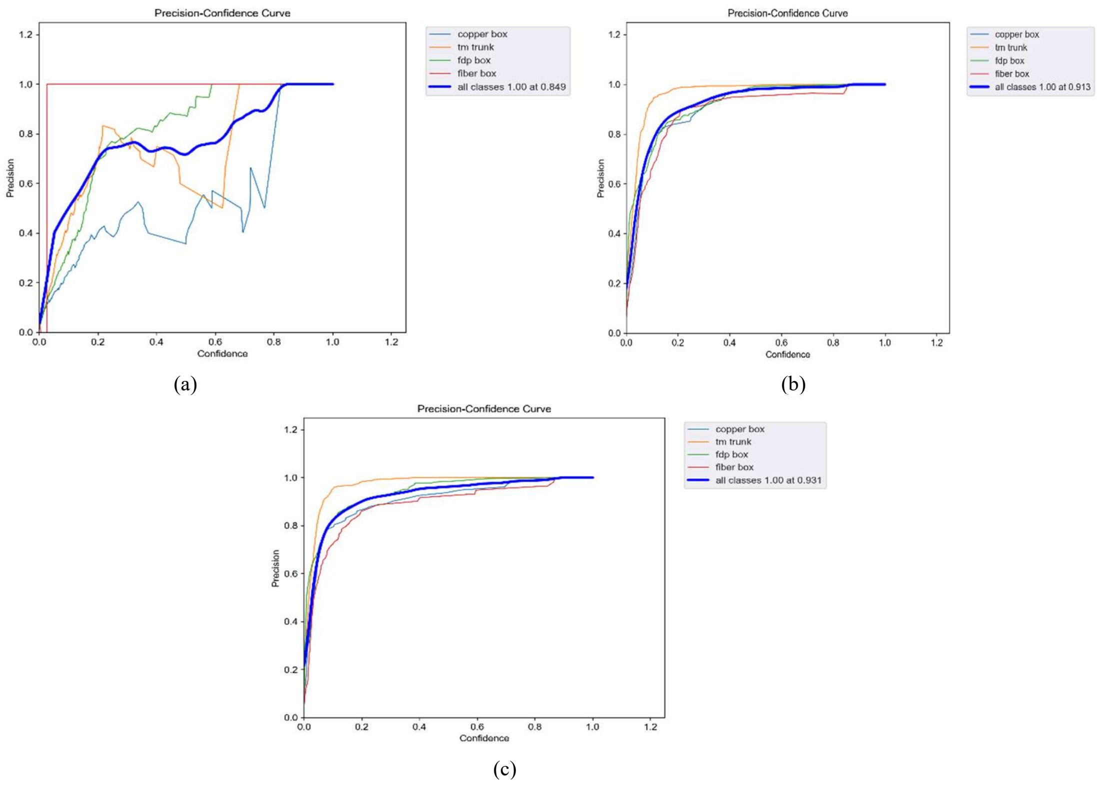

Fig. 6 Precision-Confidence Curve for (a) Experiment 1, (b) Experiment 2, and (c) Experiment 3.

Fig 7 represents the PR curve of YOLOv5s on the three tests. Experiment 3 was seen to be the best performance test from the graph as its curve is at the top-right corner of the graph and has a larger area under the PR curve for all classes, meaning both the precision and recall are maximized. The results also show that as more images are used in the training process, all the classes' mAP@0.5 increases with 0.446, 0.994, and 0.995 for each experiment, respectively.

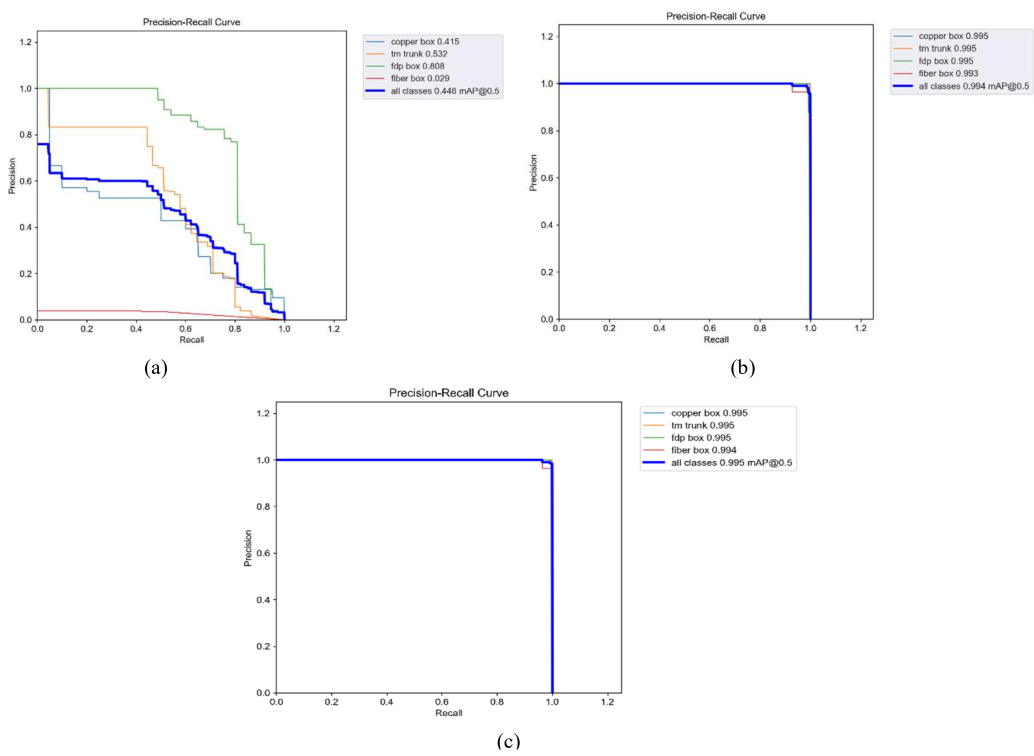

Fig. 7 PR Curve for (a) Experiment 1, (b) Experiment 2 and (c) Experiment 3.

Fig 8 shows a sample of object detection results after the YOLOv5s were trained on a custom dataset. The model for Experiment 1 was able to detect the components, especially in the FDP boxes with 77% confidence, even with less than 100 images. Copper boxes and trunks were also detected with less than 50% confidence.

(a) (b) (c)

Fig. 8 Sample result of object detection using our proposed method for (a) Experiment 1, (b) Experiment 2 and (c) Experiment 3

There are also instances where the detected objects were incorrectly classified. For example, in Fig 8(a), the copper box was identified at 41% confidence but also as an FDP box with a higher confidence of 65%. The model in Experiment 1 also produces multiple predictions and duplicate bounding boxes on a single component. The results of detection improve in Experiment 2 with only a single prediction on one object. Although the copper box was still wrongly classified as an FDP box by the model with 69% confidence, the model's ability to identify the trunk improved with 94% confidence. When trying out with a bigger dataset, YOLOv5s was finally able to correctly identify the copper box with a whopping 91% confidence while maintaining high accuracy for the trunk and FDP boxes with 95% and 92%, respectively.

Overall, YOLOv5s achieved satisfactory metrics scores. It receives more than 80% in precision, recall, and mAP scores in Experiment 3. The difference between precision and recall was high at first, but it became smaller as a larger dataset was used in our experiments. In the case of mAP@0.5:0.95 results, it receives quite low scores compared to mAP at a 0.5 threshold. However, it was still able to achieve 59% in Experiment 3, which is still considerably high given the small number of images we had. Cases of false positives like those in Fig 8(a) suggest that the model has some difficulties distinguishing between copper and FDP boxes. We believe the lack of data in our training set mainly causes this. During our data collection, we discovered there had been a variety of copper and FDP types. Therefore, there could be an imbalance in our dataset if we were to consider the different builds of each component.

Moreover, both boxes have similar colors, so our model struggles to differentiate between them. One way to address this issue is to increase the samples of each copper or fiber box variation to enhance the learning ability of YOLOv5s. As observed in Experiment 3, with a bigger dataset, the YOLO algorithm better detects and predicts objects without duplicate bounding boxes with much higher confidence.

IV. Conclusion

This research implemented computer vision approaches to detect the telecommunication fiber box (FDP box) to ensure the presence of crucial components during the installation and inspection tasks. A new dataset of FDP components in urban areas was created and annotated into four classes of essential parts. Results of YOLOv5s demonstrate the efficacy in detecting the FDP installation components amidst occlusions and type variation. From our observation, YOLOv5s significantly improve traditional inspection methods, enhancing the speed and safety of telecommunication networks. The findings highlight the potential of object detector models as a valuable tool in the telecom industry, enabling improved work efficiency while maintaining network integrity and reliability. Moreover, even with less than 500 images, the model could correctly identify every component with high confidence. To our knowledge, this is also the very first Fiber Distribution Panel dataset that existed. This provides a solid foundation for future research to improve the detection rate of our model for better component distinguishability.

Acknowledgment

We are grateful for the support given by Multimedia University and TM R&D Malaysia with grant ID. MMUE/230001.

References

[1] R. Wickramasinghe and K. A. Razak, "The Impact Of The Telecommunication Industry As A Moderator on Poverty Alleviation

and Educational Programmes To Achieve Sustainable Development Goals In Developing Countries," Journal of Informatics and Web Engineering, vol. 2, no. 1, pp. 25–37, Mar. 2023, doi: 10.33093/jiwe.2023.2.1.3.

[2] Md. M. Alam, H. Wei, and A. N. M. Wahid, "COVID -19 outbreak and sectoral performance of the Australian stock market: An event study analysis," Australian Economic Papers, vol. 60, no. 3, pp. 482–495, Nov. 2020, doi: 10.1111/1467-8454.12215.

[3] Department Of Statistics Malaysia (DOSM), "Laporan Survei Penggunaan dan Capaian ICT oleh Individu dan Isi Rumah 2022 ICT Use and Access by Individuals and Households Survey Report." Accessed: June. 02, 2023. [Online]. Available: https://www.dosm.gov.my/uploads/release-content/file_20230531113305.pdf.

[4] J. Sahni, "Role of Quality of Work Life in Determining Employee Engagement and Organizational Commitment in Telecom Industry," International Journal for Quality Research, vol. 13, no. 2, pp. 285–300, Jun. 2019, doi: 10.24874/ijqr13.02-03.

[5] S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, and B. Lee, "A survey of modern deep learning based object detection models," Digital Signal Processing, vol. 126, p. 103514, Jun. 2022, doi: 10.1016/j.dsp.2022.103514.

[6] L. Du, R. Zhang, and X. Wang, "Overview of two-stage object detection algorithms," Journal of Physics: Conference Series, vol. 1544, p. 012033, May 2020, doi: 10.1088/1742-6596/1544/1/012033.

[7] F. Sultana, A. Sufian, and P. Dutta, "A Review of Object Detection Models Based on Convolutional Neural Network," Advances in Intelligent Systems and Computing, pp. 1–16, 2020, doi: 10.1007/978-981-15-4288-6_1.

[8] T. Diwan, G. Anirudh, and J. V. Tembhurne, "Object detection using YOLO: challenges, architectural successors, datasets and applications," Multimedia Tools and Applications, Aug. 2022, doi: 10.1007/s11042-022-13644-y.

[9] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 779-788, doi: 10.1109/CVPR.2016.91.

[10] C. Wu, Q. Luo, and D. Shou, "Image Recognition Technology and Development of YOLO Algorithm," Advances in transdisciplinary engineering, Dec. 2022, doi: 10.3233/atde221146.

[11] P. Jiang, D. Ergu, F. Liu, Y. Cai, and B. Ma, "A Review of Yolo Algorithm Developments," Procedia Computer Science, vol. 199, pp. 1066–1073, Jan. 2022, doi: 10.1016/j.procs.2022.01.135.

[12] J. Terven and D. Cordova-Esparza, "A Comprehensive Review of YOLO: From YOLOv1 to YOLOv8 and Beyond," arXiv (Cornell University), Apr. 2023, doi: 10.48550/arxiv.2304.00501.

[13] O. E. Olorunshola, M. E. Irhebhude, and A. E. Evwiekpaefe, "A Comparative Study of YOLOv5 and YOLOv7 Object Detection Algorithms," Journal of Computing and Social Informatics, vol. 2, no. 1, pp. 1–12, Feb. 2023, doi: 10.33736/jcsi.5070.2023.

[14] X. Jiang, H. Hu, Y. Qin, Y. Hu, and R. Ding, "A real-time rural domestic garbage detection algorithm with an improved YOLOv5s network model," Scientific Reports, vol. 12, no. 1, Oct. 2022, doi: 10.1038/s41598-022-20983-1.

[15] I. S. Gillani et al., "Yolov5, Yolo-x, Yolo-r, Yolov7 Performance Comparison: A Survey," Computer Science & Information Technology (CS & IT), vol. 12, no. 12, p. 17, Oct. 2022, doi: 10.5121/csit.2022.121602.

[16] B. H. W. Guo, Y. Zou, Y. Fang, Y. M. Goh, and P. X. W. Zou, "Computer vision technologies for safety science and management in construction: A critical review and future research directions," Safety Science, vol. 135, p. 105130, Mar. 2021, doi: 10.1016/j.ssci.2020.105130.

[17] L. Zhou, L. Zhang, and N. Konz, "Computer Vision Techniques in Manufacturing," IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 53, no. 1, pp. 105–117, Jan. 2023, doi: 10.1109/TSMC.2022.3166397.

[18] Z. Kolar, H. Chen, and X. Luo, "Transfer learning and deep convolutional neural networks for safety guardrail detection in 2D images," Automation in Construction, vol. 89, pp. 58–70, May 2018, doi: 10.1016/j.autcon.2018.01.003.

[19] W. Fang et al., "A deep learning-based approach for mitigating falls from height with computer vision: Convolutional neural network," Advanced Engineering Informatics, vol. 39, pp. 170–177, Jan. 2019, doi: 10.1016/j.aei.2018.12.005.

[20] S. Paneru and I. Jeelani, "Computer vision applications in construction: Current state, opportunities & challenges," Automation in Construction, vol. 132, p. 103940, Dec. 2021, doi: 10.1016/j.autcon.2021.103940.

[21] K. Maharana, S. Mondal, and B. Nemade, "A Review: Data Pre-Processing and Data Augmentation Techniques," Global Transitions Proceedings, vol. 3, no. 1, pp. 91–99, Apr. 2022, doi: 10.1016/j.gltp.2022.04.020.

[22] B. Zoph, E. D. Cubuk, G. Ghiasi, T.-Y. Lin, J. Shlens, and Q. V. Le, "Learning Data Augmentation Strategies for Object Detection," Computer Vision – ECCV 2020, pp. 566–583, 2020, doi: 10.1007/978-3-030-58583-9_34.

[23] N. K. Anushkannan, V. R. Kumbhar, S. K. Maddila, C. S. Kolli, B. Vidhya and R. G. Vidhya, "YOLO Algorithm for Helmet Detection in Industries for Safety Purpose," 2022 3rd International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India, 2022, pp. 225-230, doi: 10.1109/ICOSEC54921.2022.9952154.

[24] X. Dong, S. Yan, and C. Duan, "A lightweight vehicles detection network model based on YOLOv5," Engineering Applications of Artificial Intelligence, vol. 113, p. 104914, Aug. 2022, doi: 10.1016/j.engappai.2022.104914.

[25] C. -Y. Wang, H. -Y. Mark Liao, Y. -H. Wu, P. -Y. Chen, J. -W. Hsieh and I. -H. Yeh, "CSPNet: A New Backbone that can Enhance Learning Capability of CNN," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 2020, pp. 1571-1580, doi: 10.1109/CVPRW50498.2020.00203.

[26] R. Xu, H. Lin, K. Lu, L. Cao, and Y. Liu, "A Forest Fire Detection System Based on Ensemble Learning," Forests, vol. 12, no. 2, p. 217, Feb. 2021, doi: 10.3390/f12020217.

[27] C. Fang, H. Xiang, C. Leng, J. Chen, and Q. Yu, "Research on Real-Time Detection of Safety Harness Wearing of Workshop Personnel Based on YOLOv5 and OpenPose," Sustainability, vol. 14, no. 10, p. 5872, May 2022, doi: 10.3390/su14105872.

[28] S. Liu, L. Qi, H. Qin, J. Shi and J. Jia, "Path Aggregation Network for Instance Segmentation," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 2018, pp. 8759-8768, doi: 10.1109/CVPR.2018.00913.

[29] Z. Yi, G. Wu, X. Pan and J. Tao, "Research on Helmet Wearing Detection in Multiple Scenarios Based on YOLOv5," 2021 33rd Chinese Control and Decision Conference (CCDC), Kunming, China, 2021, pp. 769-773, doi: 10.1109/CCDC52312.2021.9602337.

[30] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid and S. Savarese, "Generalized Intersection Over Union: A Metric and a Loss for Bounding Box Regression," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 2019, pp. 658-666, doi: 10.1109/CVPR.2019.00075.

[31] R. Padilla, S. L. Netto and E. A. B. da Silva, "A Survey on Performance Metrics for Object-Detection Algorithms," 2020 International Conference on Systems, Signals and Image Processing (IWSSIP), Niteroi, Brazil, 2020, pp. 237-242, doi: 10.1109/IWSSIP48289.2020.9145130.