CLOTH BAG OBJECT DETECTION USING THE YOLO ALGORITHM (YOU ONLY SEE ONCE) V5

Rizki Hesananda [1*)] ; Desima Natasya Simatupang [2] ; Ninuk Wiliani [3]

Information Technology Study Program
Bank Rakyat Indonesia Institute of Technology and Business
Jakarta, Indonesia
www.bri-insitute.ac.id
[1*)] hessananda@bri-insitute.ac.id ; [2] desima.natasya.s@bri-insitute.ac.id ; [3] ninukwiliani15@gmail.com

(*) Corresponding Author

Abstract —The use of plastic in modern life is increasing rapidly, causing the number of people who use plastic to increase, one of which is when shopping. The function of plastic bags as packaging for luggage is not comparable to the impact caused by plastic waste in the years to come. Plastic bags take a long time, even hundreds to thousands of years, to completely decompose. In order to support the government's program to reduce the use of plastic bags, this study will discuss how to detect cloth bags as a substitute for plastic bags. In this research, a system will be implemented to detect the use of cloth bags with Roboflow and Yolo v5. After carrying out all stages of the research, it can be concluded that the goodie bag detection model has been successfully created. The detection model was created using the YOLOV5 algorithm. The dataset used consists of 102 goodie bag images. The process model uses 100 epochs with the training result mAP@0.5 is 89.8%. So, in other words, it can be said that YOLO v5 can detect goodie bags very well.

Keywords: object detection, yolo v5, computer vision.

Abstrak— *Penggunaan plastik dalam kehidupan modern ini meningkat sangat pesat sehingga menyebabkan tingkat ketergantungan manusia pada plastik semakin tinggi, salah satunya pada saat berbelanja. Fungsi kantong plastik sebagai pembungkus barang-barang bawaan tidak sebanding dengan efek yang ditimbulkan dari sampah plastik sampai tahun-tahun yang akan datang. Kantong plastik membutuhkan waktu lama bahkan sampai ratusan hingga ribuan tahun untuk dapat terurai sempurna. Dalam rangka mendukung program pemerintah dalam pengurangan penggunaan kantong plastik maka dalam penelitian ini akan membahas cara untuk mendeteksi cloth bag sebagai pengganti kantong plastik. dalam penelitian ini akan mengimplementasikan sebuah sistem untuk mendeteksi penggunaan kantong kain dengan Roboflow dan yolo v5. Setelah melakukan seluruh tahapan penelitian maka dapat diambil model deteksi goodie bag sudah berhasil dibuat. Model deteksi dibuat menggunakan algoritma YOLOV5. Dataset yang digunakan terdiri dari 102 gambar goodie bag. Proses pelatihan model menggunakan 100 epoch dengan hasil mAP@0.5 adalah 89,8 %. Maka dengan kata lain dapat disimpulkan bahwa algoritma YOLO v5 dapat mendeteksi objek goodie bag dengan sangat baik*

Kata Kunci*: deteksi objek, yolo v5, computer vision.*

INTRODUCTION

The use of plastic in modern life is increasing rapidly, causing the level of human dependence on plastic to be higher, one of which is when shopping. In Indonesia, traders usually use plastic bags to accommodate consumer shopping for goods. Even though it is known that plastic bags will increase plastic waste(Sardon & Dove, 2018). Plastic waste is a material that is difficult to decompose, so it can damage the environment.

Many people use plastic bags because plastic is a packaging material or container that is practical and looks clean, easy to get, durable, and cheap. The function of plastic bags as wrapping for luggage is not comparable to the effects caused by plastic waste for years to come(Zulkifley et al., 2014). Plastic bags take a long time, even hundreds of years, to completely decompose.

In order to support the government's program to reduce the use of plastic bags is the right step and has a very significant impact on various parties. This is the first step for researchers to implement cloth bags to reduce plastic waste. Goodie bags and tote bags as shopping or daily necessities can also be used repeatedly. In addition, bags are made of cloth, plastic, cardboard, and foam art, usually used as gifts, keepsakes, celebrations, and for other household needs. The users are all age groups with various purposes.

Today the world is in the digital age. An era where almost every aspect of human life is closely related to computing technology. With the development of knowledge, humans continue to develop technology to help lighten the work. In computer vision, several problems include object detection and image classification. Object detection has recently become one of the most exciting fields in computer vision and artificial intelligence. Object detection is a computer technology related to computer vision and image processing related to detecting an object in a digital image in color and object shape (Nahdi Saubari, 2019).

Based on the description above, this research will implement a system to detect the use of cloth bags with Roboflow and yolo v5(Liunanda et al., 2020). Moreover, the hope is that this research will be able to properly detect the use of Goodie bags and Tote bags as shopping bags or daily necessities so that later the information can be helpful for those who need it.

MATERIALS AND METHODS

The first step in this study was to collect a dataset from images of cloth bags. The dataset is the core of all processes. This happens because the entire subsequent process is determined by the quality and quantity of the data set that has been collected. Datasets can be collected from various sources. Currently, it is very effective to collect datasets from the internet or take photos via smartphones(Kumari et al., 2020).

Annotation is the second step in the whole research process. Annotations are performed on each dataset image with an object image. Annotations are done by drawing one by one using a "bounding box" box. This box will tell the engine about the part of the object being searched for and the part being ignored.

The next step is preprocessing and augmentation of the dataset. This step is needed to improve the image quality so that it meets the standards required by the machine in "learning." The image will be rotated, cropped, and color changed to make the model more effective and accurate. Preprocessing and augmentation are applied to the entire dataset. So it is possible to add datasets after this process is carried out.

The training was carried out to design machines to recognize objects that have been annotated, preprocessed, and augmented. In this study, the training was conducted using yahoo YOLO v5. Training in YOLO v5 requires several parameters, including image width and age. Epoch is the number of times the model is curious. So, the higher the assumption, the better the model will be. The training process can be done repeatedly to produce a suitable model.

YOLO (You Only Look Once) is a real-time object detection algorithm (Lu et al., 2019). The basic idea behind the method is to divide the input image into a grid of cells and then, for each cell, predict the probability of the presence of an object in that cell and the bounding box coordinates for any object present. YOLO uses a single convolutional neural network (CNN) to perform object classification and bounding box regression, making it more efficient than other object detection methods that typically use two separate networks.

The architecture of YOLO consists of a sequence of convolutional and max-pooling layers (Zhao et al., 2020), followed by several fully connected layers. The network's output is a tensor of shape (S, S, (B x 5 + C)), where S is the number of grid cells, B is the number of bounding boxes per cell, and C is the number of objects classes. Each cell in the grid produces B-bounding box predictions and C-class predictions, along with a confidence score for each bounding box(Liunanda et al., 2020).

The critical innovation of YOLO is its ability to make predictions at multiple scales by using anchor boxes of different sizes (Hurtik et al., 2022). This allows the network to detect objects of different sizes in the same image and helps improve the localization accuracy of the bounding boxes(Liunanda et al., 2020).

YOLO is a speedy method for object detection, and it can process up to 45 frames per second (Adou et al., 2019) on a standard GPU. It is also relatively simple to train and use, making it a popular choice for many real-time object detection applications.

After the training on the dataset is completed, the model that has been formed needs to be tested on a random image to prove whether the model has successfully detected the object. The research is considered complete if the model has succeeded in detecting the object. If not, the training process will be repeated with different parameters(Agroui et al., 2017).

Furthermore, the model's performance can be achieved by several methods: Recall, Precision, F1, Intersection over Union, mean Average Precision, and Accuracy. The recall method can be obtained by calculating the ratio of the total number of positive samples with the correct classification results compared to the total number of positive samples(Fadilla et al., 2011). Recall with a high score indicates that the class is known correctly (little FN). Equation 1 is as follows:

$$Recall = \frac{TP}{TP+FN} \text{ .... } (1)$$

The following method is Precision. The value of Precision is calculated by dividing the total number of positive samples with a correct classification result by the total number of positive samples predicted. Equation 2 as follows:

$$Precision = \frac{TP}{TP+FP} \text{ ....(2)}$$

Information:
True Positive (TP) = actual value positive and predicted positive value
True Negative (TN) = actual value negative and predicted negative value
False Positive (FP) = the actual value is negative but predicted to be positive
False Negative (FN) = the actual value is positive but is predicted to be negative

The condition when Recall is high and Precision is low means most of the positive instances are correctly recognized (low FN), but there are still a lot of *False Positives* (high FP)(Redmon et al., 2016). Meanwhile, if the recall conditions are low and the Precision is high, the model loses many positive samples (high FN) with a few *false positive values* (low FP).

The following method is F1 Score. The FI score is calculated by comparing the average Precision and recallsweighted as in equation 3

$$F1\ Score = 2 \times \frac{Recall \times Precision}{Recall+Precision} \text{ ....(3)}$$

Intersection over Union (IoU) is an evaluation metric to measure the accuracy of object detectors in a data set. IoU can be used as long as it has a *ground-truth bounding box on the object and a prediction* dataset *bounding box* on the dataset object. IoU can be calculated by comparing *the ground-truth bounding box* with *the predicted bounding box* in the model that has been made.

While the *Mean Average Precision* (mAP)(Kumar & Srivastava, 2020) method is the average value of *Average Precision* (AP) which will form *a metric evaluation* to measure *the performance* of an object detection algorithm in order to measure the accuracy of the available models, a trial will be carried out with random images from the *testing dataset,* and analysis will be carried out using the following equation 4:

$$Accuration = \frac{\Sigma \text{ correct prediction}}{\text{N}} \times 100\% \text{............(4)}$$

RESULTS AND DISCUSSION

The dataset used in this study is all images of Goodie bags and Tote bags, while the sample in this study is images of Goodie bags and Tote bags with a total of 70 images. Images were obtained through camera capture and from google images. The images are then collected in one folder, as shown in Figure 1.

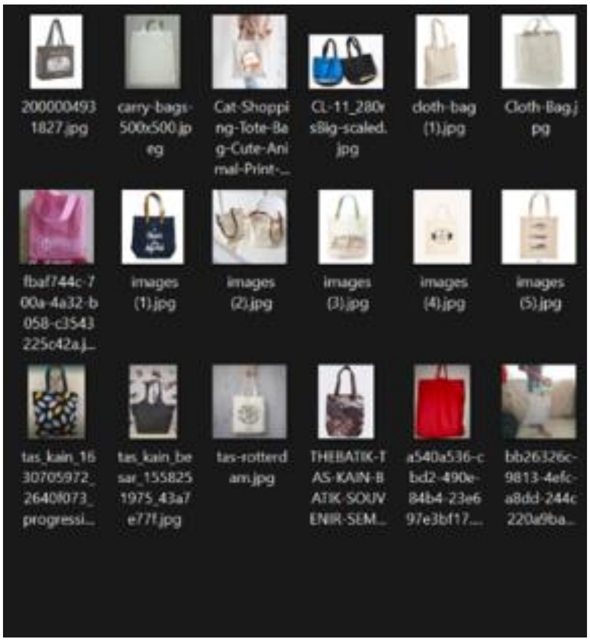

200000493 1827.jpg
carry-bags-500x500.jpeg
Cat-Shopping-Tote-Bag-Cute-Animal-Print-...
CL-11_280r sBig-scaled.jpg
cloth-bag (1).jpg
Cloth-Bag.jpg
fbaf744c-700a-4a32-b058-c3543225c42a.j...
images (1).jpg
images (2).jpg
images (3).jpg
images (4).jpg
images (5).jpg
tas_kain_1630705972_2640f073_progressi...
tas_kain_besar_1558251975_43a7e77f.jpg
tas-rotterdam.jpg
THEBATIK-TAS-KAIN-BATIK-SOUVENIR-SEM...
a540a536-cbd2-490e-84b4-23e697e3bf17....
bb26326c-9813-4efc-a8dd-244c220a9ba...

Figure 1. Goodie Bag and Tote Bag Images

The next step is to annotate all the images that have been collected. To perform annotations, researchers used a tool, namely *Roboflow Annotate,* to add boxes around Goodie bags and Tote bags that did not have them. Figures 2 and 3 are examples of the annotation process on images in the dataset.

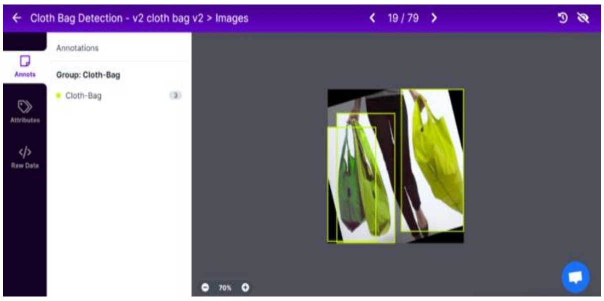


Figure 2. Annotation Process 1

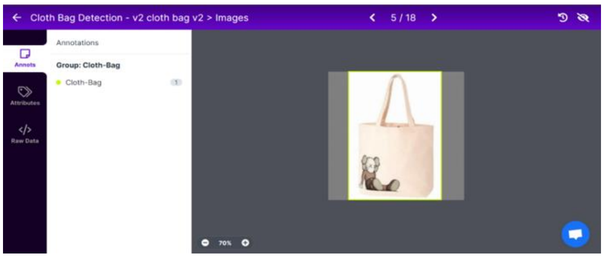


Figure 3. Annotation Process 2

The labeling begins by creating a *bounding box* on the object with the goodie bag and toot bag in the image. The bounding box is then given a class name, "cloth bag," to know that this section is the object in question.

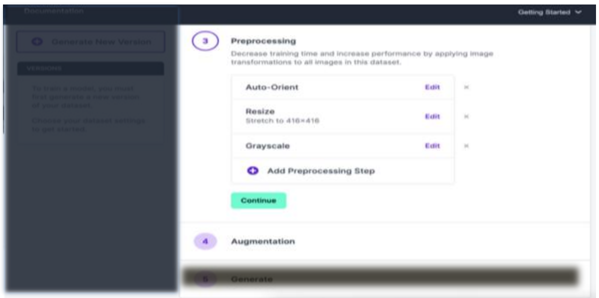

Documentation
Getting Started
Generate New Version
VERSIONS
To train a model, you must first generate a new version of your dataset.
Choose your dataset settings to get started.
3
Preprocessing
Decrease training time and increase performance by applying image transformations to all images in this dataset.
Auto-Orient
Edit
Resize
Stretch to 416×416
Edit
Grayscale
Edit
Add Preprocessing Step
Continue
4
Augmentation
5
Generate

Figure 4. *Preprocessing* and *Augmentation Process*

After all the data is annotated, the following process is *preprocessing* and *augmentation*. The annotation process aims to properly bind each labeled object even if it resizes, rotates, or. In this research, *auto-orient*, *resize,* and *grayscale are used*. Automatic orientation ensures that images are stored on disk in the same way that applications open them. *Resize* creates a consistent size for the image (in this case, a more minor to speed up the *training process*). Grayscale converts an RGB color image with three channels into a black & white image (grayscale) with one channel(Oktavianto & Purboyo, 2018).

The *augmentation process* used in this research is *random flip*, 90-degree random rotation, and 10 to 30-degree random rotation. The *preprocessing* and *augmentation processes* resulted in as many as 102 new images for the dataset, with the *training compositions set* increased to as many as 79 images, the *validation set* of 18 as many pictures, and the *testing set* of 5 images.

The final step in creating a data set is to convert the data set into a format suitable for the *training process* in Yolov5, as shown in figure 5.

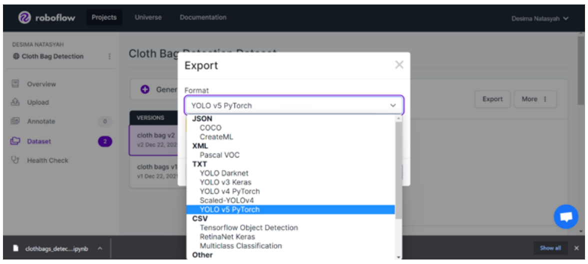
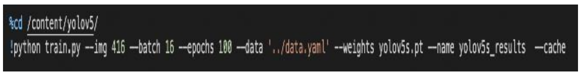


Figure 5. Export dataset according to YOLO v5 format

Training, Validation, and Testing



```
%cd /content/yolov5/
!python train.py --img 416 --batch 16 --epochs 100 --data '../data.yaml' --weights yolov5s.pt --name yolov5s_results  --cache
```

Figure 6. *Train set up* in YOLO v5

In Figure 6, *training* on the dataset is carried out after all images in the dataset have annotations of the object class to be detected. In performing training, several parameters are needed. The first parameter to be used is the image size, which is 416 pixels. Determining the image size to speed up the *training process*. Second is an epoch, an epoch is the number of iterations of data training, and after that, the batch is the number of data to be learned in each iteration. In other cases, adjustments must be made depending on the data processing machine's capabilities and available time.

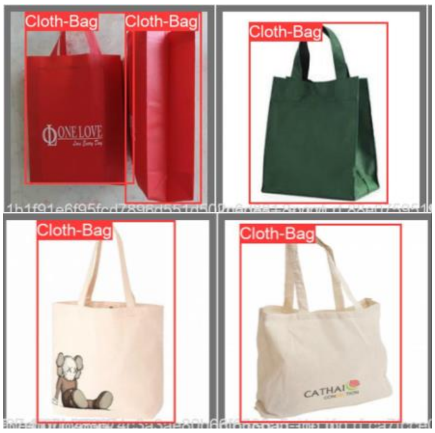
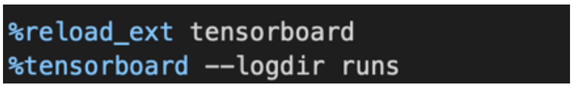


Figure 7. Image for Validation

After the dataset *training process is complete, the next step is to carry out the validation process.* The process aims to assess the performance of the training model. Validation uses 18% of the images in the dataset. These settings have been saved in the YAML file. Validation is used to evaluate the performance model. The existing model will be compared with the validation dataset to measure the level of accuracy. The results can be seen from the MAP value displayed in the evaluation process with the command shown in Figure 8.



```
%reload_ext tensorboard
%tensorboard --logdir runs
```

Figure 8. Evaluation Order Model

The evaluation stage is carried out by comparing the ground truth box and prediction box. The ground truth box is obtained from a dataset that has been labeled using the roboflow tool. The prediction box is the result of detection from the model that has been created. The dataset that has been labeled as ground truth can be seen in Figure 8 below:

The model of the training process creates the prediction box. In the corner of the box, there is a number that shows how confident the model predicts the image it is processing. If the number given by the model is close to 1 or 100%, then the model believes that the object is a cloth bag.

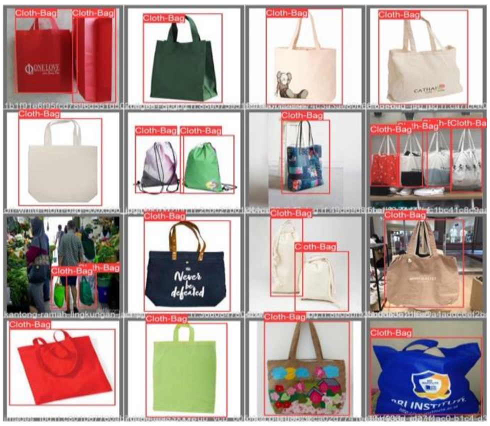


Figure 8. Basic Truths

If the number is close to 0, the model is not sure that the object is a cloth bag. The results of the box prediction can be seen in Figure 9 below:

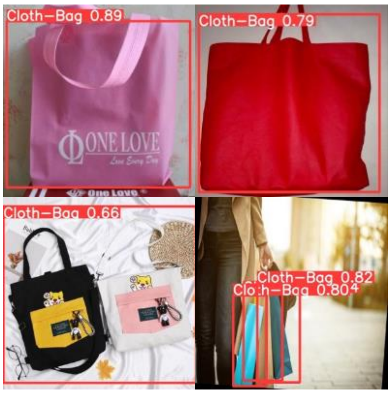


Figure 9. Prediction Results Box

After 100 epochs of training, the resulting model is detected with the following specifications:

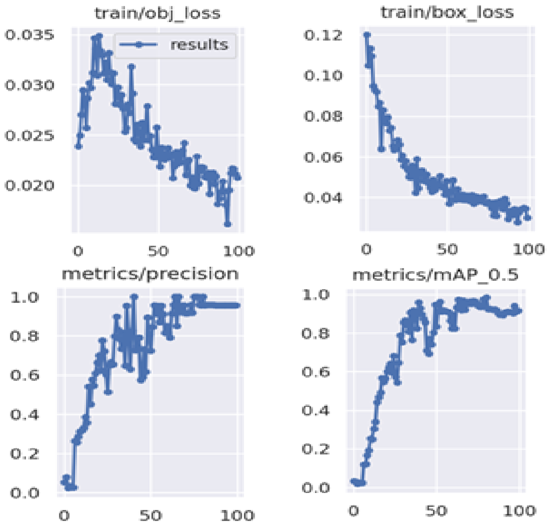


Figure 10. Results of Model Evaluation

The objective loss (obj loss) is typically a combination of two terms: a classification loss and a localization loss. The classification loss measures how well the model can predict the correct class label for each object, while the localization loss measures how well the model can predict the bounding box coordinates for each object.

The box loss (also known as the "coordinate loss") is a term that explicitly measures the localization loss. It penalizes the model when the predicted bounding box coordinates are inaccurate, encouraging it to learn to predict more accurate bounding boxes.

In YOLO, the objective loss and box loss are used together to optimize the model's performance on the object detection task. The model's parameters are adjusted during training to minimize both loss functions to improve the model's ability to predict both the class labels and bounding box coordinates of objects in an image.

It can be seen in Figure 10 that our results show that the objective loss value decreased significantly after 100 epochs of training. This suggests that the model could learn the patterns in the training data and improve its performance over time. The optimization algorithm also appears to be working effectively, adjusting the model's parameters to minimize the loss. These findings suggest that our model can fit the training data well and make accurate predictions. Further research must confirm these results and explore potential limitations or generalizability to other datasets.

During training, we observed that both Precision and mean average Precision (mAP) increased significantly after 100 epochs. Precision measures the ability of the model to correctly classify objects as positive or negative, while mAP measures the overall accuracy of the object detection task by taking into account both the Precision and the Recall of the model(Zhang et al., 2020).

The improvement in Precision and mAP indicates that the model is becoming more accurate in identifying and classifying objects in the images. This is a positive sign, as it suggests that the model is learning the patterns in the training data and can generalize to new, unseen data.

CONCLUSION

After completing all stages from designing, testing, and analyzing, it can be concluded that the goodie bag detection model has been successfully created. The detection model was created using the YOLOV5 algorithm. The dataset used consists of 102 goodie bag images. The process model using 100 epochs with training results mAP@0.5 is 89.8%. So in other words, it can be said that YOLO v5 can detect goodie bags very well.

It is worth noting that the increase in Precision and mAP may not be linear throughout training. There may be fluctuations in the metric values, especially if the model is being trained using mini-batches or if the learning rate is not set optimally. However, the model will likely improve if the overall trend is upward.

Further research is needed to confirm these findings and explore potential limitations or factors that may have contributed to the increase in Precision and mAP. It will also be interesting to see if these improvements are sustained on test datasets and real-world applications."

REFERENCE

Adou, M. W., Xu, H., & Chen, G. (2019). Insulator Faults Detection Based on Deep Learning. *Proceedings of the International Conference on Anti-Counterfeiting, Security and Identification, ASID*, *2019-October*, 173–177. https://doi.org/10.1109/ICASID.2019.8925094

Agroui, K., Pellegrino, M., & Giovanni, F. (2017). Analysis Techniques for Photovoltaic Modules Based on Amorphous Solar Cells. *Arabian Journal for Science and Engineering*, *42*(1), 375–381. https://doi.org/10.1007/s13369-016-2050-5

Fadilla, L., Rizal, A., & Rachmawati, E. (2011). *Implementasi dan Analisis Content-Based Image Retreival Pada Citra X-Ray Menggunakan Algoritma Hierarki dan Algoritma Fast Genetic K-Means* [Universitas Telkom]. https://repository.telkomuniversity.ac.id/pustaka/95543/implementasi-dan-analisis-content-based-image-retrieval-pada-citra-x-ray-menggunakan-algoritma-hierarki-dan-algoritma-fast-genetic-k-means.html

Hurtik, P., Molek, V., Hula, J., Vajgl, M., Vlasanek, P., & Nejezchleba, T. (2022). Poly-YOLO: higher speed, more precise detection and instance segmentation for YOLOv3. *Neural Computing and Applications*, *34*(10), 8275–8290. https://doi.org/10.1007/S00521-021-05978-9/METRICS

Kumar, A., & Srivastava, S. (2020). Object Detection System Based on Convolution Neural Networks Using Single Shot Multi-Box Detector. *Procedia Computer Science*, *171*(2019), 2610–2617. https://doi.org/10.1016/j.procs.2020.04.283

Kumari, R., Gupta, N., & Kumar, N. (2020). Image Segmentation using Improved Genetic Algorithm. *International Journal of Advanced Science and Technology (IJEAT)*, *29*(4), 597–601. https://doi.org/10.35940/ijeat.F9063.109119

Liunanda, C. N., Rostianingsih, S., & Purbowo, A. N. (2020). Implementasi Algoritma YOLO pada Aplikasi Pendeteksi Senjata Tajam di Android. *Jurnal Infra*, *8*(2), 235–241. http://publication.petra.ac.id/index.php/teknik-informatika/article/view/10527

Lu, S., Wang, B., Wang, H., Chen, L., Linjian, M., & Zhang, X. (2019). A real-time object detection algorithm for video. *Computers & Electrical Engineering*, *77*, 398–408. https://doi.org/10.1016/J.COMPELECENG.2019.05.009

Nahdi Saubari. (2019). Deteksi Citra Wajah Dengan Metode Haar Feature Selection. *Jurnal Teknologi Informasi Universitas Lambung Mangkurat (JTIULM)*, *4*(1), 7–12. https://doi.org/10.20527/jtiulm.v4i1.33

Oktavianto, B., & Purboyo, T. W. (2018). A Study of Histogram Equalization Techniques for Image Enhancement. *International Journal of Applied Engineering Research*, *13*(2), 1165–1170. https://www.ripublication.com/ijaer18/ijaerv13n2_45.pdf

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). *Yout Only Look Once: Unified, Real-Time Object Detection*. *27*(3), 306–308. https://doi.org/10.1021/je00029a022

Sardon, H., & Dove, A. P. (2018). Plastics recycling with a difference. *Science*, *360*(6387), 380–381. https://www.science.org/doi/abs/10.1126/science.aat4997

Zhang, X., Hao, Y., Shangguan, H., Zhang, P., & Wang, A. (2020). Detection of surface defects on solar cells by fusing Multi-channel convolution neural networks. *Infrared Physics & Technology*, *108*, 103334. https://doi.org/https://doi.org/10.1016/j.infrared.2020.103334

Zhao, Y., Deng, X., & Lai, H. (2020). A YOLO-Based Method to Recognize Structural Components from 2D Drawings. *Construction Research Congress 2020: Computer Applications - Selected Papers from the Construction Research Congress 2020*, 753–762. https://doi.org/10.1061/9780784482865.080

Zulkifley, M. A., Mustafa, M. M., Hussain, A., Mustapha, A., & Ramli, S. (2014). Robust identification of polyethylene terephthalate (PET) plastics through bayesian decision. *PLoS ONE*, *9*(12), 1–20. https://doi.org/10.1371/journal.pone.0114518