DOI: https://doi.org/10.52436/1.jutif.2024.5.5.1977
p-ISSN: 2723-3863
e-ISSN: 2723-3871

STUDENT FOCUS DETECTION USING YOU ONLY LOOK ONCE V5 (YOLOV5) ALGORITHM

Rosalina*[1], Fitri Bimantoro[2], I Gede Pasek Suta Wijaya[3]

[1,2,3]Departement of Informatics Engineering, Faculty of Engineering, Universitas Mataram, Indonesia
Email: [1]rossalinaa11@gmail.com, [2]bimo@unram.ac.id, [3]gpsutawijaya@unram.ac.id

(Article received: April 06, 2024; Revision: May 06, 2024; published: October 20, 2024)

Abstract

Education has a very important role in life, student involvement in the learning process in the classroom is an important factor in the success of learning. However, some students pay less attention to the lesson, indicating a lack of productivity in learning. The use of machine learning and computer vision techniques has undergone significant development in the last decade and is applied in a variety of applications, including monitoring student attention in the classroom. One of the commonly used techniques in machine learning and computer vision to detect objects is by applying image processing. One of the algorithms implemented for object detection that can provide good results is You Only Look Once. This research proposes the application of YOLOV5 in real time student focus detection and analyzes the performance and computational load of the five YOLOV5 architectures (YOLOV5n, YOLOV5s, YOLOV5m, YOLOV5l, and YOLOV5x) in student surveillance during classroom learning. The dataset used is video data that has been converted into image form, and 297 images are produced. Where, this dataset is divided into 2 classes, namely the "Focus" and "Not Focus" classes. The results show that YOLOV5x has the highest computational load with large parameter values and GFLOPs. However, in term model performance YOLOV5m provides more optimal results than other architectures, with precision of 83.3%, recall of 85.1%, and mAP@50 of 89.9%. The results of this study show that the proposed YOLOV5 model can be a good performing method in detecting student focus in real time.

Keywords: *education, object detection, student, YOLOV5.*

1. INTRODUCTION

Education is an important aspect that must be an integral part of an individual to achieve prosperity in life. Because through education, humans can develop better, develop critical thinking skills, and overcome various challenges faced in their lives [1]. In Law Number 20 of 2003 Article 1 paragraph 1 defines education as a conscious and planned effort to create a learning atmosphere and learning process so that students actively develop their potential to have religious spiritual strength, self-control, personality, intelligence, noble character, and skills needed by themselves, society, nation and state. Learning activities are behavior or behavior in carrying out a learning process. Student engagement is one of the factors contributing to the success of a learning activity [2]. However, there were only 46% to 67% of students who paid attention during the lesson, most of them lose focus after around 10 minutes [3]. This means that some students are not productive in their learning. Student focus in the classroom is an indicator of the level of diligence of students in paying attention and understanding the learning material. It is not only useful for assessing how effective students are in learning during the learning session, but also shows the quality of teaching in the classroom. As such, student focus in the classroom is an important factor in raising the standard of education today [4]. It is important to know the potential factors that might cause this situation, and in which classroom situations students tend to lose their focus. With this information, educators can identify potential problems during the learning process and can work to address these issues.

The use of machine learning and computer vision techniques has made great progress in the past ten years and has been successfully applied in applications such as automated assessment, security, and image data investigation [5]. Machine learning is also utilized in the field of education, such as to monitor students' attention and engagement [6]. One of the commonly used techniques in machine learning and computer vision for object detection is to apply image processing. This technique involves using machine learning algorithms and models to analyze and interpret visual information in digital images to identify specific objects, patterns or features. One of the algorithms implemented for object detection that can provide good results is You Only Look Once [7]. The object detection system using the YOLO method is proven to be more accurate and faster in detecting video objects in real-time. In real-time object detection, detection speed is very important. If the object detection takes too long, the resulting video will be broken because of the delay in each frame [8].

Along with its development, YOLO evolved into several versions. One of them is YOLOV5. Research on object detection using the YOLOV5 model has been carried out to detect objects such as detection of defects in kiwi fruit, detection of the use of face masks, detection of the use of safety helmets, and so on [9].

Previous research conducted by [10] created a model for student behavior recognition by identifying actions and emotions/face expressions to recognize students' attention or inattention during class using YOLOV5. The dataset used consists of 2 categories, namely actions and emotions. The action category consists of 9 classes (raising hand, focused, eating, distracted, reading a book, using a phone, writing, bored, and laughing), and the emotion category consists of 5 classes (happy, sad, angry, surprise, and neutral). This research resulted in model performance with an average accuracy of 76%. Research conducted by [11] tested 300 face images from 5 student samples with a total of 1500 images using the YOLOV5 algorithm for a student attendance system that allows to recognize student faces. This research uses 5 classes of student datasets. The results of research that has been done with 100 epochs, 16 batches, and 640×640pixel image size obtained a mAP value of 99.5%, precision 99.7%, and recall 99.4%. Then it refers to research [12] which analyzes the comparison of YOLOV5 and YOLOV7 by training custom models independently to consider their performance. The dataset used consists of 9,779 images with 21,561 annotations from four classes, namely Persons, Handguns, Rifles, and Knives, obtained from Google Open Images Dataset, Roboflow Public Dataset, and local datasets. YOLOV7 achieved a precision score of 52.8%, recall 56.4%, mAP@0.5 51.5%, and mAP@0.5:0.95 31.5%. While YOLOv5 has a precision of 62.6%, recall 53.4%, mAP@0.5 55.3%, and mAP@0.5:0.95 34.2%. The research shows that YOLOv5 is superior in precision, mAP@0.5, and mAP@0.5:0.95 than YOLOV7 overall.

Based on previous research, this research will design a model that is able to detect student focus in real time using the five types of YOLOV5 architecture. The difference between this research and the previous research is that the previous research used data on one individual object in one image, while this research will use data on students who are in the classroom taken from CCTV videos, where there are many objects in the recording. This research aims to detect the focus of students in the classroom environment, and compare the performance and computational load of the five types of YOLOV5 used (YOLOV5n, YOLOV5s, YOLOV5m, YOLOV5l, YOLOV5x), and determine the type of architecture that has the most optimal performance. The selection of the method is because YOLOV5 is quite fast in detecting objects in real time and produces quite high accuracy and has the ability to recognize smaller objects well, additionally it does not require large memory capacity [13], [14]. In addition, YOLOV5 also has advantages in terms of deployment because the resulting model is lighter and has a smaller size [15]. This research is hoped to be able to help educators in monitoring the level of student focus during class, so that it can be used as an evaluation material to improve their performance.

2. METHOD

This research detects focused and unfocused students using the You Only Look Once V5 (YOLOV5) method with 5 architectures, namely, YOLOV5n, YOLOV5s, YOLOV5m, YOLOV5l, and YOLOV5x. The stages of this research include dataset collection, pre-processing, Trained YOLOV5, and model evaluation. For more details, it can be seen in the research flow chart as shown in Figure 1.

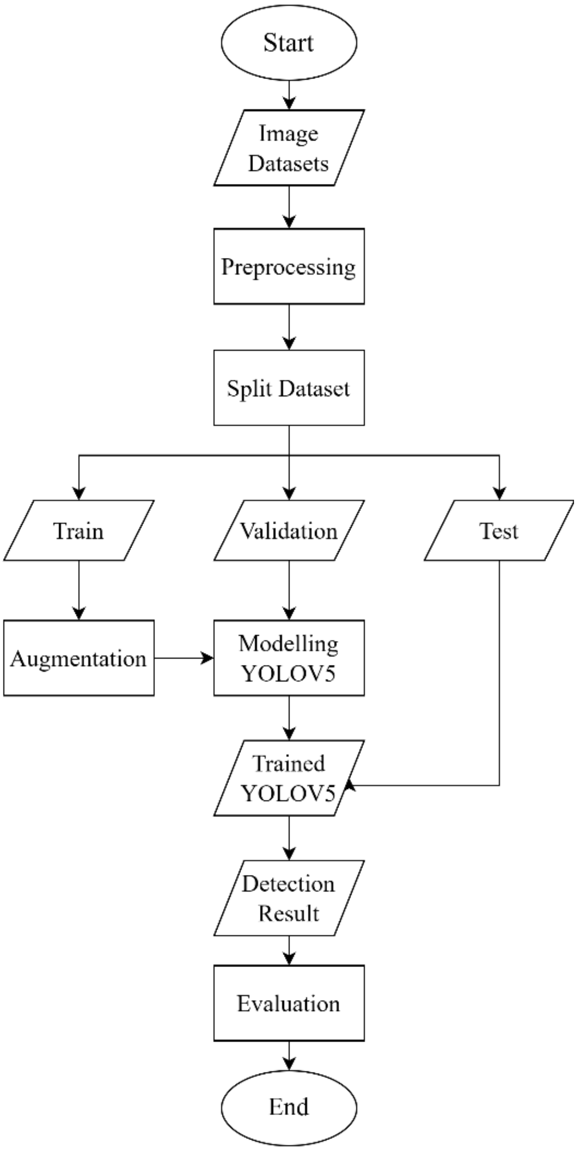


Figure 1. Research Flow Diagram

2.1. Data Sources

This research started by taking a dataset that will be used as input for the detection of focused and unfocused objects for students. The dataset used is a video dataset collected by the researcher himself. This dataset was taken in classroom A3.01, Building A, Faculty of Engineering, Mataram University using a Canon EOS 1300D camera.

2.2. Pre-processing

The pre-processing stage in this research consists of image annotation, split data, and resize. The following is an explanation of each of these stages:

1. **Image Annotation**

The image is labeled and bounding box according to the class of the object to be detected using tools from the Roboflow platform. Roboflow is one of the platforms in object detection that can be used to label or annotate images [16]. In addition, the platform provides a variety of public datasets that are easily accessible. Users are also given access to upload datasets according to their needs.

2. **Split Data**

The next step is to divide the collected and annotated dataset into training data, validation data, and test data. Data separation is done to make the training process more efficient and reduce the possibility of overfitting and underfitting on the dataset [11].

3. **Resize**

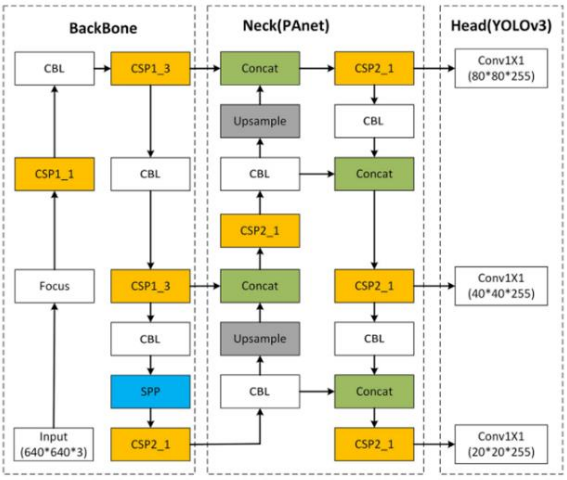
This process is done to resize the image to a uniform size, so that it matches the size read by the model [17].

2.3. Augmentation

Augmentation is the process of altering an image dataset with the intention of creating variations of the existing data, with the aim of training a model using a variety of training examples so that the model can be more adaptive and able to recognize objects or patterns in a variety of different situations. Examples of augmentation include changing rotation, brightness, contrast, flip, and combining images into collages [17].

2.4. YOLOV5 Model

YOLO v5 is a convolutional neural network that excels in object detection speed [18]. YOLOV5 it's also an innovative object detection algorithm renowned for its reliability, high accuracy, and simplicity. It was released by Ultralytics in June 2020. YOLOV5 has five main models, including: YOLOV5n, YOLOV5s, YOLOV5m, YOLOV5l and YOLOV5x. The five models have different characteristics in terms of inference speed, parameter size, and model accuracy [15]. YOLOV5 also has three main components namely: backbone, neck, and head. In YOLOV5, the backbone used is CSPDarknet53. The backbone acts as part of the convolutional neural network that is tasked with extracting important features in the image. Meanwhile, CSPDarknet53 has a role in iteratively splitting and merging gradient information, and integrating gradient changes into the feature map. This has the effect of improving the accuracy and efficiency of the model and reducing the size of the model by reducing the number of parameters [19]. While the neck is located in the middle of the backbone and the head acts as a link between the two [20]. The YOLOV5 neck serves to collect and refine the features extracted by the backbone, focusing on enhancing spatial and semantic information at multiple scales. And the YOLOV5 head consists of three branches, each predicting features at different scales. Each head generates a bounding box, class probability, and confidence score. YOLOV5 uses the same head structure as YOLOV3 and YOLOV4. Finally, the network uses Non-maximum Suppression (NMS) to filter out overlapping bounding boxes [19].



Gambar 2. YOLOV5 Architecture [3]

The first process in the YOLOV5 architecture starts with inputting the image into the backbone component, CSPDarknet53, which is responsible for extracting features from the image. These features are then combined through the PANet (neck) before finally being sent to the head component. The head has the role of detecting objects in the image. YOLOV5 has three different detection stages, designed to ensure detection of objects from small to large scale in the image. To achieve this, YOLOV5 uses three different scales in each grid, namely 8, 16, and 32.

For example, if an image of 640×640 pixels is used, then an 80×80 grid is used to detect small objects, a 40×40 grid for medium objects, and a 20×20 grid for large objects. Each of these scales has a grid that generates 3 anchor boxes of different sizes. Thus, the total bounding boxes generated are ((80×80) + (40×40) + (20×20)) × 3 = 25200. The detection results from these three stages are then combined, and a Non-Max Suppression (NMS) process is used to select one bounding box with the highest confidence value if there are multiple bounding boxes covering a single object [20].

2.5. Trained YOLOV5 Model

In this study, the YOLOV5 models trained are YOLOV5n, YOLOV5s, YOLOV5m, YOLOV5l, and YOLOV5x.

2.6. Evaluation

At this stage, the performance of the trained model is evaluated to determine the extent to which the model has been successful. Usually, the model will be assessed using specific pa-rameters to determine how well it performs. Model performance results are measured using a confusion matrix. Confusion matrix is a matrix that displays the actual classification prediction and the predicted classification [21].

Table 1. Confusion Matrix

		Prediction	
		Positive	Negative
Actual	Positive	TP	FN
	Negative	FP	TN

There are four classifications in the confusion matrix, namely True Negative (TN), True Positive (TP), False Negative (FN), and False Positive (FP) obtained from actual and predicted values. The following is an explanation of the four classifications [22]:

1) TP (True Positive) is the number of positive samples that are correctly classified.
2) TN (True Negative) is the number of negative samples that are correctly classified.
3) FP (False Positive) is the number of negative predictions that are incorrectly classified as positive.
4) FN (False Negative) is the number of positive samples that are misclassified as negative.

Table 1. Shows the evaluation matrix that will be used to calculate the performance of the model, which can be calculated using the precision, recall, and mAP@50 obtained from the confusion matrix.

1. **Precision**

Precision is a basic metric used in object detection to assess the accuracy of the model's positive predictions. It measures the percentage of positive objects correctly identified by the model out of the total objects predicted as positive [23].

$$Precision = \frac{\text{TP}}{\text{TP+FP}} \qquad (1)$$

2. **Recall**

Recall is the ability of a model to find all relevant cases (all bounding boxes of real objects). It is the percentage of correct positive predictions among all given ground truths [24].

$$Recall = \frac{\text{TP}}{\text{TP+FN}} \qquad (2)$$

3. **Mean Average Precision (mAP)**

Mean Average Precision (mAP) is a commonly used evaluation metric in object detection. mAP measures the balance between precision and recall by calculating the average precision (AP) for each class and then taking the average value of all classes. AP measures precision at various recall levels by calculating the area under the prsecision-recall curve [25].

$$mAP = \frac{\mathbf{1}}{\boldsymbol{N}} \sum_{i=1}^{N} APi \qquad (3)$$

3. RESULTS

This research uses Python programming language and Google Collaboratory to build the model. This research analyzes the computational load and performance of the five YOLOV5 architectures, namely YOLOV5n, YOLOV5s, YOLOV5m, YOLOV5l, and YOLOV5x.

3.1. Dataset

The dataset utilized in this research consists of video dataset that has been converted into individual image frames. The result of this conversion yielded a total of 297 images. An example of these image dataset can be seen in Figure 3.

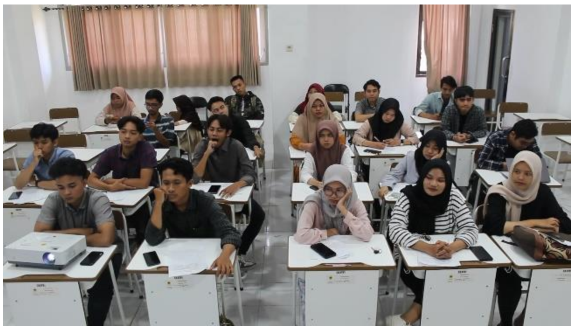


Figure 3. Sample Dataset

Figure 3 displays video dataset that has been converted into individual image frames. Within these images are students situated within a classroom environment. This image data is now ready for the training process.

3.2. Pre-processing

1. **Image Annotation**

In this research, dataset labeling was performed using Roboflow. The labeling was divided into 2 classes, namely "Focus" and "Not Focus". The annotated data only included the facial parts of student objects.

The annotation process can be observed in Figure 4.

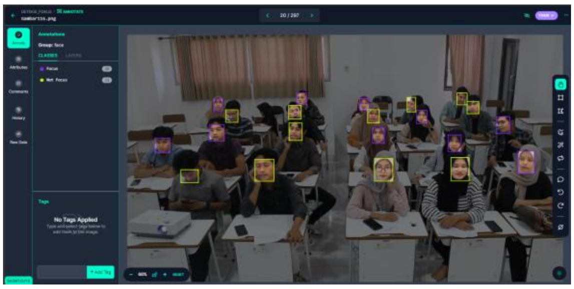
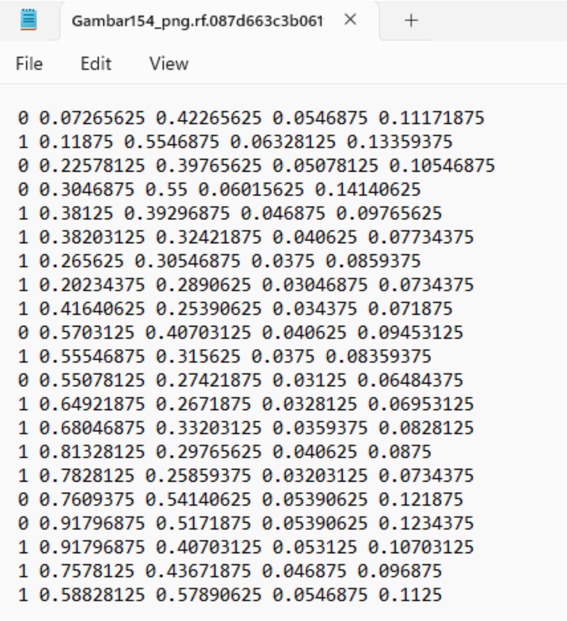


Figure 4. Image Annotation

From the results of the image annotation process, a file is generated containing the label coordinates of each image with the extension .txt. The contents of the .txt file can be seen in Figure 4.



```
0 0.07265625 0.42265625 0.0546875 0.11171875
1 0.11875 0.5546875 0.06328125 0.13359375
0 0.22578125 0.39765625 0.05078125 0.10546875
0 0.3046875 0.55 0.06015625 0.14140625
1 0.38125 0.39296875 0.046875 0.09765625
1 0.38203125 0.32421875 0.040625 0.07734375
1 0.265625 0.30546875 0.0375 0.0859375
1 0.20234375 0.2890625 0.03046875 0.0734375
1 0.41640625 0.25390625 0.034375 0.071875
0 0.5703125 0.40703125 0.040625 0.09453125
1 0.55546875 0.315625 0.0375 0.08359375
0 0.55078125 0.27421875 0.03125 0.06484375
1 0.64921875 0.2671875 0.0328125 0.06953125
1 0.68046875 0.33203125 0.0359375 0.0828125
1 0.81328125 0.29765625 0.040625 0.0875
1 0.7828125 0.25859375 0.03203125 0.0734375
0 0.7609375 0.54140625 0.05390625 0.121875
0 0.91796875 0.5171875 0.05390625 0.1234375
1 0.91796875 0.40703125 0.053125 0.10703125
1 0.7578125 0.43671875 0.046875 0.096875
1 0.58828125 0.57890625 0.0546875 0.1125
```

Figure 5. Content of .txt File

Figure 5. shows the contents of the .txt file generated from the image annotation process. The first column represents the type of object, in this study there are only "Focus" and "Not Focus" objects. The next four columns include the position information of the object, namely x, y, w, and h. Each image corresponds to one .txt file that can contain multiple object categories. The x and y coordinates are the center of the target, while w and h represent the width and height of the target respectively. All these coordinates undergo normalization, with x and w using the width of the original diagram, while y and h using the height of the original diagram.

1. **Split Data**
   In this research, data splitting is done using tools on Roboflow with a ratio of 70%: 20% : 10%. From a total of 297 images, the data division results obtained are 208 training data images, 59 validation data images, and 30 testing data images.
2. **Resize**
   Furthermore, resizing the image is done to change the image size from 1920×1088 pixels to 640×640 pixels, the goal is to standardize the image size and adjust it to the needs of the model [9].

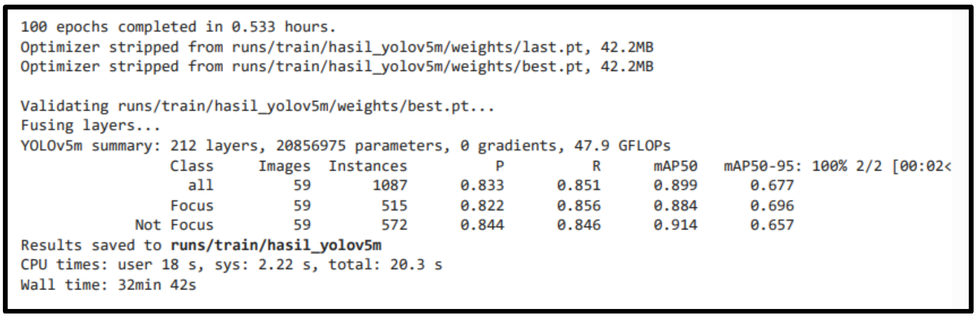
3.3. Augmentation

In this research, the image augmentation process carried out in the form of a horizontal flip. And augmented 3 times resulting in 713 images. The amount of data that has been annotated is 5,399. Where the "Focus" class has 2,904 images, and the "Not Focus" class has 2,495 images. An example of a data image that has been augmented is in Figure 5.

Figure 6. Horizontally Flipped Data

3.4. Trained YOLOV5 Model

In this study, the YOLOV5 model was trained using 50 and 100 epochs, 16 batches, and 640x640 pixels.



```
100 epochs completed in 0.533 hours.
Optimizer stripped from runs/train/hasil_yolov5m/weights/last.pt, 42.2MB
Optimizer stripped from runs/train/hasil_yolov5m/weights/best.pt, 42.2MB

Validating runs/train/hasil_yolov5m/weights/best.pt...
Fusing layers...
YOLOv5m summary: 212 layers, 20856975 parameters, 0 gradients, 47.9 GFLOPs
                 Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 2/2 [00:02<
                   all         59       1087      0.833      0.851      0.899      0.677
                 Focus         59        515      0.822      0.856      0.884      0.696
             Not Focus         59        572      0.844      0.846      0.914      0.657
Results saved to runs/train/hasil_yolov5m
CPU times: user 18 s, sys: 2.22 s, total: 20.3 s
Wall time: 32min 42s
```

Figure 7. Training YOLOV5

DOI: https://doi.org/10.52436/1.jutif.2024.5.5.1977
p-ISSN: 2723-3863
e-ISSN: 2723-3871

3.5. Evaluation

1. Test Result with 50 Epoch

Based on Table 2. The computational load is calculated based on the parameters, GFLOPs, and time required for the training process. Judging from the parameters and GFLOPs, YOLOV5x has the most parameters and GFLOPs and YOLOV5s has the least parameters and GFLOPs. This of course also affects the time required during the training process, the more parameters that must be set, the longer it takes to train the model. So YOLOV5x also has more time for the train-ing process compared to other types of architectures. In general, Table 3. Shows that YOLOV5x has the highest computational load and YOLOV5s has the lightest computational load among all models.

Table 2. Computation Comparison of YOLOV5 50 Epoch

Model	Size	Params (M)	GFLOPs	Training Time
YOLOV5n	640	1.76	4.1	13min 29s
YOLOV5s	640	7.01	15.8	13min 57s
YOLOV5m	640	20.8	47.9	19min 22s
YOLOV5l	640	46.1	107.7	25min 45s
YOLOV5x	640	86.1	203.8	45min 55s

Table 3. Shows the performance of each of the five types of YOLOV5 based on precision, recall and mAP@50 values with 50 epoch trials. Based on these values, YOLOV5m has the highest precision value than the other architectures. As for the recall value, YOLOV5n has the highest recall value. And for the mAP@50 value, YOLOV5m has the highest value among other architectures. Where the high mAP value indicates that the model has a good balance between precision and recall at the 0.5 confidence level. Table 4. Shows that YOLOV5m has a more optimal performance than other architectures.

Table 3. YOLOV5 Evaluation Metrics 50 Epoch

Model	Precision	Recall	mAP@50
YOLOV5n	49.9%	**93.1%**	61.8%
YOLOV5s	57.9%	82.2%	71.1%
YOLOV5m	**70.6%**	82%	**81.6%**
YOLOV5l	69.7%	81.5%	78.6%
YOLOV5x	70%	81.6%	80.6%

2. Test Result with 100 Epoch

Table 4. Shows the computational load performed on the 100 Epoch experiment, resulting in the YOLOV5x model having the largest computational load compared to other architectures. Based on Table 2. The parameter values and GFLOPs of each YOLOV5 architecture for the 50 and 100 Epoch experiments there is no difference, only the difference lies in the time required during the training process. This is possible because the 100 epochs experiment has additional iterations so it takes longer than the 50 Epoch experiment.

Table 4. Computation Comparison of YOLOV5 100 Epoch

Model	Size	Params (M)	GFLOPs	Training Time
YOLOV5n	640	1.76	4.1	24min 6s
YOLOV5s	640	7.01	15.8	25min 31s
YOLOV5m	640	20.8	47.9	32min 42s
YOLOV5l	640	46.1	107.7	48min 5s
YOLOV5x	640	86.1	203.8	1h 34min 12s

Table 5. Shows the performance of the five YOLOV5 architectures with 100 epoch trials. Where YOLOV5x has higher precision and mAP@50 values compared to the other architectures. As for the recall value, YOLOV5l has the highest recall value. However, when viewed from the computational load and performance, YOLOV5m is more optimal than other types of architecture, because YOLOV5m has a computational load that is not too large and not too small, and its precision and recall values have a good balance.

Table 5. YOLOV5 Evaluation Metrics 100 Epoch

Model	Precision	Recall	mAP@50
YOLOV5n	74.8%	79%	83.2%
YOLOV5s	81.4%	82.9%	89.3%
YOLOV5m	**83.3%**	**85.1%**	**89.9%**
YOLOV5l	79.6%	88.9%	90.5%
YOLOV5x	83.9%	87.4%	**91%**

3. Detection Result

Based on the experiments conducted, it was found that YOLOV5m has the most optimal performance between other architectures. Therefore, the test was conducted with the YOLOV5m model. The graph below shows the test results using the

YOLOV5m architecture with 100 epochs, 16 batches, and 640 × 640 pixels.

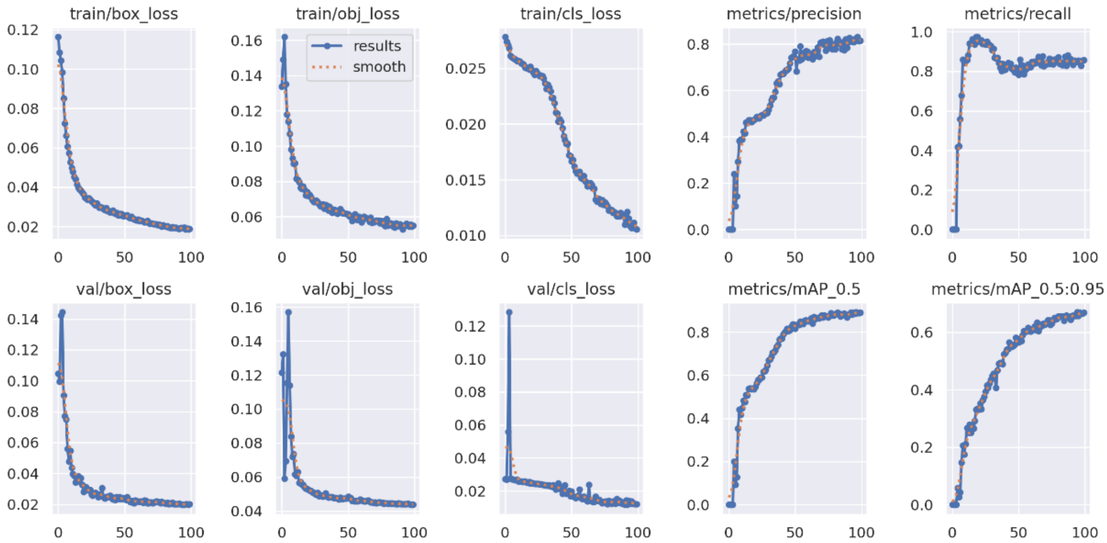


Figure 8. Graph of Tests Results with YOLOV5m

Figure 8. shows the graph of training and validation results from the student focus dataset using the YOLOV5m architecture. The train/box_loss value decreases linearly from the first epoch to the 100th epoch from 0.12 to 0.02, indicating that the model is increasingly accurate in determining the location of objects in the training image. Similarly, the val/box_loss value decreases from 0.14 to 0.02, which means that the object detection model is learning well and is increasingly accurate in predicting the location of objects in the image. This also indicates that the model is not overfitting. While the value of train/obj_loss has decreased from 0.16 to below 0.06, as well as val/obj_loss which decreased from 0.16 to 0.04. This shows that the model has successfully learned and generalized well on the validation data. Then the train/cls_loss value drops quite steadily from 0.030 to 0.010, indicating that the object detection model has learned well in classifying objects detected in the training image. In addition, the val/cls_loss value also decreased from 0.12 to below 0.02, indicating that the object detection model was able to accurately classify objects in the validation data. Then there is precision which gets the best results with a value of up to 0.833, while recall reaches 0.851. In addition, there is a value of mAP@0.5 that exceeds 0.8, and the value of mAP@0.95 reaches a result above 0.6.

The following is a test conducted using image data taken in the classroom during the learning process. The image data used is data that has been divided into test data categories which then the system will detect students who are focused and unfocused based on the student's facial expressions.

Figure 9. Shows the test results using the YOLOV5m architecture with 100 epochs. Based on the experimental results, YOLOV5m successfully detects objects with 2 classes, namely the focus and non-focus classes of students based on their facial expressions.

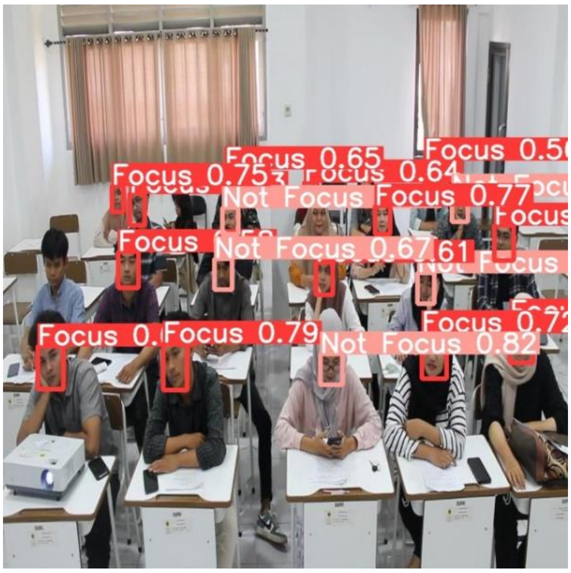


Figure 9. Test Result with YOLOV5m

4. DISCUSSION

Similar research has been conducted by [10] for recognizing student behavior by identifying 2 categories, namely actions and emotions/facial expressions. Where in the action category, there are 9 classes, one of which is the focus class. By comparing the five architectures of YOLOV5, the experiment results from that study obtained a mAP value of 76%. Based on the conducted tests in this study, comparing the five types of YOLOv5 architectures yielded an optimal mAP value of 89.9%. This proves that the YOLOV5 algorithm with 100 epochs experiment, hyperparameters including 16 batches, a threshold value of 0.5, and image size of 640x640 pixels, is capable of detecting student focus objects well. Although this research yielded a high mAP value, it is important to note that further research is needed to increase the dataset size. The interpretation of this research is that by utilizing an optimized approach with the YOLOV5 algorithm, this study successfully enhanced the capability to detect student focus objects within the classroom. This demonstrates advancements in technology development for object recognition in an educational context, which can have a positive impact on monitoring and interaction within the classroom. With improved technology for detecting student focus, educators can more effectively monitor the level of student engagement in learning.

5. CONCLUSION

This research uses the YOLOV5 method to detect student focus, using an image dataset of 297 images. Based on the analysis of computational load and performance of the five YOLOV5 architectures (YOLOV5n, YOLOV5s, YOLOV5m, YOLOV5l, YOLOV5x) in detecting student focus, two experiments were conducted using 50 epochs and 100 epochs. YOLOV5x has the largest computational load, because it has many parameters that affect the training time. YOLOV5x produced the highest mAP value in the 100 epoch experiment. However, YOLOV5m has the most optimal performance compared to other architectures with 100 epoch trials, because there is a balance between precision and recall values. The resulting precision value is 83.3%, recall 85.1%, and mAP@50 89.9%. The optimal object detection capability of YOLOV5m, which is seen from the balance of precision and recall values, shows that this architecture has the ability to recognize students who are focused with high mAP. As for efforts to improve the development results in future research, there are several suggestions that might be done in order to get maximum results, namely by increasing the number of datasets. And consideration of lighting in data collection is also needed to maximize model performance.

ACKNOWLEDMENT

This research is supported by several parties. Therefore, the author would like to express gratitude to the supervising lecturer from the Intelligent Systems Laboratory, Department of Informatics Engineering, Mataram University, who has provided input and advice during the author's research and journal work. Then, thanks to both parents who always provide encouragement and prayers to the author. Finally, thanks to the comrades who have always accompanied the author throughout the journey.

REFERENCES

[1] M. R. Hamandia and Z. Jannati, “Penerapan komunikasi nonverbal: Sebuah alternatif dalam peningkatan perhatian mahasiswa pada proses pembelajaran,” *Jurnal Komunikasi Islam dan Kehumasan (JKPI)*, vol. 4, no. 1, pp. 75–89, 2020.

[2] F. R. Nasution, N. M. Adlika, and B. Tampubolon, “Analisis Perhatian Dan Keterlibatan Siswa Pada Pembelajaran Secara Daring,” *Jurnal Pendidikan Sosiologi dan Humaniora*, vol. 13, no. 1, p. 91, Feb. 2022, doi: 10.26418/j-psh.v13i1.52321.

[3] “A Novel Architecture for Student’s attention detection in classroom based on Facial and Body expressions,” *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 5, pp. 7357–7366, Oct. 2020, doi: 10.30534/ijatcse/2020/68952020.

[4] W. Jia *et al.*, “Real-time automatic helmet detection of motorcyclists in urban traffic using improved YOLOv5 detector,” *IET Image Process*, vol. 15, no. 14, pp. 3623–3637, Dec. 2021, doi: 10.1049/ipr2.12295.

[5] J. N. Mindoro, N. U. Pilueta, Y. D. Austria, L. Lolong Lacatan, and R. M. Dellosa, “Capturing Students’ Attention through Visible Behavior: A Prediction Utilizing YOLOv3 Approach,” in *2020 11th IEEE Control and System Graduate Research Colloquium, ICSGRC 2020 - Proceedings*, Institute of Electrical and Electronics Engineers Inc., Aug. 2020, pp. 328–333. doi: 10.1109/ICSGRC49013.2020.9232659.

[6] M. Villa, M. Gofman, S. Mitra, A. Almadan, A. Krishnan, and A. Rattani, “A Survey of Biometric and Machine Learning Methods for Tracking Students’ Attention and Engagement,” in *Proceedings - 19th IEEE International Conference on Machine Learning and Applications, ICMLA 2020*, Institute of Electrical and Electronics Engineers Inc., Dec. 2020, pp. 948–955. doi: 10.1109/ICMLA51294.2020.00154.

[7] M. Sarosa and N. Muna, “Implementasi Algoritma You Only Look Once (YOLO) Untuk Deteksi Korban Bencana Alam,” vol. 8, no. 4, 2021, doi: 10.25126/jtiik.202184407.

[8] D. I. Mulyana and M. A. Rofik, “Implementasi Deteksi Real Time Klasifikasi Jenis Kendaraan Di Indonesia Menggunakan Metode YOLOV5,” *Jurnal Pendidikan Tambusai*, vol. 6, no. 3, pp. 13971–13982, 2022.

[9] H. Dawami, E. Rachmawati, and M. D. Sulistiyo, “Deteksi Penggunaan Masker

Wajah Menggunakan YOLOv5," *eProceedings of Engineering*, vol. 10, no. 2, 2023.

[10] Z. Trabelsi, F. Alnajjar, M. M. A. Parambil, M. Gochoo, and L. Ali, "Real-Time Attention Monitoring System for Classroom: A Deep Learning Approach for Student's Behavior Recognition," *Big Data and Cognitive Computing*, vol. 7, no. 1, Mar. 2023, doi: 10.3390/bdcc7010048.

[11] L. Susanti, N. K. Daulay, and B. Intan, "Sistem Absensi Mahasiswa Berbasis Pengenalan Wajah Menggunakan Algoritma YOLOv5," *JURIKOM (Jurnal Riset Komputer)*, vol. 10, no. 2, p. 640, Apr. 2023, doi: 10.30865/jurikom.v10i2.6032.

[12] 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, 2023.

[13] Y. Liu, B. Lu, J. Peng, and Z. Zhang, "Research on the use of YOLOv5 object detection algorithm in mask wearing recognition," *World Scientific Research Journal*, vol. 6, no. 11, pp. 276–284, 2020.

[14] N. Hidayat, S. Wahyudi, A. Aufa Diaz, I. Teknologi Sepuluh Nopember, and K. Keputih-Sukolilo, "Pengenalan Individu Melalui Identifikasi Wajah Menggunakan Metode You Only Look Once (YOLOv5) (Individual Recognition Through Face Identification Based On You Only Look Once (YOLOv5) Method)." [Online]. Available: https://magestic.unej.ac.id/

[15] B. A. Septyanto, S. A. Wibowo, and C. Setianingsih, "Implementasi Face Recognition Berbasis Deep Neural Network Sebagai Sistem Kendali Pada Quadcopter," *eProceedings of Engineering*, vol. 9, no. 6, 2023.

[16] A. Imran, C. Setianingsih, and R. E. Saputra, "Deteksi Pelanggaran Pada Bahu Jalan Tol Dengan Intelligent Transportation System Menggunakan Algoritma Yolov5," *eProceedings of Engineering*, vol. 10, no. 5, 2023.

[17] G. C. Utami, C. R. Widiawati, and P. Subarkah, "Detection of Indonesian Food to Estimate Nutritional Information Using YOLOv5," *Teknika*, vol. 12, no. 2, pp. 158–165, Jun. 2023, doi: 10.34148/teknika.v12i2.636.

[18] L. Suroiyah, Y. Rahmawati, and R. Dijaya, "Facemask Detection Using YOLO V5," *Jurnal Teknik Informatika (Jutif)*, vol. 4, no. 6, pp. 1277–1286, Dec. 2023, doi: 10.52436/1.jutif.2023.4.6.1043.

[19] L. S. Riva and J. Jayanta, "Deteksi Penyakit Tanaman Cabai Menggunakan Algoritma YOLOv5 Dengan Variasi Pembagian Data," *Jurnal Informatika: Jurnal Pengembangan IT*, vol. 8, no. 3, pp. 248–254, 2023.

[20] Y. Zhao and S. Geng, "Face occlusion detection algorithm based on yolov5," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Sep. 2021. doi: 10.1088/1742-6596/2031/1/012053.

[21] I. P. Sary, S. Andromeda, and E. U. Armin, "Performance Comparison of YOLOv5 and YOLOv8 Architectures in Human Detection using Aerial Images," *Ultima Computing: Jurnal Sistem Komputer*, vol. 15, no. 1, pp. 8–13, 2023.

[22] Z. J. Zheng, G. J. Liang, H. Bin Luo, and H. C. Yin, "Attention assessment based on multi-view classroom behaviour recognition," *IET Computer Vision*, 2022, doi: 10.1049/cvi2.12146.

[23] E. Casas, L. Ramos, E. Bendek, and F. Rivas-Echeverria, "Assessing the Effectiveness of YOLO Architectures for Smoke and Wildfire Detection," *IEEE Access*, vol. 11, pp. 96554–96583, 2023, doi: 10.1109/ACCESS.2023.3312217.

[24] U. Nepal and H. Eslamiat, "Comparing YOLOv3, YOLOv4 and YOLOv5 for Autonomous Landing Spot Detection in Faulty UAVs," *Sensors*, vol. 22, no. 2, Jan. 2022, doi: 10.3390/s22020464.

[25] K. Khairunnas, E. M. Yuniarno, and A. Zaini, "Pembuatan Modul Deteksi Objek Manusia Menggunakan Metode YOLO untuk Mobile Robot," *Jurnal Teknik ITS*, vol. 10, no. 1, pp. A50–A55, 2021.