Efficient object detection for augmented reality based english learning with YOLOv8 optimization

Arya Krisna Putra[1], Fiqri Ramadhan Tambunan[1], Samson Ndruru[1], Andry Chowanda[2]
[1]Departement of Computer Science, BINUS Graduate Program-Master of Computer Science, Jakarta, Indonesia
[2]Department of Computer Science, School of Computer Science, Bina Nusantara University, Jakarta, Indonesia

Article Info

Article history:

Received Feb 9, 2025
Revised Mar 24, 2025
Accepted Jul 2, 2025

Keywords:

Augmented reality
Educational technology
Ghostnet
Image processing
Machine learning
Object detection
YOLO

ABSTRACT

This study develops a mobile-based augmented reality (AR) application with machine learning for elementary school students to enhance basic English vocabulary learning. The application integrates an optimized YOLOv8 object detection model, designed to recognize 20 common classroom objects in real-time. The model optimization involves replacing standard Conv layers with GhostConv and the C2f block with the C2fCIB block that has significantly improved computational efficiency. Evaluation results show the optimized model reduces the parameters by 22.003% and decreases the file size from 6.2 MB to 4.9 MB. The model performance improved by achieving precision of 83.7%, recall of 73.5% and a mean Average Precision (mAP) of 81.4%. The model was integrated into the Unity platform via the Barracuda library, enabling real-time detection and interactive display of 3D objects. This aplication also complete with English text, translations, example sentences also audio pronunciation. 3D objects representing classroom vocabulary were specifically created to support AR-based learning. Performance testing on a Samsung A14 showed an improved frame rate of 6–12 FPS compared to the original model's 5–10 FPS. These results demonstrate that the optimized YOLO model effectively integrates with AR technology, creating a more interactive and enjoyable vocabulary learning experience.

Corresponding Author:

Arya Krisna Putra
Departement of Computer Science, BINUS Graduate Program-Master of Computer Science
Bina Nusantara University
Jakarta, Indonesia
Email: arya.putra006@binus.ac.id

1. INTRODUCTION

In this era, technology in education has reached in very innovative level that allows learning to be interactive such as artificial intelligence (AI), virtual reality (VR) also augmented reality (AR). AR approach is gaining interest due to its ability to combine digital elements with the real world, enabling more direct and interactive experiences [1], [2]. In primary schools, English language is essential to build foundational language skills early, that enabling children to access global knowledge, improve literacy, cultural awareness, and communication confidence [3], [4]. Elementary school students find that traditional English teaching methods are boring and lacking interactivity [5]. With that, this research will combine deep learning approach integrated into an AR application to support English vocabulary in one application. There is a lot of deep learning models but in this research, the model used is you only look once (YOLOv8). YOLOv8 is known for its speed and accuracy in real-time object recognition [6]. Computational efficiency in the mobile application will be improved by using GhostNet, which generates more features with fewer operations making it suitable for devices with limited resources and channel interaction block (CIB) to efficiently extract key features and improve recognition accuracy without added complexity [7], [8].

AR based mobile application specifically designed to help students learn basic English vocabulary by interacting with 3D objects that appear on their smartphone screens. Some features such as graphical visualizations and audio pronunciations for each word are designed for students to learn in a more dynamic and immersive way compared to traditional methods [9]. This research focuses on integrating deep learning technology using the YOLOv8 model, which enables real-time object recognition with high speed and accuracy [10], [11]. This research aims to develop a more effective and enjoyable learning process and help elementary school students build strong English skills early on through an innovative and customized deep learning technology approach.

Recent literature highlights opportunities to integrate deep learning models with AR to enhance interactive learning. Saikumar *et al.* [12] successfully integrated the MobileNet model into AR applications, achieving 85% accuracy and efficient real-time performance when running it on Augmented Reality. Meanwhile, YOLO (You Only Look Once) models have demonstrated superior performance in real-time detection compared to other approaches, as seen in YOLOv3 and YOLOv4 studies, which showed high accuracy but faced limitations in mobile computational efficiency [13], [14]. In the study of Lysakows *et al.* [15] the YOLOv8 model was integrated with Unity with a library called barracuda, resulting in 60% accuracy on the COCO Dataset, motivated from there we want to explore the use of YOLOv8 more deeply by changing some of its architecture.

To address these gaps, researchers explored enhancements using GhostNet and Channel Interaction Block (CIB). Zeng *et al.* [16] and Ferdi *et al.* [17], combining YOLOv11 with GhostNet obtained significant results which had the advantages of better efficiency and smaller model size in object detection. In addition to taking an approach to the backbone architecture, several studies have stated that YOLO performance improvements can be improved by replacing several layers such as C2f Blocks which are replaced with C2fCIB as in the research conducted by Akhmedov *et al.* [18] and Li *et al.* [19] using CIB gets increased efficiency and without sacrificing accuracy, and produces lighter computation.

Motivated by these insights, this research explicitly addresses the identified gaps by optimizing YOLOv8 architecture through integration with GhostNet and C2fCIB, aiming to balance computational efficiency with high accuracy. Our contribution lies in developing a lightweight and efficient deep learning model that can be seamlessly implemented into an AR-based application, enabling real-time, interactive, and immersive English vocabulary learning for elementary school students.

2. METHOD

This research aims to develop an augmented reality (AR) application for mobile devices to assist elementary school students in learning basic english vocabulary. This application will then be integrated using deep learning object detection model which is YOLO to detect 20 common objects in classroom. The experiments will be conducted to improve the model's performance. This research will proceed in two main steps. The first step is to build a modified YOLO to ensure the model is well prepared for an effective object detection. Then the second step involves integrating the YOLO model with Unity which will lead to the development of AR application. Figure 1 shows the general flow of our research.

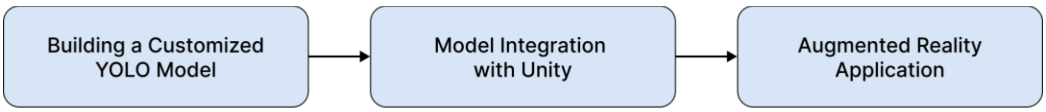


Figure 1. The general flow of research

2.1. Building a modified YOLO model

The development of this modified YOLO model will focus on being able to detect 20 common objects found in the classroom, such as tables, chairs, blackboards, bookshelves, clocks, wall magazines, trash cans, erasers, sharpeners, pens, books, rulers, scissors, fans, laptops, bags, remote controls, pants, shoes, and hats. This process will include several main steps, starting from data preparation consisting of data collection, data annotation, data pre-processing, and data splitting. The next step is building our proposed model architecture, followed by model training and model evaluation. Figure 2 below shows the development flow of the our proposed model.

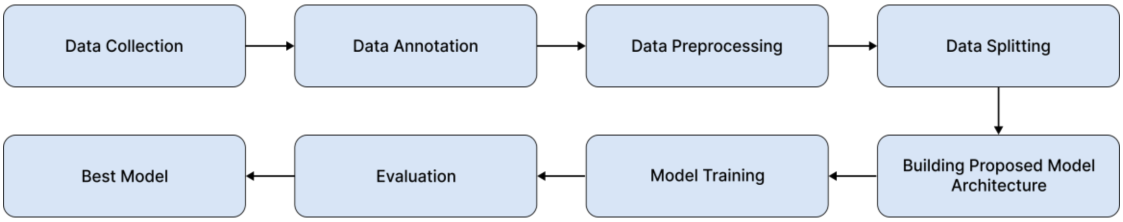


Figure 2. Development flow of the modified YOLO model

2.1.1. Data preparation

In this study, when preparing data to build an object detection model consists of four main stages, starting from data collection, data annotation, data preprocessing, and data splitting. In the data collection stage, we collected 4,000 images using manual methods such as photography and image data collection using the help of a web search engine with 20 different classes. Each class consists of 200 images and the images collected only contain images that are often found in the classroom, to ensure that the image data is relevant to the AR application. After that, the collected images are labeled for annotation using the computer vision annotation tool (CVAT), where bounding boxes are created around the desired objects and assigned to their respective classes, to ensure accurate identification and classification [20]. In the data preprocessing stage, all images are resized to 224x224 pixels as the requirements of the model used, and the bounding box coordinates are recalculated to maintain the exact location of the object. The final step is to split the dataset into training, validation, and testing, with 80% allocated for training data and the remaining 20% divided into validation data and testing data taken as 20% of the validation data

2.1.2. Building proposed model architecture

In this step, we conducted experiments on the YOLOv8 model by replacing several layers in the backbone and neck. First, we replaced the standard convolutional (Conv) layer with GhostConv, this modification aims to reduce the number of parameters and computational cost without sacrificing model performance [21]. This is because GhostConv block works by generating more feature maps from cheap operations, called ghost features, making it suitable for devices with limited data resources [17], [22]. Furthermore, we made modifications by replacing the YOLOv8 C2f block with C2fCIB which aims to improve the efficiency of feature extraction, with the aim of maintaining detection accuracy even though using lighter computation [8], [23].

The purpose of making these two modifications is to develop an object detection model that ensures a balance between accuracy and speed. Additionaly, these improvements ensure that the model can work optimally on mobile devices that have limited computing resources. The architecture of our proposed model is shown in Figure 3, where changes are shown in the square-shaped blocks.

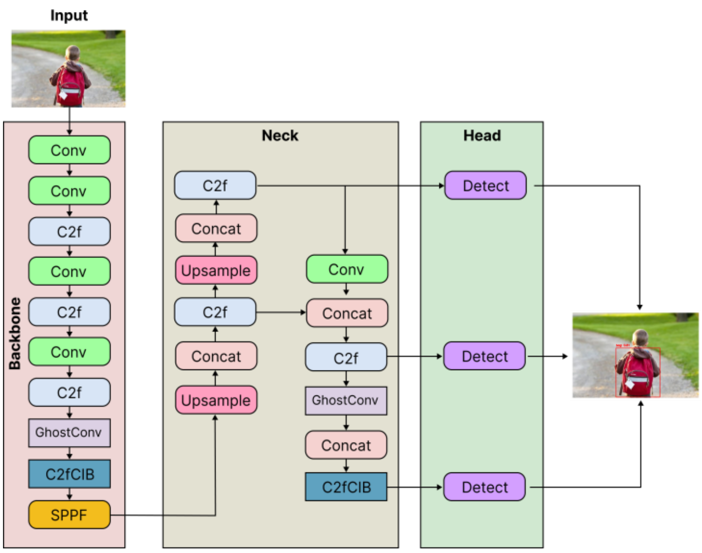


Figure 3. Proposed model architecture

2.1.3. Model training

At this stage, YOLO model training uses previously defined hyperparameter values, while other parameters use default values provided by the framework. The values of the hyperparameters used in training this model are presented below in Table 1.

Table 1. Hyperparameter setting for experiment

Hyperparameter	Value
Epochs	100
Batch Size	16
Learning Rate	0.001
Image Size	224 x 224
Optimizer	AdamW

2.1.4. Model evaluation

At the performance evaluation stage of the developed YOLO model, a comprehensive evaluation was carried out using various metrics to ensure the accuracy and efficiency of the model. The most important evaluation is the mean average precision (mAP), which indicates the overall accuracy of the model in detecting objects [24]. In addition, other evaluation metrics such as recall and precision were analyzed to assess the model's ability to detect objects while minimizing errors during detection [25]. The number of parameters of the model was also evaluated to provide insight into the complexity of the model and the need for computing resources [26]. The combination of these metrics will certainly provide a detailed picture of the model's performance in calculating accuracy, speed, and resource efficiency.

2.2. Model integration with Unity

After the best YOLO model was developed through training and performance evaluation, the next step was to incorporate it into a mobile application built using Unity. Unity was chosen because of its ability to create interactive applications, support real-time data transmission, and facilitate the integration of 3D elements [27]. In addition, Unity allows the implementation of augmented reality (AR) technology to combine object detection results with virtual visualization elements directly in the real environment, which enhances the user experience [28].

The application starts by turning on the camera for taking a picture of the surrounding environment. Then, the YOLO model processes the image to find the objects present. Once the application detects an object, a button appears that the user can press to start an AR visualization based on that object. The application displays a 3D object of the detected object, which can be moved to increase user interactivity.
In addition, text will appear with the name of the object in Indonesian and its meaning in English, as well as examples of the use of these words in English sentences. A more interactive and immersive learning experience occurs when users listen to example sentences by playing audio. This method ensures a smooth integration between object detection models, AR visualizations, and in-app learning features. Figure 4 shows the complete application flowchart.

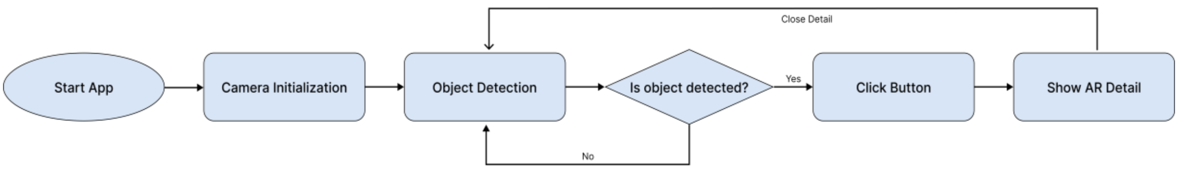


Figure 4. Application flow diagram

3. RESULT AND DISCUSSION

In this section, the findings from the experiments conducted during the research are presented. Our key findings indicate that replacing standard Conv layers with GhostConv and C2f blocks with C2fCIB improves efficiency significantly while maintaining detection accuracy. This includes the detailed analysis of the performance of modified YOLO model also the results of integrating the model into a Unity based application. All findings are discussed to give a comprehensive understanding and system's capabilities in order to receive the research objectives.

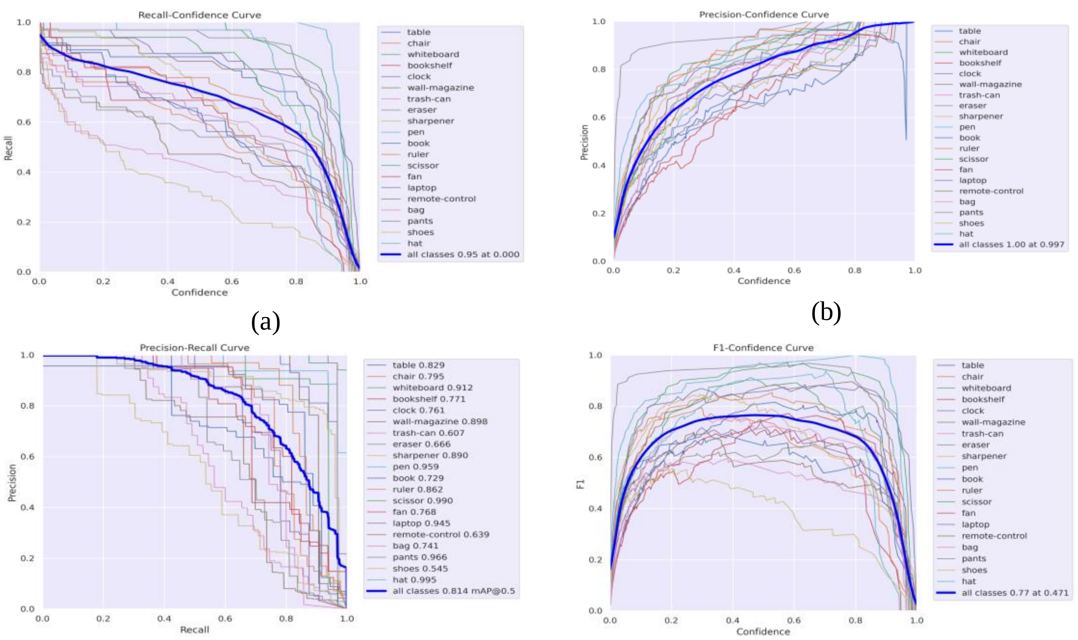
3.1. Performance of modified YOLO

This section presents the performance analysis of the modified YOLO model. Focusing on its effectiveness and efficiency to detect an object in a classroom. The experiment setup involves the use of Google Colab with a Tesla GPU to ensure the optimal processing capabilities. Dataset used for training and evaluation consisted of 4,000 images of common objects in the classroom, which is underwent certain the pre-processing steps to improve the performance of the model created. We evaluate the modified YOLO model using several key metrics such as precision, recall and mean Average Precision (mAP). We compare our proposed model with the base model YOLOv8n, as well as with the YOLOv8n + GhostNet and YOLOv8n + C2fCIB models to evaluate the impact of using the GhostNet and C2fCIB blocks on the model performance. The results of this evaluation are summarized in this Table 2.

As we can see in the evaluation results, the proposed model has a total of 2,351,388 parameters which is 22.003% fewer compared to the base YOLOv8n model with 3,014,748 parameters. Although the proposed model have more compact architecture, but the proposed model has successfully improves performance metrics with precision, recall, and mAP compared to the baseline model. The file size of the proposed model also decreased to 4.9 MB, 1.3 MB smaller than the baseline model's file size of 6.2 MB. While YOLOv8n + C2fCIB shows higher accuracy with advantages in recall and mAP, the proposed model excels in the lightweight category. With a smaller number of parameters, approximately 8.43% smaller, and a smaller file size about 7.55% of YOLOv8n + C2fCIB, the proposed model becomes a more efficient choice for resource-constrained devices. The reduction in model size and parameter count directly translates into better performance on mobile devices, enhancing interactivity and responsiveness in educational settings. Figure 5 presents the performance evaluation of the proposed method. Specifically, Figure 5(a) shows the recall-confidence curve, Figure 5(b) illustrates the precision-confidence curve, Figure 5(c) shows the precision-recall curve, and Figure 5(d) displays the F1-confidence curve. Meanwhile, Figure 6 shows the comparison of training and validation loss during model training, where Figure 6(a) represents the bounding box loss (box_loss), Figure 6(b) illustrates the classification loss (cls_loss), and Figure 6(c) shows the distribution focal loss (dfl_loss).

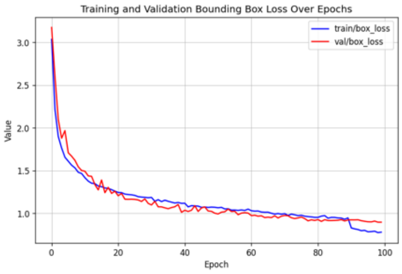
Table 2. YOLO model performance comparison

Method	Number of Parameters	Precision (%)	Recall (%)	mAP (%)	File Size (MB)	Training time (hours)
YOLOv8n (Based Model)	3,014,748	82.9	69.7	80.2	6.2	0.788
YOLOv8n + GhostNet	2,798,364	82.9	73.4	81	5.8	**0.757**
YOLOv8n + C2fCIB	2,567,772	82.3	**76.7**	**82.9**	5.3	0.798
YOLOv8n + GhostNet + C2fCIB (Proposed Model)	**2,351,388**	**83.7**	73.5	81.4	**4.9**	0.830

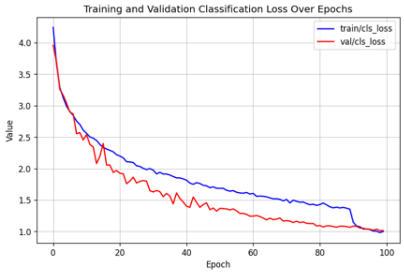


(a) (b) (c) (d)

Figure 5. Performance of proposed method: (a) recall-confidence curve, (b) precision-confidence curve, (c) precision-recall curve, and (d) F1-confidence curve



(a)



(b)

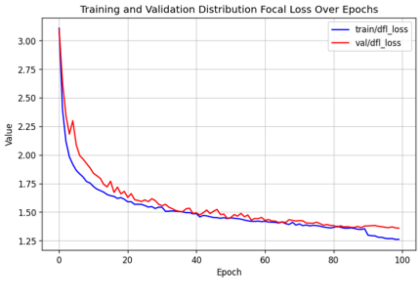
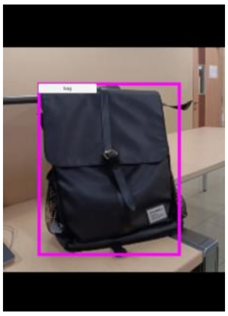


Figure 6. Comparison of training and validation loss during model training: (a) bounding box loss (box_loss), (b) classification loss (cls_loss), and (c) distribution focal loss (dfl_loss)

Although the proposed model showed a reduction in the number of parameters and the size of the model weights, the training time was slightly longer. Based on research conducted by Shah et al. [26], the length of training time is not always directly proportional to the number of parameters or the number of layers present in the model. For example, convolutional layers usually require more operations than dense layers, although the time per epoch for dense layers tends to be greater than that for convolutional layers. In other words, a model that has a smaller number of parameters or a simpler layer structure does not necessarily guarantee a shorter training time.

3.2. Model integration with Unity

Once the best YOLO model has been developed, the next step is to integrate the model with Unity to create an augmented reality (AR) application that can recognize objects in real-time. This integration allows the application to capture and detect objects in the user's environment through camera input, and display relevant 3D elements based on the detected objects. Figure 7 shows the integrated application, where Figure 7(a) shows the object detection process and Figure 7(b) shows the details of the object using augmented reality.



(a)

(b)

Figure 7. Result of integration: (a) object detection with YOLO and (b) augmented reality integration

Our model has been successfully implemented on a Samsung A14 5G mobile device which performs quite well. The hardware specifications of the mobile device that contribute to the model performance are detailed in Table 3.

The test results show that the based model has a frame per second (FPS) of 5-10, while the proposed model achieves 6-12. With such FPS results, this application is capable of providing a smooth and responsive user experience, enhancing user interaction with the displayed AR elements. The Table 4 summarizes the FPS results among the models.

Table 3. Mobile device spesification

Spesification	Detail
OS	Android 14
Chipset	Mediatek Dimensity 700 (7 nm)
CPU	Octa-core (2x2.2 GHz Cortex-A76 & 6x2.0 GHz Cortex-A55)
GPU	Mali-G57 MC2

Table 4. Comparison of FPS for each model

Method	Frame per second (FPS)
YOLOv8n (based)	5-10
YOLOv8n + GhostNet	5-11
YOLOv8n + C2fCIB	5-11
YOLOv8n + GhostNet + C2fCIB (Proposed Model)	**6-12**

With this integration, the application can recognize objects in real-time. From camera input convert them into interactive visual representations of 3D objects. The application also includes text descriptions that explain the object's name in Indonesian and its meaning in English, as well as the examples how the word used in sentences. With this feature the users not only can see the detected objects but also understand the context and meaning of those objects. This application not only provides an engaging visual experience but also supports more effective and interactive learning process for users especially students.

4. CONCLUSION

In conclusion, our optimized YOLOv8 model significantly reduces computational complexity while maintaining accuracy, proving highly effective when integrated into AR-based mobile learning applications for elementary education. The development of this application integrates the modified YOLO object detection model with Unity to create a more interactive and immersive learning experience. Modifications to the YOLO model were made by replacing the Conv layer with GhostConv and the C2f block with the C2fCIB block in the YOLOv8 architecture. These changes succeeded in reducing the number of model parameters by 22.003% (from 3,014,748 to 2,351,388 parameters) and reducing the model file size from 6.2 MB to 4.9 MB from the based model. Despite a slight increase in training time, the proposed model showed improved performance in terms of precision by 83.7%, recall by 73.5%, and mAP by 81.4% compared to the basic YOLOv8n model.

The integration model with Unity resulted in an AR application that was able to detect 20 common objects in the classroom in real-time. This application not only displays a 3D representation of the detected object, but also provides a text description in Indonesian and its translation in English, complete with example sentences and audio to reinforce learning. Testing on a Samsung A14 mobile device shows that the proposed model performs better with a frame rate of 6-12 FPS, compared to the base model which only reaches 5-10 FPS. This proves that the proposed model is more efficient and responsive in a mobile device environment. Future research should consider incorporating additional optimization techniques such as quantization or pruning to further enhance the model's efficiency on resource-constrained devices.

Overall, the developed application has succeeded in increasing the effectiveness and interactivity in the English learning process for elementary school students. The integration of optimized YOLO architecture into an AR-based educational tool demonstrates that deep learning can significantly enrich English vocabulary learning by making it interactive, effective, and highly accessible. The optimized YOLO model has proven to be lighter and more efficient, and can be effectively integrated with the AR platform to provide a fun and educational learning experience.

ACKNOWLEDGMENTS

The authors declare that no acknowledgments are applicable. All work was conducted solely by the authors.

FUNDING INFORMATION

The authors state that there is no funding involved in the conduct of this research.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Arya Krisna Putra	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	
Fiqri Ramadhan Tambunan	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			
Samson Ndruru	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			
Andry Chowanda				✓								✓		

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

Open Data is available at: https://www.kaggle.com/datasets/aryakrisnaputra/objects-in-the-classroom.

REFERENCES

[1] G. Lampropoulos, E. Keramopoulos, K. Diamantaras, and G. Evangelidis, “Augmented reality and gamification in education: a systematic literature review of research, applications, and empirical studies,” *Applied Sciences (Switzerland)*, vol. 12, no. 13, p. 6809, 2022, doi: 10.3390/app12136809.

[2] T. A. Syed *et al.*, “In-depth review of augmented reality: tracking technologies, development tools, AR displays, collaborative AR, and security concerns,” *Sensors*, vol. 23, no. 1, p. 146, Jan. 2023, doi: 10.3390/s23010146.

[3] M. Hu and M. Chen, “Stimulating ideological education in elementary school english teaching with interest,” *Journal of Education Teaching and Social Studies*, vol. 4, no. 4, p. p62, Nov. 2022, doi: 10.22158/jetss.v4n4p62.

[4] T. Kustini, “Teacher's perspective: english in elementary school, is IT Necessary or Not?,” *Journal of English Language Learning (JELL)*, vol. 5, no. 2, pp. 119–123, 2021, doi: 10.31949/jell.v5i2.3420.

[5] R. A. Kinanti and A. H. Hernawan, “Teacher strategies for creating interesting and dynamic learning,” *Scaffolding: Jurnal Pendidikan Islam dan Multikulturalisme*, vol. 4, no. 3, pp. 679–689, 2022, doi: 10.37680/scaffolding.v4i3.4141.

[6] M. Hussain, “YOLO-v1 to YOLO-v8, the rise of YOLO and its complementary nature toward digital manufacturing and industrial defect detection,” *Machines*, vol. 11, no. 7, p. 667, Jul. 2023, doi: 10.3390/machines11070677.

[7] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu and C. Xu, "GhostNet: more features from cheap operations," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, 2020, pp. 1577-1586, doi: 10.1109/CVPR42600.2020.00165.

[8] X. Liu, R. Du, L. Tan, J. Xu, C. Chen, H. Jiang, and S. Aldwais, "CIB-SE-YOLOv8: optimized YOLOv8 for real-time safety equipment detection on construction sites," *arXiv*, 2024, doi: 10.48550/arXiv.2410.20699.

[9] G. Papanastasiou, A. Drigas, C. Skianis, M. Lytras, and E. Papanastasiou, “Virtual and augmented reality effects on K-12, higher and tertiary education students’ twenty-first century skills,” *Virtual Real*, vol. 23, no. 4, pp. 425–436, Dec. 2019, doi: 10.1007/s10055-018-0363-2.

[10] X. Wang, “Deep learning in object recognition, detection, and segmentation,” *Foundations and Trends in Signal Processing*, vol. 8, no. 4, pp. 217–382, 2016, doi: 10.1561/2000000071.

[11] M. Safaldin, N. Zaghden, and M. Mejdoub, “An improved YOLOv8 to detect moving objects,” *IEEE Access*, vol. 12, pp. 59782–59806, 2024, doi: 10.1109/ACCESS.2024.3393835.

[12] Saikumar P and Divya TL, “Real-time object detection and augmentation,” *World Journal of Advanced Engineering Technology and Sciences*, vol. 12, no. 2, pp. 938–944, Aug. 2024, doi: 10.30574/wjaets.2024.12.2.0359.

[13] V. Li, G. Amponis, J.-C. Nebel, V. Argyriou, T. Lagkas, and P. Sarigiannidis, “Object recognition for augmented reality applications,” *Azerbaijan Journal of High Performance Computing*, vol. 4, no. 1, pp. 15–28, 2021, doi: 10.32010/26166127.2021.4.1.15.28.

[14] T. Napier and I. Lee, “Using mobile-based augmented reality and object detection for real-time abalone growth monitoring,” *Computers and Electronics in Agriculture*, vol. 207, p. 107744, Apr. 2023, doi: 10.1016/j.compag.2023.107744.

[15] M. Łysakowski, K. Żywanowski, A. Banaszczyk, M. R. Nowicki, P. Skrzypczyński, and S. K. Tadeja, "Real-time onboard object detection for augmented reality: enhancing head-mounted display with YOLOv8," *2023 IEEE International Conference on Edge Computing and Communications (EDGE)*, Chicago, IL, USA, 2023, pp. 364-371, doi: 10.1109/EDGE60047.2023.00059.

[16] J. Zeng and H. Zhong, “YOLOv8-PD: an improved road damage detection algorithm based on YOLOv8n model,” *Scientific Reports*, vol. 14, p. 12052, Apr. 2024, doi: 10.1038/s41598-024-62933-z.

[17] A. Ferdi, "Lightweight G-YOLOv11: Advancing efficient fracture detection in pediatric wrist X-rays," arXiv, 2024, doi: 10.48550/arXiv.2501.00647.

[18] F. Akhmedov, R. Nasimov, and A. Abdusalomov, “Dehazing algorithm integration with YOLO-v10 for ship fire detection,” *Fire*, vol. 7, no. 9, p. 332, Sep. 2024, doi: 10.3390/fire7090332.

[19] Y. Li, C. Zhu, Q. Zhang, J. Zhang, and G. Wang, “IF-YOLO: An efficient and accurate detection algorithm for insulator faults in transmission lines,” *IEEE Access*, vol. 12, pp. 167388–167403, Nov. 2024, doi: 10.1109/ACCESS.2024.3496514.

[20] J. Murrugarra-Llerena, L. Kirsten and C. R. Jung, "Can we trust bounding box annotations for object detection?," *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, New Orleans, LA, USA, 2022, pp. 4812-4821, doi: 10.1109/CVPRW56347.2022.00528.
[21] F. Gao, Z. Yang, Q. Liu, Z. Zhang, B. Li, and H. Zhang, "Enhanced oracle bone corrosion detection using attention-guided YOLO with ghost convolution," *Discover Artificial Intelligence*, vol. 4, no. 1, p. 1, Dec. 2024, doi: 10.1007/s44163-024-00178-5.
[22] H. Hou, M. Guo, W. Wang, K. Liu, and Z. Luo, "Improved lightweight head detection based on GhostNet-SSD," *Neural Process Lett*, vol. 56, no. 2, p. 126, Apr. 2024, doi: 10.1007/s11063-024-11563-7.
[23] Y. Yang, Z. Song, T. D. Palaoag, and S. Li, "An improved model for people detection based on YOLOv8," in *9th International Conference on Information Technology and Digital Applications*, 2024, pp. 1–7. doi: 10.1109/ICITDA64560.2024.10809805.
[24] P. Henderson and V. Ferrari, "P. Henderson and V. Ferrari, "End-to-end training of object class detectors for mean average precision," *arXiv*, 2016, doi: 10.48550/arXiv.1607.03476.
[25] D. M. W. Powers, "Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation," *arXiv*, 2020, doi: 10.48550/arXiv.2010.16061.
[26] B. Shah and H. Bhavsar, "Time complexity in deep learning models," *4th International Conference on Innovative Data Communication Technology and Application*, Procedia Computer Science, 2022, pp. 202–210. doi: 10.1016/j.procs.2022.12.023.
[27] K. Kishor, R. Rani, A. K. Rai, and V. Sharma, "3D Application development using unity real time platform," *Proceedings of Fourth Doctoral Symposium on Computational Intelligence*, 2023, pp. 665–675. doi: 10.1007/978-981-99-3716-5_54.
[28] Z. Ye, L. Huang, Y. Wu, and M. Hu, "AR visualization system for ship detection and recognition based on AI," arXiv, 2023, doi: 10.48550/arXiv.2311.12430.

BIOGRAPHIES OF AUTHORS

Arya Krisna Putra, S.Kom. is a master's student at Bina Nusantara University. He earned his bachelor's degree in Informatics from Universitas Diponegoro, where he developed a strong foundation in software engineering. Before continuing his education, he gained practical experience in the industry as a frontend software developer. During this time, he also worked as a mentor in software engineering, sharing his knowledge and helping others advance in their careers. Currently, he is passionate about data science and deep learning, exploring these fields with enthusiasm. Known for his curiosity and eagerness to learn, he is always open to new ideas and strives to grow both personally and professionally. He can be contacted at email: arya.putra006@binus.ac.id.

Fiqri Ramadhan Tambunan, S.Kom. is a Master of Computer Science student at Bina Nusantara University. He completed his Bachelor of Computer Science degree at the University of North Sumatra and graduated with Cum Laude predicate. Before continuing his studies, he had worked in one of the domestic industries while working on several jobs as a software engineer. Has the motivation to always be enthusiastic in learning, and really likes new things especially in the field of research related to machine learning and artificial intelligence. He can be contacted via email: fiqri.tambunan@binus.ac.id.

Samson Ndruru, S.Kom. is a Master's student in Computer Science at Bina Nusantara University. He earned his Bachelor's degree in Computer Science, where he developed a strong foundation in software development, data structures, and system architecture. His research interests include Augmented Reality (AR), Immersive Technology, Human-Computer Interaction (HCI), Artificial Intelligence (AI) and Big Data. Currently, he is focusing on developing AR technology to enhance educational experiences, particularly through interactive storytelling and biblical narratives. His work integrates AR platforms with intelligent virtual assistants to create immersive learning environments. He actively engages in research and collaborative projects to push the boundaries of interactive digital experiences. He can be contacted at email: samson.ndruru@binus.ac.id.

Ir. Andry Chowanda, S.Kom., MM, Ph.D., MBCS, CCP, CME, IPM, SMIEEE earned his Bachelor's degree in Computer Science from Bina Nusantara University (Indonesia, 2009), a Master's in Business Management from BINUS Business School (Indonesia, 2011), and a Ph.D. in Computer Science from Nottingham University (England, 2017). He is now a Computer Science Lecturer at Bina Nusantara University. His research is in agent architecture and machine (and deep) learning. His work is mainly on how to model an agent that can sense and perceive the environment based on the perceived data and build a social relationship with the user over time. In addition, he is also interested in serious game and gamification design. He can be contacted at email: achowanda@binus.edu.