Analyzing Performance YOLOv8 in Detection, Amount, and Type of Acne Lesions

1st Devi Zahra Aulia
School of Computing
Telkom University
Bandung, Indonesia
devizahra@student.telkomuniversity.ac.id

2th Putu Harry Gunawan
School of Computing
Telkom University
Bandung, Indonesia
phgunawan@telkomuniversity.ac.id

3rd I Wayan Palton Anuwiksa
School of computing
Telkom University
Bandung, Indonesia
paltonanuwiksa@telkomuniversity.ac.id

4th Narita Aquarini
E ´cole Doctorale Science Economics
Universit ´e de Poitiers Intervenant Finance
La Rochelle, France
aquarinin@excelia-group.com

5nd Ida Ayu Manik Partha Sutema.
Clinical Pharmacy Dept, Faculty of Sciences
Bali International University
Bali, Indonesia
manikparthasutema@unbi.ac.id

***Abstract*—Acne is one of the most common skin diseases and usually appears on the facial area. The onset of acne is caused by oily skin conditions and excessive pores, resulting in the appearance of small bumps. Acne has 5 types, namely whiteheads, blackheads, nodules, pustules, and papules. Although the five types of acne have different characteristics, there are still dermatologists who make mistakes in making diagnoses, making it difficult for dermatologists to provide the right diagnosis. This research uses the YOLOv8 model. YOLOv8 has broad potential in the field of dermatology, because this model can detect objects of various sizes and has accurate and efficient performance results to reduce diagnostic errors that previously often occurred due to differences in the characteristics of five types of acne. In addition, YOLOv8 is a model that can perform several tasks at once such as quantitative analysis, segmentation, and classification of acne. Therefore, this study performs three detections, namely acne detection, number, and type of acne. The data used for the three detections contains 823 images of people who have acne and labels to help the process of detecting acne and acne type. The accuracy produced by this model is 89% for the detection of acne, 82% for the number of acne, and 49% for the type of acne. However, the detection of acne types resulted in low accuracy due to an imbalance in the number of labels, similarity between acne types, errors in labeling, poor selection of hyperparameters, and a less diverse dataset. Future efforts should be able to correct dataset imbalances, such as the number of datasets, labeling acne type classes evenly to improve detection results by producing better accuracy, using the right hyperparameters, and data augmentation. This study confirms that YOLOv8 can improve the efficiency of dermatologists in diagnosing accurate acne so that patients can receive appropriate treatment.**

***Index Terms*—YOLOv8, acne, acne detection, deep learning, dermatology.**

I. Introduction

Acne is one of the most common skin diseases and usually appears on the facial area. The onset of acne is caused by oily skin conditions and excessive pores, resulting in the appearance of small bumps [1]. The appearance of acne can interfere with a person's psychology such as depression, anxiety, fear, low self-esteem, and shame which can lead to suicidal tendencies. Before entering this phase, acne sufferers will usually see a dermatologist first to get more in-depth information about the type of acne and the right treatment to reduce the acne [2].

The diagnosis process carried out by dermatologists still uses many traditional methods, which is detecting by examining acne on the patient's face one by one [3]. This process can cause misdiagnosis due to the similarity between one type of acne and another and takes a long time to detect acne [4]. Therefore, traditional methods are less effective in the detection process because they are often subjective [5]. To produce a correct diagnosis, a more advanced methodology is needed, such as detecting acne types with deep learning. This method can be done using YOLO [6].

The You Look Only Once (YOLO) method is known for their detection accuracy. One such researcher used the YOLOv4 model with 91.25% accuracy in detecting acne types. The researchers focused on 4 types of acne types, including whiteheads, nodules, pustules, and papules. The model had some incorrect predictions, such as the model detecting a papule, but the bounding box identified it as a nodule. This can happen because nodules and papules look similar to each other [7].

To improve detection compared to previous researchers, a model called YOLOv8 can be used. YOLOv8 is the newest variation of YOLO. This model is more sophisticated in detection because it has a more flexible architecture. One of the researchers detected people who did not wear helmets from video pieces using the YOLOv8 model resulting in an accuracy value of 93%. This indicates that the YOLOv8 model provides a higher accuracy value than the previous model. This model can detect objects in real time taken from video snippets. Not only that, YOLOv8 has more accurate detection results and can detect very small and complex objects [8].

In this experiment, YOLOv8 is used to detect acne, amount of acne, and type of acne. YOLOv8 was chosen because the model can perform multiple tasks which makes it suitable for medical imaging applications. The main focus of this paper is how effective YOLOv8 is in performing detection using current technological advances. The model is a reliable automated system built by inputting images to detect acne, amount, and type of acne which is expected to help dermatologists in choosing the best treatment for patients.

II. Methodology

A. YOLO

YOLO (You Only Look Once) is a model presented by Joseph Redmon, et al in 2016. There are 8 versions of YOLO, one of the eight YOLO variations is YOLOv8. YOLOv8 is the result of the evolution of YOLOv7 which was published in 2023 [9]. The newest version, YOLOv8 has a more powerful architecture than the previous version that can detect multiple objects in images or videos with a complex size range [10]. From its algorithm architecture, YOLOv8 has three main components:

1) Backbone: Backbone can extract the important features present in the image. In YOLOv8, backbone has three modules: CSP, C2f, and SPP. The CSP (Cross Stage Partial Network) module is designed to split the features into separate components for convolution operations. The C2f (Cross Stage Partial Connection with 2 Feature Maps) model is a variation of CSP to increase the information between layers and reduce the number of parameters. Meanwhile, the SPP (Spatial Pyramid Pooling) module improves detection for all data with various scales [11].
2) Neck: The neck in YOLOv8 is effectively performing feature fusion. Feature fusion is a process of combining different features extracted from data to improve model performance. YOLOv8 uses feature fusion with the PAN-FPN module which comprises sampling fusion and downsampling fusion [12].
3) Head: Head in YOLOv8 was responsible for supervising the last step in the detection process, which resulted in the class anchor box and information about the bounding box. [13].

From the three components above, the process flow of YOLOv8 in detecting which performs its tasks sequentially to produce optimal, structured, and efficient performance and produce specific and accurate predictions. The detection process is presented in the form of labels and Bounding Box [14]. Bounding box is used to indicate the location of the detected object based on the number of labels available [15].

B. Flowchart System

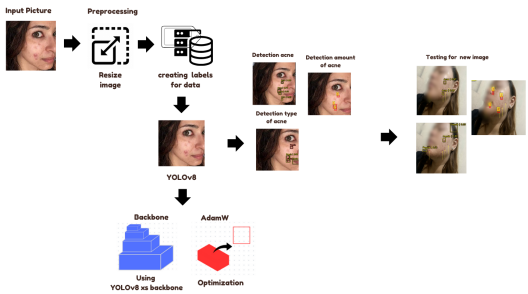

Fig. 1. System flow of YOLOv8 in acne detection.

The method used in this paper is called YOLOv8 which is based on a deep learning approach using Keras as a library and TensorFlow as a framework. In addition, this model consists of backbone, AdamW, and callback. Each has the function to extract features, perform optimization by measuring accuracy and loss values in detecting acne by generating a bounding box [16]. While the callback is used to anticipate overfitting of the model [17]. Fig. 1 is the flow of the system to perform three detections: detection of acne, number, and type of acne using the YOLOv8 model.

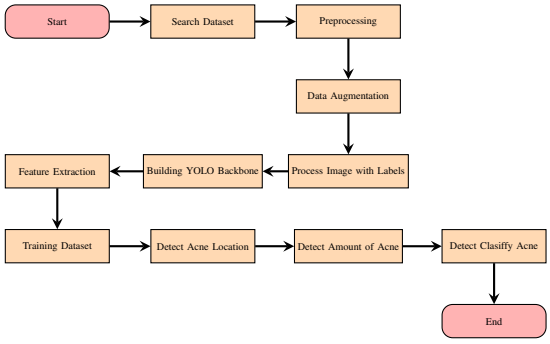

Fig. 2. Research methodology for detecting acne, number, and type of acne.

The first step is to search for datasets on various platforms, in this case using Kaagle to search dataset. After getting a dataset related to acne detection, the next step is preprocessing. This step checks for empty information and the format of the label data is appropriate or not so the YOLOv8 model can read it. After preprocessing, data must be labeled and adjusted to desired detection. For example, if the detection of acne types is desired, the existing labels must be changed first and adjusted to the types of acne to be detected. Next, building a YOLOv8 model with a backbone, the model can process the input image and extract the important features in the image. The extracted features will generate predictions according to detection what is being run. The next step is data augmentation which serves to improve the performance and generalization of the YOLOv8 model, and avoid overfitting. This data augmentation process uses JitteredResize to resize the image to the target size (640x640), apply random scaling by a factor between 0.8 to 1.25, and maintain the aspect ratio of the image.

After performing data augmentation, the next step is training the dataset, the YOLOv8 model will learn the patterns and characteristics of acne based on the labels. In this process, the model will use hyperparameters such as epoch and learning rate to get optimal results. After training the dataset, the model will perform three detections, including acne detection, amount, and type of acne. The final step in this process is to perform testing on test data and user-entered data in the form of bounding boxes and descriptions according to the detection used. The implementation process of the above flow can be seen in Fig. 2. Moreover, this process also checks how reliable the YOLOv8 model is in detecting image data, as each image has different light conditions and image quality.

C. Dataset

The dataset used in this paper was derived from Kaagle and published by a Kaagle user named Arturo Bandini [1]. Kaagle is a platform which consists with various sources of scientific data projects to make it easier for someone to find data without having to search and collect the data themselves [18]. After choosing the dataset to be used, the dataset was saved in Google Drive to facilitate the preprocessing and implementation of the YOLOv8 model.

[1]https://www.kaggle.com/datasets/osmankagankurnaz/acne-dataset-in-yolov8-format

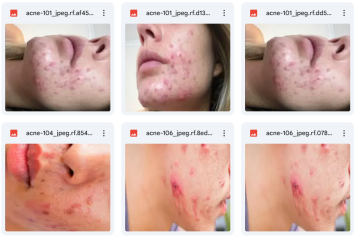

Fig. 3. Some example images from the test dataset.

This dataset has a grand total of 938 images divided into three categories: training data, test data, and valid data. The numbers for each of these data are: 823, 59, 56. Then each data has two subfolders: one contains face images containing acne, and the other contains labels. As seen in Fig. 3 is an example image of the test data. Each data image has a label, where the label content consists of 5 categories:

1) Object class is the content of the type of object to be detected.
2) The x coordinate is the midpoint of the x coordinate for the object or bounding box.
3) The y coordinate is the midpoint of the y coordinate for the object or bounding box.
4) Height is the height of the bounding box.
5) Width is the width of the bounding box.

The five categories of labels are saved in .txt file. This paper has two types of files for labeling, which are the labels for acne detection and acne type. Table I is an example of the contents of the label file for acne detection. The file only has one class called class 0, which indicates acne.

TABLE I
EXAMPLE OF TEXT FILE FOR ANNOTATIONS IN TEST DATA

Class	X Center	Y Center	Width	Height
0	0.07421875	0.24296875	0.08828125	0.10625
0	0.70078125	0.03125	0.0359375	0.0625
0	0.6078125	0.41796875	0.06484375	0.08828125
0	0..09140625	00.42421875	0.03203125	0.05
0	0.6625	0.15703125	0.05546875	0.08203125

Meanwhile, to detect the type of acne, a different label is used than before. The difference with this file is using numbers: 1 for blackheads, 2 for whiteheads, 3 for nodules, 4 for papules, and 5 for pustules. Table II shows the latest label file of one image from the data train for detecting acne types. The results from the Table II show data has several types of acne such as 1 comedo, 1 nodule, 1 papule, and 2 pustules.

TABLE II
EXAMPLE OF TEXT FILE FOR ANNOTATIONS IN TEST DATA

Class	X Center	Y Center	Width	Height
3	0.07421875	0.24296875	0.08828125	0.10625
4	0.70078125	0.03125	0.0359375	0.0625
5	0.6078125	0.41796875	0.06484375	0.08828125
1	0..09140625	00.42421875	0.03203125	0.05
5	0.6625	0.15703125	0.05546875	0.08203125

For detection of acne and the amount of acne, it uses the full of dataset, but to detect the type of acne, it only uses 130 images for train data, 47 images for valid data, and 45 images for test data. The reason why only part of the dataset was used is because to avoid redundancy, reduce the risk of overfitting, and using too much data requires more TPU resources

D. Evaluation Metrics

In this study, the system was evaluated using test data that yield evaluation metrics including accuracy, precision, recall, and F1-Score [19]. The evaluation metric called confusion matrix are consists of several variables, such as:

- True Positive (TP): The number of positive samples that are correctly predicted as positive.
- False Positive (FP): The number of negative samples that are incorrectly predicted as positive (type I error).
- False Negative (FN): The number of positive samples that are incorrectly predicted as negative (type II error).
- True Negative (TN): The number of negative samples that are correctly predicted as negative.

Acuraccy is one of the most frequently used metric to access a model's performance in classification where the ratio of samples to be compared to the total number of samples to produce a correct classification [20]. Here is the formula for calculating accuracy:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \tag{1}$$

Precision is the percentage of correctly classified positive samples out off all predicted positive samples. It gauges how accurately recognized of the positive predictions made by the model [20]. Here is the formula:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

Recall represents the proportion of correctly classified positive samples among all true positive samples. It assesses the model's ability to identify all relevant positive cases [20]. Here is the formula for calculating recall:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

The F-measure combines of precision and recall into single metric that balances between the two. It has a value between from 0 to 1. The F1-score is a specific variation of the F-measure where precision and recall have a fair evaluation of the model's performance by giving equal weight [20]. Here is the formula for calculating the F1-score:

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

All the above-mentioned formulas can be explained using the confusion matrix. To produce a good and ideal confusion matrix, effective hyperparameters must be used. These hyperparameters are configured before training to control how the model learns and detects objects. There are 4 hyperparameters used for this training process, specifically:

1) *Learning rate*: To regulate and control how many steps the model takes, update the weights during optimization.
2) Weight decay: Regulatory method to prevent overfitting.
3) Global clipnorm: Set limits for maximum gradient values during training for stability.
4) Epoch: The amount of time it takes for the entire dataset to complete during the training process.

TABLE III
HYPERPARAMETER FOR TRAINING DATA

Detection	Learning rate	Weight decay	Global clipnorm	Epoch
Acne	0.0007	0.0009	10.0	140
Amount of acne	0.0007	0.0009	10.0	140
Type of acne	0.0007	0.0009	5.0	140

All hyperparameters in the Table III have an important role in impacting on the performance of the YOLOv8 model in detecting objects, such as increasing the accuracy value and improving the efficiency of training data. For acne type detection, a global ellipnorm value of 10.0 was not used and an ellipnorm value of 0.5 was chosen to maintain a balance between understanding important features and preventing overfitting.

III. RESULT AND EVALUATION RESULTS

A. Location of Acne

Fig. 4. Acne detection for data train.

The first detection is acne detection. In this case, the model detects the train data showing a red bounding box and a description of the location has acne can be seen in the Fig. 4. After performing detection on the train data, the model predicts the test data by generating a new yellow bounding box. The two bounding boxes are given different colors to show the performance differences of the YOLOv8 model such as error in understanding and predicting acne. On the Fig. 5, there are two bounding boxes: the actual bounding box (red bounding box) and the predicted bounding box (yellow bounding box).

Fig. 5. Prediction acne detection for data test.

Acne detection was evaluated using the confusion matrix shown in Fig. 6. The results of the confusion matrix revealed a strong ability of the model to identify acne in 512 samples. However, there are still incorrect detection results which cause one of the factors of the model's inability to detect. For example, there were 105 samples which had acne but the model detected as non-acne. Furthermore, there are 83 samples whose locations do not have acne but are detected as acne.

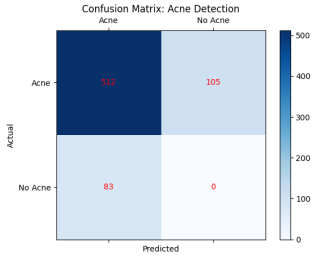

Fig. 6. Confusion matrix of acne detection.

TABLE IV
PERFORMANCE METRICS FOR ACNE DETECTION

Metric	Value
Accuracy	0.89
Precision	0.82
Recall	0.89
F1-score	0.85

From the confusion matrix results, the model gained precision, recall, and F1 values of 0.82, 0.89, 0.85 and an accuracy of 89%. As shown in Table IV , these metrics show the model can accurately detect acne and perform detection without missing anything and maintain balance in the data.

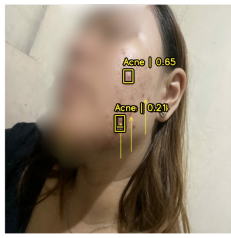

Fig. 7. Result of detection acne using data from user.

Under this detection, user can input an image into the system and then the system will detect acne in the image. The result of this detection shows a yellow bounding box and the confidence value of the acne, the illustration can be seen in Fig. 7. The results of the image suggest the model can only detect three acne with different confidence score values. The confidence score value is an indication of how confident the model can be in making the detection. These difference occurs due to the presence of noise that interferes with the detection process and can reduce the confidence score value of the model. Solution for this problem are to normalize, optimize the training process by adding more varied data, and optimize the image resolution because these factors can affect the detection results learned by the model.

B. Amount of Acne

The second detection is detection amount of acne by presenting the bounding box with numbers. The bounding box and number caption are determined based on the number of labels present. This detection does not add a new bounding box to predict the test data, because the results of detection and prediction will be the same as the amount of labels in each image. Results of detection the amount of acne for one of the data train has 20 acne, the illustration can be seen in the Fig. 8.

The results in Fig. 9 produce a loss value of 2.805, a box loss metrix of 1.636, and a class loss of 1.169. From these

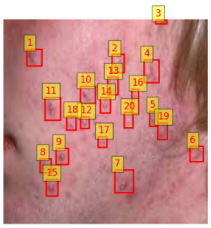

Fig. 8. Result of detection amount of acne with YOLOv8.

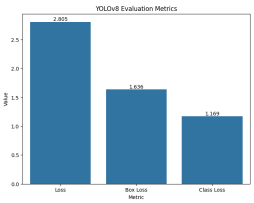

Fig. 9. Evaluation matrix for amount acne detection

results, it was explained the loss, the box loss, and class loss value in this detection model can detect well. From the confusion matrix results, the model gained precision, recall, and F1 values of 0.80, 0.82, 0.81 and an accuracy of 82%. As shown in Table V, these metrics show the model can accurately detect amount of acne and perform detection without missing anything and maintain balance in the data.

TABLE V
PERFORMANCE METRICS FOR AMOUNT ACNE DETECTION

Metric	Value
Accuracy	0.82
Precision	0.80
Recall	0.82
F1-score	0.81

Fig. 10. Result of detection amount acne using data from user

Additionally, user can check the amount of total acne on the previously inputted photo. The results in Fig. 10 show the model successfully detected a total of 100 acne. This result only has 2 acne right on the face area as the rest are outside the face area. This problem can occur due to the lack of segmentation, low image resolution, and presence of noise which causes the YOLOv8 model to have trouble differentiating actual acne from visualizations outside the facial area. Solutions to these problems can be solved by implement face segmentation and fine-tuning the model to include penalties for areas outside the face.

C. Type of Acne

The last detection of this paper is acne type detection. The result includes a red bounding box and a description of the type of acne experienced by the patient. Classification process on the train data shown acne type based on labels in the file.txt. The labels include 5 types of acne: blackheads, whiteheads, nodules, pustules, and papules. For example, the results of the acne type detection performed for the test data can be seen in Fig. 11 which results just 3 types of acne: nodules, pustules, and papules.

Fig. 11. Detection type of acne for train data.

Same as acne detection, this model predicts the test data by creating a new yellow bounding box. The difference between the bounding boxes are actual bounding box (red bounding box) and prediction bounding box (yellow bounding box). The illustration can be seen in Fig. 12.

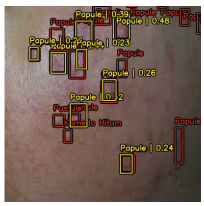

Fig. 12. Deetection type of acne for test data.

The illustration in Fig. 12 show how model can only detect one type of acne (papules). Then, in the training process model can only detect 3 types of acne: nodules, pustules, and papules which causes the model only detect one type of acne in the test data. Then, based on the Fig. 13 produces an evaluation graph of acne type detection with 3 categories: loss, box loss, and class loss. On the loss graph shows the models can still generate high accuracy values. Next, the box loss graph shows the model has difficulty in defining the right position and size for the acne bounding box. Lastly, the class loss graph shows the model can detect the type of acne after the bounding box is raised.

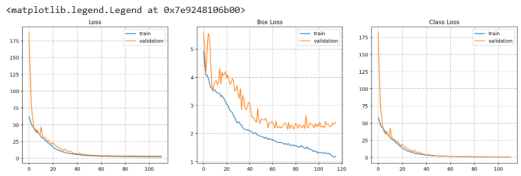

Fig. 13. Graphic loss detection type of acne.

This condition occurs due to an imbalance in the label, where the train and test data are more dominant in the papule type, and the model only focuses on recognizing the papule class even though the other classes are less optimal. For an illustration of the class differences in the training data and valid data, refer to Fig. 14.

Model in this detection resulted in precision, recall, and F1 values of 0.72, 0.49, 0.58, and an accuracy of 49% as shown in Table VI. Based on this detection, the accuracy value is small due to biases in the model such as low data quality,

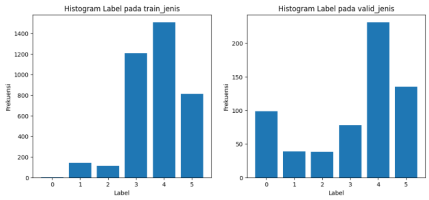

Fig. 14. Histogram for label type of acne.

unequal number of labels, and mismatched hyperparameters. These problems can be overcome by sampling techniques, using appropriate loss functions, adding data, using ideal hyperparameters in order to detect other types of acne.

TABLE VI
PERFORMANCE METRICS FOR TYPE OF ACCNE DETECTION

Metric	Value
Accuracy	0.49
Precision	0.72
Recall	0.49
F1-score	0.58

In this case, user can make the detection using a new image. The image shows how the model can only detect the 3 acne papules illustrated in Fig. 15. The YOLOv8 model does not comprehend the variation of acne types due to the unequal amount of labeling, inconsistent lighting in the image which causes shadows or bright areas not recognized by the model, and different predicted face positions from the training process. Solutions to these problems are applying data augmentation which includes various face positions and viewpoints or using face alignment techniques, establishing lighting normalization with adaptive histogram equalization techniques for the model to recognize acne types in different lighting conditions, and performing an equalization process on the number of labels so that there are no dominant or minority labels.

Fig. 15. Result of detection type acne using data from user

IV. Conclusion

This paper show the model YOLOv8 performs well in detecting acne and the amount of acne with a high accuracy around 89% and 82%. However, a challenge comes when the model detects acne types, where the model produces a lower accuracy of 49% due to the imbalance of the dataset, especially the dominance of acne types on the label (papules). This imbalance affects the model's ability to recognize minority acne types. In addition to data imbalance, there are other factors such as lack of lighting, low image resolution, and environmental conditions that hinder the effectiveness of the model. These issues suggest that future research should expand the dataset, perform label equalization, improve image quality through preprocessing techniques such as lighting normalization, increase resolution, and fine-tune hyperparameters to increase model robustness. Moreover, doing data augmentation to improve the generalization ability of the models, so the model can handle various skin conditions, acne types, and other external factors. Furthermore, the further development of the YOLOv8 model ensures that the proposed model is reliable for use in clinical decisions. Thus, the success of future research can bring a broader positive impact in the field of dermatology in producing appropriate diagnosis and treatment for patients.

References

[1] S. Alzahrani, B. Al-Bander, and W. Al-Nuaimy, "Attention mechanism guided deep regression model for acne severity grading," *Computers*, vol. 11, no. 3, p. 31, 2022.

[2] U. Khalid, L. Chen, A. A. Khan, and F. Mehmood, "D-gan: An automatic acne detection, severity, and assessment framework using generative adversarial network with deep neural network," 2024.

[3] J. Zhang, L. Zhang, J. Wang, X. Wei, J. Li, X. Jiang, and D. Du, "Sa-rpn: A spacial aware region proposal network for acne detection," *IEEE Journal of Biomedical and Health Informatics*, 2023.

[4] H. Zhang and T. Ma, "Acne detection by ensemble neural networks," *Sensors*, vol. 22, no. 18, p. 6828, 2022.

[5] M. B. Islam, M. S. Junayed, A. Sadeghzadeh, N. Anjum, A. Ahsan, and A. S. Shah, "Acne vulgaris detection and classification: a dual integrated deep cnn model," *Informatica*, vol. 47, no. 4, 2023.

[6] J. Wang, C. Wang, Z. Wang, A. H. Hounye, Z. Li, M. Kong, M. Hou, J. Zhang, and M. Qi, "A novel automatic acne detection and severity quantification scheme using deep learning," *Biomedical Signal Processing and Control*, vol. 84, p. 104803, 2023.

[7] N. A. M. Isa and N. N. A. Mangshor, "Acne type recognition for mobile-based application using yolo," in *Journal of Physics: Conference Series*, vol. 1962, no. 1. IOP Publishing, 2021, p. 012041.

[8] A. Aboah, B. Wang, U. Bagci, and Y. Adu-Gyamfi, "Real-time multi-class helmet violation detection using few-shot data sampling technique and yolov8," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 5350–5358.

[9] 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. 677, 2023.

[10] M. Safaldin, N. Zaghden, and M. Mejdoub, "An improved yolov8 to detect moving objects," *IEEE Access*, 2024.

[11] D. Wang, Y. Zhang, R. Zhang, G. Nie, and W. Wang, "Detection and assessment of post-earthquake functional building ceiling damage based on improved yolov8," *Journal of Building Engineering*, vol. 98, p. 111315, 2024.

[12] F. Solimani, A. Cardellicchio, G. Dimauro, A. Petrozza, S. Summerer, F. Cellini, and V. Renò, "Optimizing tomato plant phenotyping detection: Boosting yolov8 architecture to tackle data complexity," *Computers and Electronics in Agriculture*, vol. 218, p. 108728, 2024.

[13] M. S. Z. Abid, B. Jahan, A. Al Mamun, M. J. Hossen, and S. H. Mazumder, "Bangladeshi crops leaf disease detection using yolov8," *Heliyon*, vol. 10, no. 18, 2024.

[14] X. Wang, H. Gao, Z. Jia, and Z. Li, "Bl-yolov8: An improved road defect detection model based on yolov8," *Sensors*, vol. 23, no. 20, p. 8361, 2023.

[15] A. Fauzi, S. Madenda, E. P. Wibowo, A. F. N. Masruriyah *et al.*, "The importance of bounding box in motion detection," in *2020 Fifth International Conference on Informatics and Computing (ICIC)*. IEEE, 2020, pp. 1–5.

[16] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 568–578.

[17] M. K. Rusia and D. K. Singh, "An efficient cnn approach for facial expression recognition with some measures of overfitting," *International journal of information technology*, vol. 13, no. 6, pp. 2419–2430, 2021.

[18] I. Martinez, E. Viles, and I. G. Olaizola, "Data science methodologies: Current challenges and future approaches," *Big Data Research*, vol. 24, p. 100183, 2021.

[19] R. Yacouby and D. Axman, "Probabilistic extension of precision, recall, and f1 score for more thorough evaluation of classification models," in *Proceedings of the first workshop on evaluation and comparison of NLP systems*, 2020, pp. 79–91.

[20] A. Tharwat, "Classification assessment methods," *Applied Computing and Informatics*, vol. 17, no. 1, pp. 168–192, 2021.