WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Improved small-object detection using YOLOv8: A comparative study

Bindu Madhav shrotriya*, Dr Vishal Shrivastava , Dr Akhil Pandey, Dr Karuna Sharma

*Bachelor of Technology in Artificial Intelligence and Data Science, Arya College Of Engineering and Information and Technology, Jaipur(RJ)
bindumadhavshrotriy@gmail.com

1. Introduction

Object detection plays a key role in computer vision and uses a wide range of applications in areas such as realtime monitoring, autonomous driving, and intelligent security systems. Before 2014, traditional methods for object recognition were strongly based on manual characteristic extraction, which is not only time-consuming, but also susceptible to instability. The deformable partial base model (DPM) [1] has improved the specific adaptability of deformations once as state of the art. But it fought with a considerable spin and lacked strength. In recent years, the development of folding networks (CNNs) has led to considerable advances in deep learning-based object recognition algorithms, and two branches have been developed with anchor-free and anchor-based methods [2]. The anchor-based method revolutionizes object recognition and consists of the emergence of CNNS (CNS) with folding nerves (CNNs) with two major branches of two major branches. Anchor-free and anchor-based approaches [2].

The anchor-based model is further divided into a single size and two stages of detectors. Two-stage models such as RCNN are used to create proposals from areas where the region is increasing and to predict limit boxes. These methods provide high average accuracy (MAP), but slows down the dependence on several network stadiums [3]. We have attempted to improve speed with complex architectures, with improvements such as almost R-CNN [4], faster R-CNN [5], MASK R-CNN [6], and more, but we have attempted to improve speed for real-time applications. Despite the continuous improvements between Yolo versions, including recently published Yolov8, detection of small and dense objects is particularly challenging. The standard recognition head in Yolov8 often does not record fine details on objects below 8 u 8 U 8 raw pixels, leading to lack of recognition or overlapping. To improve this limitation, this study proposed an optimized version of Yolov8 using a newly defined detection head, which increased the small received fields and increased number. These changes improve performance, especially in scenarios involving a large number of small objects. Our model includes an average frame rate of 30 fps, improving recall rates from below 60% to above 80%, indicating actual use by counting dense objects such as sand particles. To address this limitation, this study proposes an optimized version of YOLOv8 with redefined detection heads that have a smaller receptive field and increased count. These modifications enhance performance, especially in scenarios involving numerous small objects.

2. Method

2.1. Models Architecture

The backbone and head of neural networks are two fundamental components of the Yolov8 architecture, and are improved compared to previous iterations of the Yolo algorithm [8]. The currentlyrevised CS architecture consists of 35 foldable layers, which use cross-boost connections to improve data transmission between layers and serves as the basis for Yolov8. Thebounding boxes, element evaluations, and probabilities of the recognized classes are predicted by a Yolov8 head made up of a series of folding layers, followed by a fully connected layer. A notable aspect of Yolov8 is the inclusion of devices for custody [9] in the head of the network. This function allows the model to concentrate on different regions of the image and allow the element to be changed according to its relevance. Another notable aspect of Yolov8 is its ability to recognize objects at many scales achieved through distinctive hierarchical networks [10]. The model can reliably recognize objects of different sizes in an image, as many layers of the network recognize objects with different standards. Figure 1 shows a typical Yolov8 model structure.

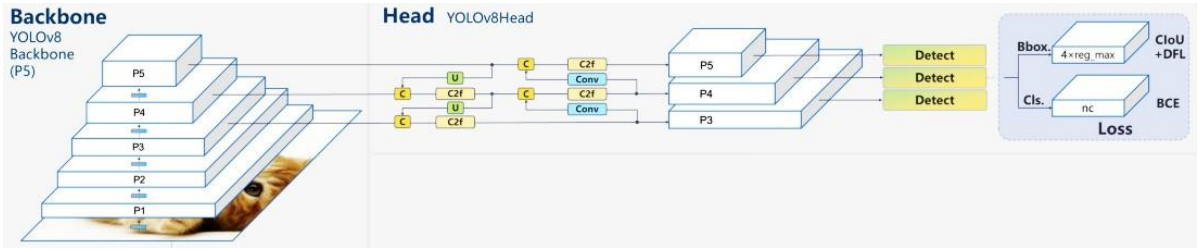


Figure 1. Structure of YOLOv8.

2.2. Head

To commit traditional methods, the regression approach has proven to be the focus of research in the field of three-dimensional estimation of human pose poses. These methods use deep learning techniques to learn the mapping relationships between poses and shapes from photographs, allowing for direct regression of human poses and shapes. This approach achieves more accurate estimates through extensive training data and deep neural networks. In Yolov8, the head part refers to the hierarchical structure of neuronal networks responsible for processing functional cards according to characteristics extraction from the basic level. In particular, the Yolov8 part contains three main important components: The sample layer and the root layer that distinguish between layers contain three main important components: The distinguishing layer is responsible for converting the input function card into a recognition containment box. Typically, the distinguishing layers are converted to Yolov8 features in bounding boxes of different scales and corresponding categorical predictions through collapse operations. Each layer of recognition is assigned to an anchor box to recognize objects of different scales. The UP test layer is used to increase the resolution of the feature card. These levels typically use the development process to use upsampling and conversions with low resolution features at high resolution. The upsampling layer is primarily used to increase awareness of small sizes by the model. The root layer is used to connect different levels of feature cards. You can connect the feature card from the previous layer to the feature card from the previous level to get a feature card with different scaling function information. This numerous feature fusion helps the model recognize objects of different sizes and types. In summary, it can be said that part of Yolov8's head will enable cards at recognition boundaries regarding the combination of recognition layers, several standard and type differentiation combinations, to achieve differentiation and type combinations of characteristics and type differentiation.

2.3. Deficiency and optimizer

In standard object detection tasks, the problem of missing detection or poor detection effect often occurs when there are small objects in the data set. The reason is stated as follows: The YOLOv8 model has 3 detection heads by default, which can perform multi-scale detection of targets. Among them, P3/8 corresponds to a detection feature map size of 80*80, which is utilized for recognizing items larger than 8*8; P4/16 relates to a detection feature map size of 40*40, which is applied to recognize items larger than 16*16; and P5/32 corresponds to a recognition characteristics map size of 20*20, which is used to identify items larger than 32*32, as illustrated below:

Updated head: detecting small objects	
1	[-1, 1, nn.Unsample, [None, 2, ‘nearest’]]
2	[[-1, 6], 1, Concat, [1]] # cat backbone P4
3	[-1, 3, C2f, [512]] # 12
4	
5	[-1, 1, nn.Upsample, [None, 2, ‘nearest’]]
6	[[-1, 4], 1, Concat, [1]] # cat backbone P3
7	[-1, 3, C2f, [256]] # 15 (P3/8-small)
8	
9	[-1, 1, Conv, [256, 3, 2]]
10	[[-1, 12]. 1, Concat, [1]] # cat head P4
11	[-1, 3, C2f, [512]] # 18 (P4/16-medium)
12	
13	[-1, 1, Conv, [512, 3, 2]]
14	[[-1, 9], 1, Concat, [1]] # cat head P5
15	[-1, 3, C2f, [1024]] # 21 (P5/32-large)
16	[[15, 18, 21], 1, Detect, [nc]] # Detect(P3, P4, P5)

It then instinctively shows that there is a problem with the bad ability to recognize one of the smaller objects or dimensions (width and height) that is smaller than a given scale. This study introduces a small object detection layer (160*160 recognition function for identifying goals via 4*4) to improve detection performance of small targets.

To achieve this improvement, we retain the original results in the backbone part, but adapt the model structure of the headboard. See below.

Optimizer: New detection head.

1	[-1, 1, nn.Upsample, [None, 2, ‘nearest’]]
2	[[-1, 2], 1, Concat, [1]] # cat backbone P3
3	[-1, 3, C2f, [128]] # 18 (P2/4-xsmall)

2.4. Dataset

Typically, two different data records are applied to test the performance of your network. The first is SOD data record (small object recognition) [11]. This is a collection of images that are specifically curated and annotated for small object detection tasks. Small object detection is intended to identify and highlight the most visually distinctive objects or regions of an image. Images for such data records are scaled to 640*640, with the average size of

the object being about 25*25. This data record is trained on several models with epoch = 30, image size = 640, stack size = 3. The second is the bacteria l colony data record [12]. This is a collection of images that are particularly concentrated in bacterial colonies grown in the laboratory field. This is usua lly used in microbiological and bioinformatic studies to explore bacterial growth patterns, analyze colony properties, and develop automated algorithms to detect colonies and classification. Images of such data records are scaled to 1280*1295. Various large bacteria (approx. 10*10) and small bacteria (a pprox. 2*2) form each image. This data record is trained on several models with epoch = 15, image size = 640, stack size = 10.

We also performed Yolov3 and Yolov5 validations on various data records to assess the performance of the optimized Yolov8n network. This allowed us to analyze the differences in validation speed and accuracy between the three models. We compared Yolov8 with Yolov3 and Yolov5 from a structur al and parametric perspective to determine the advantages and disadvantages of the modified Yolov8n network. This comprehensive comparison allows us to assess the practical value of the optimized model. It is important to recognize that a modified Yolov8n network can surpass only other networks in certain circumstances. This change focuses on adding a small object detection layer to extract more flatter features, which can result in poor performa nce in normal cases. Therefore, the purpose of this comparison is to further explore the niche that the model distinguishes and distinguishes between its future developments and its potential.

3. Result and discussion

3.1. Performance

Recall rate, accuracy, and map are three important criteria for object recognition. Therefore, this paper highlights how important it is to compare these metrics to assess the performance of an object recognition model. A recall rate refers to the percentage of actual objects in a photo that the model correc tly recognizes. Accuracy refers to the percentage of recognized objects correctly and is not misrecognised. Maps (medium average accuracy at 50% inte rsections via union) is a way to summarise and recall accuracy through several classes of object recognition tasks, providing an overall view of the performance of the model. The P-R curve of the model is shown in Figure 4.

3.1.1. Comparison with YOLOv8n network

The Yolov8n network has been experienced by many optimizations, and the results below have been encouraging: Comparing the optimized network with the original Yolov8n model, its performance metrics showed a clear improvement. A visual comparison with Yolov8 can be found in Figure 5.

A wide range of metric comparisons are shown in Figure 6 and Table 1,particularly when improved models were trained on SOD data records, especially when significant improvements were observed in both training speeds during the first 5 epochs. The final accuracy of the optimized network is 92.4%, a 4% improvement compared to the original network. After optimizing the Yolov8n network, the highest reall rate increased by 4% to 73.4%. After adding detection heads, the MAP50 rose from 74.2% to 78.4%. This means that the optimized network can predict the location of objects in greater detail.

It should be emphasized that these improvements were not only limited to accuracy metrics, but also occurred at training speed. In the first five epochs, the MAP50 for the optimized network is about 10% higher than the original network. At the same time, the accuracy and recall rate of the optimized network increased faster, further highlighting the advantage over the original version. Due to the number of computing power limits and training epochs , network improvements are not important. The authentic image of the SOD dataset is 1280*1295 pixels. However, to reduce the calculation, the image was compressed to 640*640 pixels. Therefore, the improvement was not noticeable. Furthermore, loss of training and loss of validation showed no improvement. The optimized network appears to have some flaws when applied to different data records. Optimized networks show no improvement when trained with bacterial colonydata records. This should be further investigated due to small amounts of datarecords and training time.

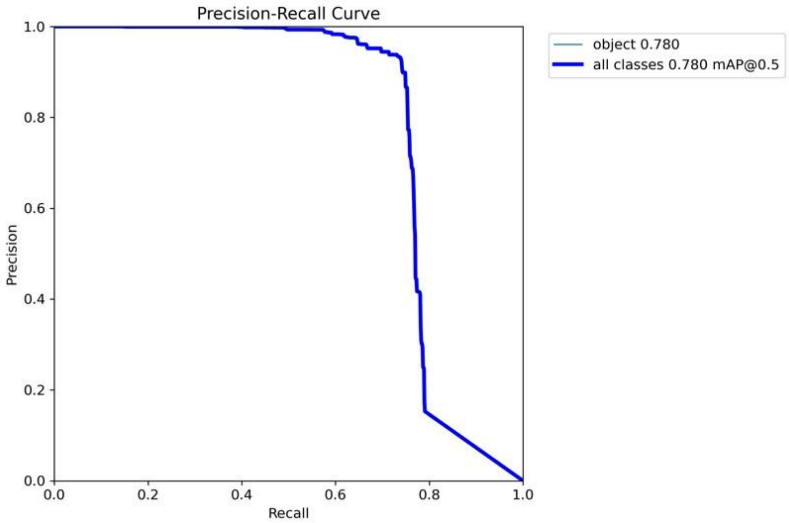
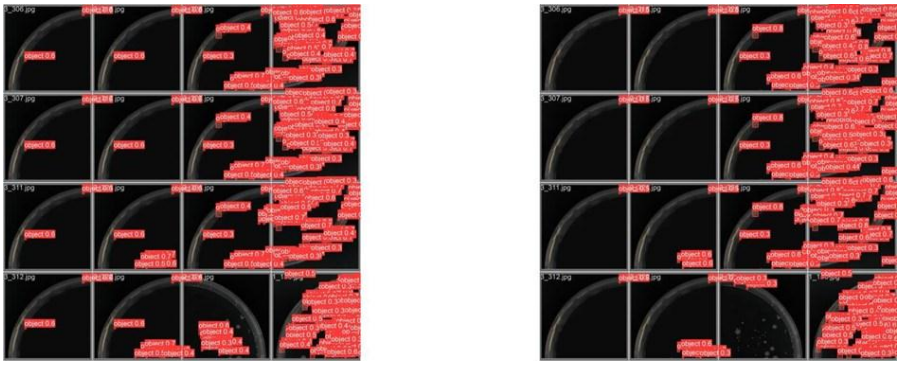


Figure 4. Precision-Recall curve.



(a) (b)

Figure 5. Detection results on bacteria colonies. (a) by optimized YOLOv8. (b) original YOLOv8

3.1.2. Comparison with other detection networks

There are major improvements compared to previous versions of the Yolo network. After training 15 epochs on the SOD data records, the Yolov5n has a recall rate of 65.5%, and the optimized Yolov8n is at a rate of 72.4%, which is more obvious. The accuracy and map improvements are also higher than previous comparisons. This is due to the performance of Yolov8n itself. The training speed of the original yolov8n network corresponds roughly to the training speeds of yolov5n and yolov3. Therefore, it proves that the network training speed actually increased with early training. However, regarding network loss, the optimized Yolov8n network showed even higher losses than previous versions of the Yolo model. This indicates that the robustness of the optimized network has not improved after optimization.

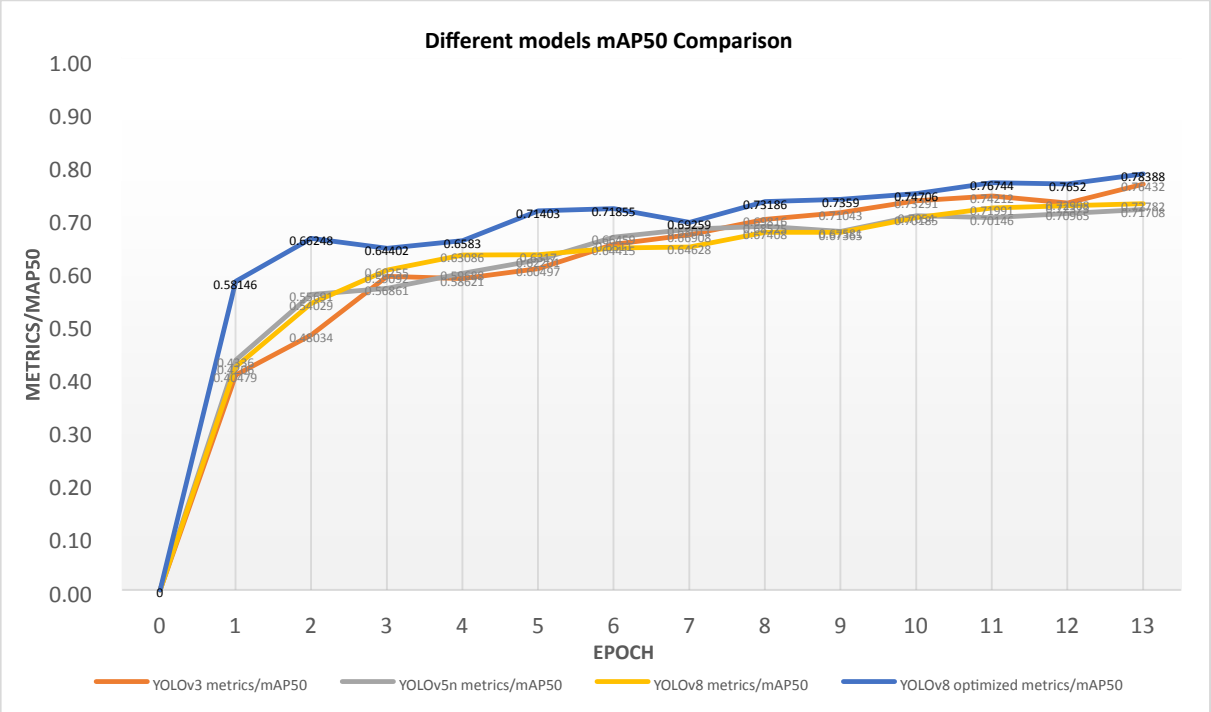


Figure 6. Different YOLO models' mAP metric curves.

Table 1. Other metrics of YOLO models.

Model	train/box_loss	validate/box_loss	metrics/recall	metrics/precision
YOLOv3	1.6007	1.5243	0.72661	0.91641
YOLOv5n	1.7389	1.6979	0.65559	0.88717
YOLOv8 original	1.6547	1.5848	0.69713	0.88717
YOLOv8 optimized	1.7455	1.7082	0.72483	0.92441

3.2. Application

Optimized lesions that allow you to recognize small, dense objects can quickly bring about great benefits when they actually become widely used. One of the most promising areas of application is autonomous vehicles. Low volume and low speed make it suitable for real-time detection of self-driving cars. Street signs and pedestrians are sometimes too small to recognize. To ensure people's safety, it is necessary to efficiently recognize road conditions that contain many small objects. This model can be used in medical diagnosis. Small tumors, lesions, or other abnormal structures are difficu lt to recognize through mere eye or traditional object detection models. Early diagnosis results in a higher cure rate for these diseases. However, medica l photographs representing these diseases are often low. Therefore, the cognitive model presented in this paper can help to increase the cure rate. Optimized models also help you recognize objects in satellite air photos. Satellite air photography is often used for weather forecasting, forest fire moni toring and other disaster predictions. The objects in satellite air photographs are relatively small, allowing the original lie model to be detected. Howeve r, optimized networks are good for processing these images.

3.3. Expectation

You can further improve your Yolov8n network. Improvements were limited by training epochs and data records and were not sufficient. Better data re cords, such as photos with objects less than 4*4 pixels, can increase the optimization effect. An increased number of trainings can also improve the perf ormance of the model. Too many detection heads can cause side effects. There may be too many bounding boxes. Another exam will allow you to focus on changing the connection between the head and the network backbone. Changing the structure of the conversion level of a feature is also an effective way to optimize your network.

4. Conclusion

To improve Yolov8 performance, this paper adds a detection head to the model's head while the backbone structure is preserved. As a result, the modifi ed model can find small objects with only 4*4 pixels. Compared to the original Yolov8 model, our model showed a 4.2% higher accuracy rate and a 4.0 % recall rate in bacterial colony detection, and a 9.2% increase with regard to the card. In fact, the model can visually recognize almost any colony. This means that the model reaches its main goal, that is, something is counted. This experiment proves that adding a specific detection head improves Yolov8's ability to recognize small objects. The model can be used in several fields. B. Calculations of power transport rivers using satellite cameras include pursuing the growth of bacterial colonies. However, if this task has too many recognition heads, it will likely slow down the training and infere nce process. This paper suggests that removing detection heads and making them a specific model of a particular task can be a good choice. Or, in this paper, it is a simple level before the input area that receives input requests and automatically changes the identification head. In another aspect, this mo del does not take into account object coverage. This is extremely difficult for small ones and is a considerable step towards the end goal of counting san d. This paper leaves it for future research.Authors Contribution

All the authors contributed equally and their names were listed in alphabetical order.

References

[1] Felzenszwalb, P.F.,, Girshick, R.B.,, McAllester, D., and Ramanan, D., 2010. Object detection with discriminatively trained part-based models. IEEE Transactions on Pattern Analysis and Machine Intelligence, 32(9), pp.1627–1645.

[2] Jiao, L.,, Zhang, F.,, Liu, F.,, Yang, S.,, Li, L.,, Feng, Z., and Qu, R., 2019. A survey of Deep Learning-based object detection. IEEE Access, 7, pp.128837–128868. Redmon, J. et al. (2016) 'You only look once: Unified, real-time object detection', 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) [Preprint]. doi:10.1109/cvpr.2016.91.

[3] Girshick, R.,, Donahue, J.,, Darrell, T., and Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. 2014 IEEE Conference on Computer Vision and Pattern Recognition.

[4] Girshick, R., 2015. Fast R-CNN. 2015 IEEE International Conference on Computer Vision (ICCV).

[5] Ren, S.,, He, K.,, Girshick, R., and Sun, J., 2017. Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(6), pp.1137–1149.

[6] He, K.,, Gkioxari, G.,, Dollár, P., and Girshick, R., n.d. Mask R-CNN. Computer Vision and Pattern Recognition.

[7] Redmon, J.,, Divvala, S.,, Girshick, R., and Farhadi, A., 2016. You only look once: Unified, realtime object detection. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[8] Gongguo, Z., and Junhao, W., 2021. An improved small target detection method based on Yolo V3. 2021 International Conference on Electronics, Circuits and Information Engineering (ECIE).

[9] Lin, Z., Feng, M., Santos, C.N.D., Yu, M., Xiang, B., Zhou, B. and Bengio, Y., 2017. A structured self-attentive sentence embedding. 2017 International Conference on Learning Representations

[10] Lin, T.-Y.,, Dollar, P.,, Girshick, R.,, He, K.,, Hariharan, B., and Belongie, S., 2017. Feature Pyramid Networks for Object Detection. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[11] Ma, Z.,, Lei Yu, and Chan, A.B., 2015. Small instance detection by integer programming on object density maps. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[12] Zhang, B.,, Zhou, Z.,, Cao, W.,, Qi, X.,, Xu, C., and Wen, W., 2022. A new few-shot learning method of bacterial colony counting based on The edge computing device. Biology, 11(2), p.156.