Pay Attention, for You Only Look Once

Yang Zhili, Li Pinzhao, Dong gengshang and Xiao Shize

January 6, 2024

Abstract

This project is based on YOLOv8. Considering the common issues in the YOLO series, such as insufficient accuracy in detecting small targets and poor detection capability for fine objects, we introduced a novel attention mechanism called EMA[1]. With the goal of preserving information on each channel while reducing computational costs, this mechanism reshapes some channels into batch dimensions, grouping channel dimensions into multiple sub-features. This ensures a uniform distribution of spatial semantic features in each feature group, ultimately enabling the capture of pixel-level pairwise relationships. The experiments were conducted on a V100-32G GPU, with a training time of 25 hours and a total of 150 epochs. The final mAP50-95 value reached 0.35. Although it is slightly lower than YOLOv8-n by 0.02 (0.37), considering that the training process did not converge (as indicated in the training results graph) and the significant difference in the number of epochs compared to the original YOLOv8 training, there is reason to believe that continuing the training of this model could result in better performance than the original YOLOv8-n. Our experimental code is available at: https://github.com/radicalyyyahaha/yolov8

Introduction

Object detection is not a novel computer vision task; it gained prominence over a decade ago with well-known algorithms such as $fast - RNN$[2] and YOLO. Despite recent trends in object detection tasks entering the public eye through applications like game cheating, its development has not waned. For instance, as of the end of 2022, YOLOv5 has iterated through five generations, and in early 2023, YOLOv8 emerged, introducing the Ultralytics framework, making it more user-friendly.

YOLOv8 represents a state-of-the-art (SOTA) model, building upon the success of previous YOLO versions and incorporating new features and improvements to enhance performance and flexibility. Innovations include a new backbone network, a novel Anchor-Free detection head, and a new loss function, allowing it to run on various hardware platforms from CPUs to GPUs.

However, Ultralytics did not directly name the open-source library YOLOv8; instead, it chose the term Ultralytics. This decision stems from positioning the library as an algorithm framework rather than a specific algorithm. Notably, one of its key features is scalability. Ultralytics aims for the library to not only be applicable to YOLO series models but also to support various tasks such as classification, segmentation, pose estimation, and more. The main advantages of the Ultralytics open-source library are its integration of various state-of-the-art technologies and its future support for other YOLO series models and algorithms beyond YOLO.

The backbone network and Neck sections may have drawn inspiration from the YOLOv7 ELAN design philosophy. They replaced YOLOv5's C3 structure with the more gradient-rich C2f structure, adjusting the channel numbers for different scale models. This meticulous fine-tuning of the model structure, tailored to different scales, significantly enhances model performance. However, certain operations in the C2f module, such as Split, may not be as hardware-friendly as before. The Head section underwent significant changes compared to YOLOv5, adopting the popular decoupled head structure, separating classification and detection heads. It also shifted from Anchor-Based to Anchor-Free. For loss computation, the TaskAlignedAssigner positive sample allocation strategy and Distribution Focal Loss were introduced. In terms of data augmentation during training, the final 10 epochs incorporated the exclusion of Mosaic augmentation, a technique from YOLOX, effectively boosting accuracy.

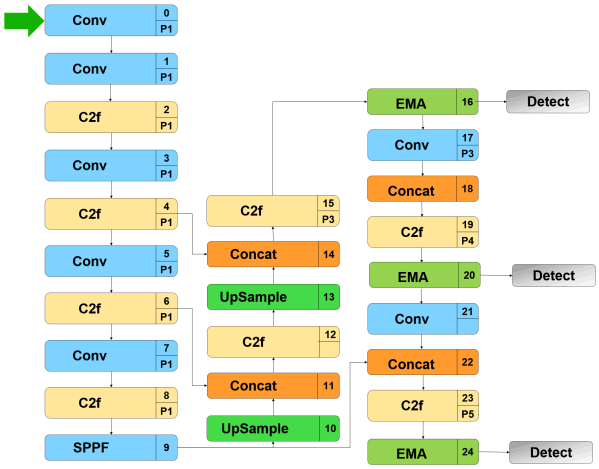

Figure 1: *Structure of YOLOv8 with EMA*

Before delving into our work, I'd like to introduce

some concepts used in the evaluation of object detection. Firstly, the Intersection over Union (IoU) is a threshold used to describe the ratio of the intersection to the union of two bounding boxes. (Image)

Next, IoU is employed to determine the values of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). TP refers to True Positive, meaning correctly predicting positive samples, and the others follow suit. Generally, when IoU is greater than a certain threshold, it is considered TP. If IoU is less than the threshold or there are too many redundant boxes predicting the same Ground Truth, it is classified as FP. FN indicates undetected instances. Building on this, we introduce the definitions of precision and recall:

- Precision is the ratio of correctly predicted positive observations to the total predicted positives.

$$Precision = \frac{TP}{TP + FP}$$

- Recall is the ratio of correctly predicted positive observations to all actual positives.

$$Recall = \frac{TP}{TP + FN}$$

Based on Precision and Recall, a Precision-Recall (P-R) curve can be plotted, where Average Precision (AP) refers to the area under this curve after averaging. The mean Average Precision (mAP) is then the sum of the AP values calculated for each class.

Precision-Recall curve provides a visual representation of the trade-off between precision and recall for different threshold values. AP quantifies the overall performance by considering the precision-recall trade-off, providing a single scalar value that summarizes the model's ability to make accurate positive predictions across various levels of recall.

In the context of object detection evaluation, mAP is often used to assess the model's overall performance across multiple classes. It reflects how well the model performs in terms of both precision and recall, offering a comprehensive measure of its effectiveness in detecting objects across diverse scenarios.

$$mAP = \frac{\sum_{q=1}^{Q} AP(q)}{Q}$$

mAP50 signifies that the evaluation is based on an IoU threshold of 50. On the other hand, mAP50-95 indicates that the mean Average Precision is calculated by averaging the performance across the IoU threshold range from 50 to 95. This range encompasses a spectrum of IoU values, providing a more comprehensive assessment of the model's object detection capabilities across various degrees of bounding box overlap. The mAP50-95 metric is particularly valuable for evaluating the model's robustness and accuracy in detecting objects with varying levels of spatial overlap.

Related Works

One major focus of our work is the integration of attention mechanisms into YOLOv8. Attention mechanisms in the field of computer vision are not new, with early proposals differentiating attention into soft and hard attention, focusing on spatial and channel thresholds, among other aspects.

As early as 2015, even before the introduction of Self-Attention, the Jaderberg team proposed a spatial threshold attention mechanism called $SpatialTransformerNetworks$[3]. This work utilized attention mechanisms to transform spatial information from the original image to another space while retaining crucial details. Subsequently, in 2017, the $SE-Net$[4] emerged, introducing the classic channel threshold attention. The innovative part of the SENet structure lies in its intermediate module, which is the attention mechanism module. This attention mechanism consists of three parts: squeeze, excitation, and attention scaling. A similar attention mechanism relevant to our project is the $CBAM$[5] introduced in 2018, an improvement based on the SE-Net network. In this context, the channel-wise attention is seen as teaching the network 'what' to look for, while spatial attention teaches the network 'where' to look. Therefore, CBAM has an advantage over the SE Module in the latter aspect. Additionally, CBAM is particularly lightweight, making it convenient for deployment at the edge.

There is limited literature on the specific improvements to the YOLO series, with more instances of researchers incorporating established techniques based on practical task requirements. For instance, there are online posts discussing the integration of a technique called $DynamicHead$[6] proposed by Microsoft into the YOLO detection head. Surprisingly, this resulted in an approximate 30-point increase in mAP. However, as we haven't reproduced this work, our perspective on this structure remains reserved. Another approach involves directly replacing some structures in YOLO with the latest $BiFormer$[7] structure from 2023, yielding promising results.

As for the addition of attention mechanisms to YOLO, most related work can be traced back a few years, where researchers integrated attention mechanisms that were not necessarily the latest in computer vision, such as CA[8] and GAM[9]. In contrast, our work involves incorporating the latest EMA attention module into the YOLOv8 head section.

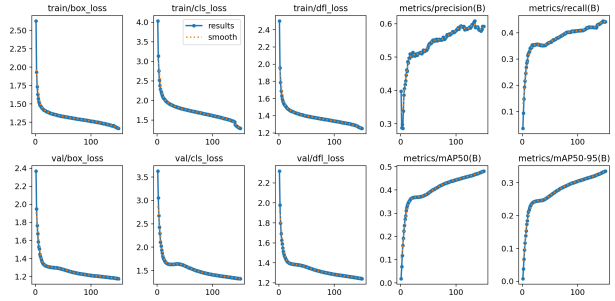

Figure 2: *Training data*

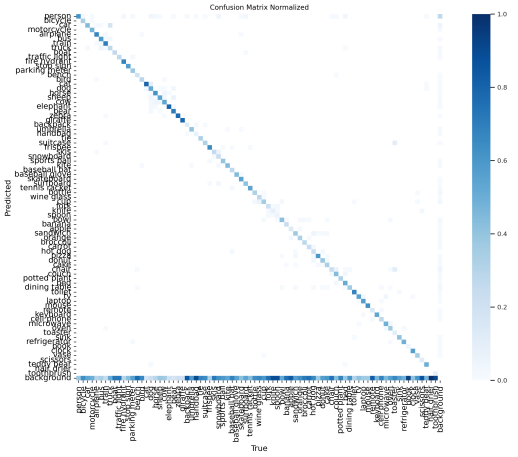

Figure 3: *Normalized confusion_matrix*

Hypothesis

Based on the successful experiences of attention mechanisms in the Natural Language Processing (NLP) domain and an analysis of the YOLOv8 structure, we have decided to incorporate the EMA attention module before the three detection heads in YOLOv8. The aim is to enable the model to acquire more robust visual representations before entering the detection module. This addition is expected to facilitate clearer feature fusion, ultimately improving the model's performance on the COCO2017 dataset beyond that of the original model.

Experiment

The training was conducted on the COCO2017 Train dataset. After removing two images with duplicate labels and one image with missing labels, the remaining data had controlled image sizes of 640x640. The training spanned 150 epochs, and the final results are depicted in Figure 2.

Three different loss values correspond to the performance of the results calculated by the three detection heads on both the training and testing sets. These losses include target box loss, category loss, and confidence loss. Until the end of training, all three losses showed a stable decrease without displaying a clear convergence trend.

Two crucial indices, mAP50 and mAP50-95, reflect the progress of model training. Results from the figure indicate an overall improvement throughout the training process, except for a small portion in the middle where a noticeable model degradation occurred. We speculate that this might be related to momentum, as the model could be sensitive to some hyperparameters, causing a slight degradation in the middle.

The reason we believe this model's final performance will surpass that of the original model lies in the last two mAP charts. Despite training for only 150 epochs and achieving a final mAP50-95 value of 0.35, which is 0.02 lower than the original model's 0.37, the training chart suggests that it has not shown signs of convergence. This implies that continuing the training will likely yield further improvement. Therefore, if used as a pretrained model and continued with the same training configuration, the final performance is expected to exceed initial expectations.

Confused Matrix

The confusion matrix is a visualization technique to illustrate the performance of an algorithm. Each row in the matrix represents a true class, and each column represents the predicted class. From the above figure, it can be observed that the horizontal axis represents True, and the vertical axis represents Precision. Each number in the chart corresponds to TP, FP, FN, and TN, from which various evaluation metrics such as Precision (P), Recall (R), F1, and more can be derived.

After completing the training, we used the best.pt to generate Figure 3, representing the degree of confusion between predicted values and ground truth. We observed that the prediction accuracy for utensils was not sufficient. We speculate that this may be related to the YOLOv8 backbone feature extraction block. Utensils often accompany various other items (tables, cabinets, snacks, meals, etc.), and the encoder layer may struggle to effectively extract features of these items. Even with the addition of attention during feature fusion, it seems insufficient, causing the model to focus on the background without effectively extracting meaningful features. This ultimately results in poor detection performance for this category of objects.

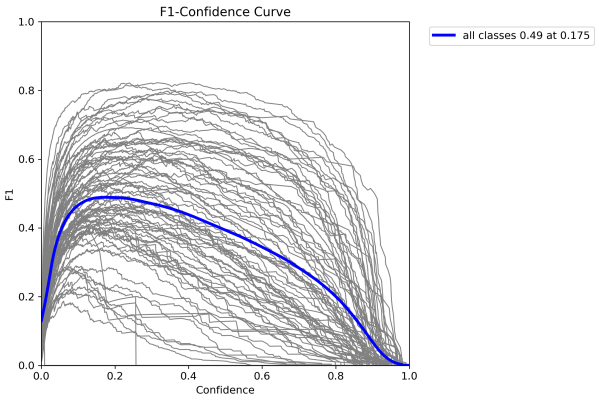

Figure 4: *F-1 score*

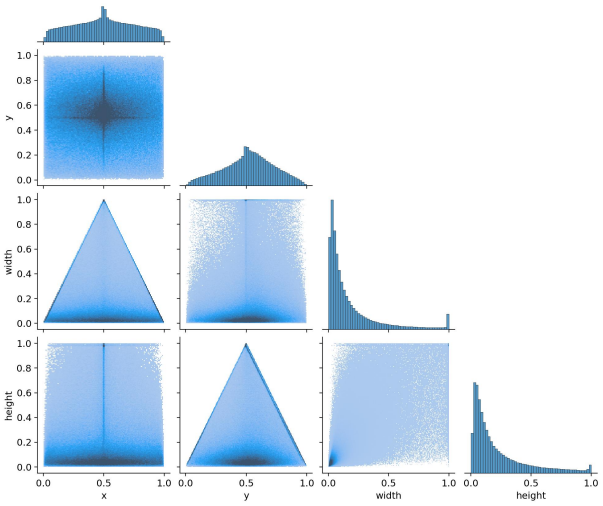

Figure 5: *label*

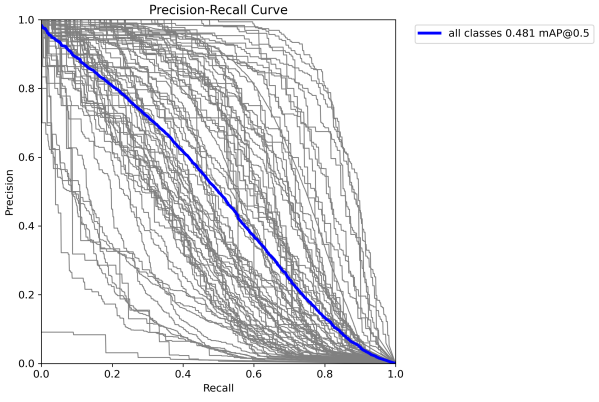

Figure 6: *P-R curve*

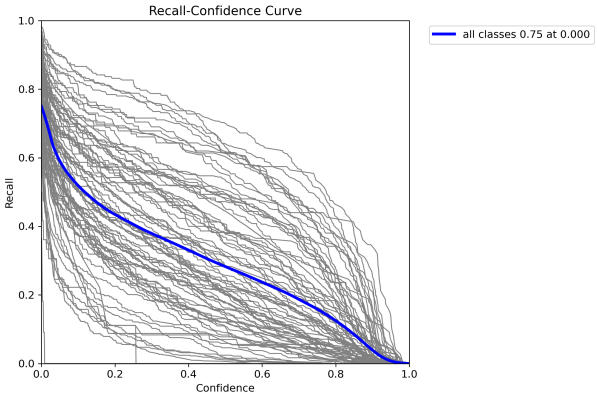

Figure 7: *R curve*

Other Index

F1-score is a metric for measuring classification performance. It is the harmonic mean of precision and recall, ranging from 0 to 1, with 1 being the best and 0 the worst. In simple terms, F1-score aims to simultaneously control recall and precision to evaluate the model's effectiveness. Our results are presented in Figure 4.

Figure 5 is a color matrix that illustrates the correlation between predicted labels during the training process of the object detection algorithm. The matrix's rows and columns represent the labels (classes) used during model training, and each cell represents the correlation between the predicted results for the corresponding labels. Darker colors in the matrix indicate stronger correlations between the corresponding labels, while lighter colors indicate weaker correlations. Colors along the diagonal represent the self-correlation of each label, typically appearing the darkest.

Figure 6 is a PR Curve, which stands for Precision-Recall Curve. It depicts the relationship between precision and recall at different thresholds. Precision (P) represents the proportion of true positives among the samples predicted as positive, while recall (R) represents the proportion of true positives among the actual positive samples. In the PR Curve, the horizontal axis represents recall, and the vertical axis represents precision. Generally, when recall is high, precision is low, and vice versa. The PR Curve reflects this trade-off relationship. When the PR Curve is closer to the top-right corner, it indicates that the model can simultaneously ensure high precision and high recall during predictions, meaning more accurate predictions. Conversely, when the PR Curve is closer to the bottom-left corner, it suggests that the model struggles to ensure both high precision and high recall during predictions, resulting in less accurate predictions.

Recall-Confidence Curve (RCC) is a method used

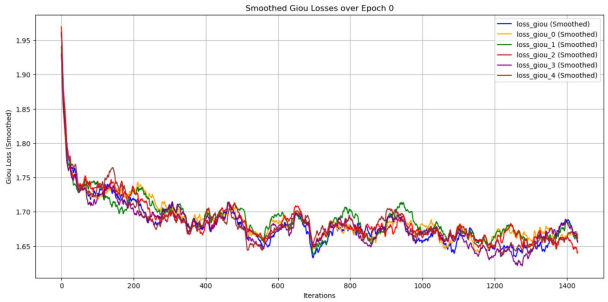

Figure 8: *Loss*

in object detection to evaluate algorithm performance. It visually represents the variation of recall at different confidence thresholds. Typically, we aim for an algorithm to maintain high precision while achieving high recall. When the curve in the RCC graph exhibits high recall at a relatively high confidence level, it indicates that the algorithm can accurately predict the presence of targets during object detection. Moreover, even after filtering out predictions with low confidence, it can still maintain a high recall rate. This suggests that the algorithm performs well in object detection tasks. In Figure 7, the curve has a steep slope, indicating a significant improvement in recall after filtering out predictions with low confidence. This, in turn, enhances the model's detection performance.

More On Attention

$DETR(DetectionTransformer)$[10] is a model used for object detection that adopts the Transformer architecture. In object detection tasks, Intersection over Union (IoU) is a common metric for measuring the similarity between predicted bounding boxes and true bounding boxes. However, IoU has certain limitations, especially when two bounding boxes do not intersect, IoU cannot effectively reflect their distance and shape differences. To overcome these limitations, the original text introduces a variant of IoU, Generalized IoU (GIoU); while our group uses Distance IoU (DIoU). We conducted experiments to compare the detection effects of these two metrics.

We hypothesize that the quality of bounding boxes depends not only on their overlap but also significantly on the distance between their center points. The closer the center points of the bounding boxes, the higher their localization accuracy usually is. Therefore, we replaced GioU with DioU in the DETR model and conducted comparative experiments to verify our conclusions. In the end, the effect of DioU was not as good as that of GioU, which may be due to our insufficient training volume.

Our work showed on Figure8, we only trained for one epoch, so we compared the loss changes of diou and giou under different iterations in the same batch. From the graph, we can see that diou training was successful because the loss was continuously decreasing. However, it is still higher than the loss of giou. This may be due to our insufficient training batches, or it may be that our hypothesis does not hold.

Conclusiton

Object detection algorithms are continually evolving, especially with the annual advancements and the notable impact of attention mechanisms introduced in the NLP domain. These mechanisms have brought immeasurable improvements to object detection tasks. We attempted to incorporate attention mechanism modules into YOLO, enabling better feature fusion before the detection heads, and the anticipated results have been promising. Future work could explore the addition of more attention mechanism modules within YOLO's backbone to enhance its ability to extract target features more effectively.

Reference

1. Ouyang, D., He, S., Zhang, G., Luo, M., Guo, H., Zhan, J., & Huang, Z. (2023, June). Efficient Multi-Scale Attention Module with Cross-Spatial Learning. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 1-5). IEEE.
2. Girshick, R. (2015). Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision* (pp. 1440-1448).
3. Jaderberg, M., Simonyan, K., & Zisserman, A. (2015). Spatial transformer networks. *Advances in neural information processing systems*, *28*.
4. Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7132-7141).
5. Woo, S., Park, J., Lee, J. Y., & Kweon, I. S. (2018). Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 3-19).
6. Dai, X., Chen, Y., Xiao, B., Chen, D., Liu, M., Yuan, L., & Zhang, L. (2021). Dynamic head: Unifying object detection heads with attentions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 7373-7382).
7. Zhu, L., Wang, X., Ke, Z., Zhang, W., & Lau, R. W. (2023). BiFormer: Vision Transformer with Bi-Level Routing Attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 10323-10333).
8. Hou, Q., Zhou, D., & Feng, J. (2021). Coordinate attention for efficient mobile network design. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 13713-13722).
9. Liu, Y., Shao, Z., & Hoffmann, N. (2021). Global attention mechanism: Retain information to enhance channel-spatial interactions. *arXiv preprint arXiv:2112.05561*.
10. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., & Zagoruyko, S. (2020, August). End-to-end object detection with transformers. In European conference on computer vision (pp. 213-229). Cham: Springer International Publishing.

Contribution

Yang Zhili: 40% responsible for YOLO structure and the most of report

Lin Pinzhao: 25% responsible for Detr and some part of report

Dong Gengshang: 25% responsible for Detr and some part of report

Xiao Shize: 10% responsible for some other works.