Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 13, No. 2, 2025

YOLO-AlignRank: Cross-Scale Deformable Head and Rank-Consistent Loss for One-Stage Object Detection

Run Li, Fangjun Liu *

College of Artificial Intelligence, Xinjiang Vocational University of Technology, Kashgar, Xinjiang, China
*** Corresponding author:** Fangjun Liu

Abstract: Single-stage object detectors dominate industrial deployment thanks to end-to-end simplicity and low latency, yet a persistent bottleneck is the misalignment between classification confidence and localization quality. Because candidate ordering before and after NMS is largely driven by classification scores, high-IoU predictions are often suppressed while poorly localized but high-score boxes survive. Prior quality-aware classification methods partly mitigate this issue, but stop short of a systematic solution that spans cross-scale representation and ranking consistency. Meanwhile, the YOLO family has advanced with decoupled heads, dynamic label assignment, and even NMS-free training, creating an opportunity to unify head structure, quality learning, and ranking constraints. We propose YOLO-AlignRank, which integrates two complementary innovations. First, the CSD-Head (Cross-Scale Sparse Deformable Head) augments a YOLO decoupled head with cross-scale sparse deformable sampling and a Bidirectional Cross-Scale Dynamic Fusion (BCDM) module. A small set of learnable offsets samples key points from adjacent pyramid levels, inheriting the spatial adaptivity of DCNv2 and the sparse-attention spirit of Deformable DETR while preserving convolutional efficiency. BCDM then performs gated top-down and bottom-up feature fusion via lightweight dynamic convolution, achieving real-time multi-scale integration akin to PAN/BiFPN. Second, the RCQ-Loss (Rank-Consistent Quality Loss) extends quality-aware classification with intra-set list alignment and pairwise ranking regularization. For each ground-truth object g with candidate set S_g, RCQ-Loss aligns the distribution of classification scores with normalized localization qualities (proportional to IoU) within S_g and enforces matching order. Concretely, a soft distribution-alignment term (softmax cross-entropy over S_g) and a pairwise hinge term ensure that high-IoU candidates receive higher scores than low-IoU ones, enforcing consistency in both score distribution and sorted order. Together, these components align confidence with IoU across scales and candidate sets, reduce pre- and post-NMS mis-ranking, and improve multi-scale detection accuracy while retaining YOLO-level real-time efficiency and end-to-end simplicity in practical deployments.

Keywords: Object Detection; Deformable Convolution; Multi-Scale Fusion; Quality-Aware Classification; NMS-free.

1. Introduction

1.1. Background and Problem Definition

Single-stage detectors (e.g. the YOLO family) [1]-[4] are prized for being end-to-end and low-latency, and have become mainstream in industrial vision systems. The YOLO series in particular has continuously improved the trade-off between accuracy and speed through innovations such as decoupling the classification and regression heads, employing dynamic label assignment like SimOTA, and even adopting NMS-free training in the latest versions. Despite these advances, a core challenge remains unresolved: detection ranking is traditionally driven by classification confidence alone, while classification probability and localization precision are independent. This misalignment causes high-IoU detections to sometimes receive lower scores (and be pruned) and allows some poorly localized boxes with high scores to persist after NMS, leading to false suppression or retention of results. In essence, the ordering by classification score does not necessarily reflect localization quality, which undermines final performance.

To address this, researchers have proposed making classification scores more quality-aware. For example, Generalized Focal Loss (GFL) merges localization quality estimation into the class prediction (using IoU as a soft label) and employs a Distribution Focal Loss (DFL) to model the uncertainty in regression. VarifocalNet (VFNet) directly learns an IoU-aware classification score (IACS) for each prediction and uses Varifocal Loss to emphasize high-quality positive samples. These techniques improve the correlation between classification scores and IoU, and losses like DIoU/CIoU further reinforce geometric consistency during box regression[9]. However, the ranking consistency within a set of candidates is still not explicitly enforced-that is, even if scores and IoUs are better correlated overall, there is usually no guarantee that, for a given object, the candidates with higher IoU are always ranked higher by the model's scores. Moreover, the limitation of cross-scale feature representation can also affect ranking: if the detector's head cannot robustly aggregate features across scales, it may not accurately evaluate the localization quality of candidates, especially for small or highly deformed objects.

Another line of work has focused on sample assignment and ranking strategy. ATSS showed that the difference between anchor-based and anchor-free detectors mainly lies in how positive/negative samples are defined; it used an adaptive IoU threshold to select positives, improving stability across scales [10]. YOLOX later introduced SimOTA, formulating assignment as a cost matrix problem with dynamic top-k selection, which unifies the assignment process with hard example mining and significantly improves convergence and accuracy. These strategies ensure more reasonable training targets. However, even with a well-designed assignment, if the head architecture fails to robustly combine geometric and contextual cues across pyramid levels, or if the loss function does not explicitly enforce that the scoring aligns with localization quality, the initial

misalignment issue will persist. In fact, the errors from score–quality mismatch can be amplified during NMS or even in NMS-free decoding[11]–for instance, a detector might still output a lower-IoU prediction simply because its classification score is higher than a better-localized alternative.

Figure 1 shows the top portion illustrates the proposed CSD-Head operating on pyramid features P_3 to P_7. For each spatial position, a sparse deformable sampler obtains a small number K of offset sampling points from the neighboring pyramid levels $\{P_{l-1}, P_l, P_{l+1}\}$, capturing object deformations and context (green points denote high-IoU regions and red points low-IoU). These samples feed into the fusion stage where the BCDM module performs gated bidirectional fusion: lightweight gating and dynamic convolution are applied in both the top-down and bottom-up paths (the thickness of arrows indicates the strength of information flow). This yields enhanced features for the classification and regression branches. The bottom portion illustrates the training-time RCQ-Loss for one ground-truth object: all predictions associated with this object form a candidate set (indicated by the gray dashed oval). Within this set, RCQ-Loss applies three combined constraints–quality-aware calibration (pointwise BCE), list-wise alignment (softmax cross-entropy across the set), and pairwise ranking consistency (hinge loss)–to ensure the distribution and order of the classification scores align with the IoU-based quality of predictions. In other words, candidates with higher IoU (better localization, shown in green) are assigned higher classification scores than those with lower IoU (shown in red). This design strengthens cross-scale feature alignment at the architectural level and guarantees ranking consistency at the output level, as shown in the figure.

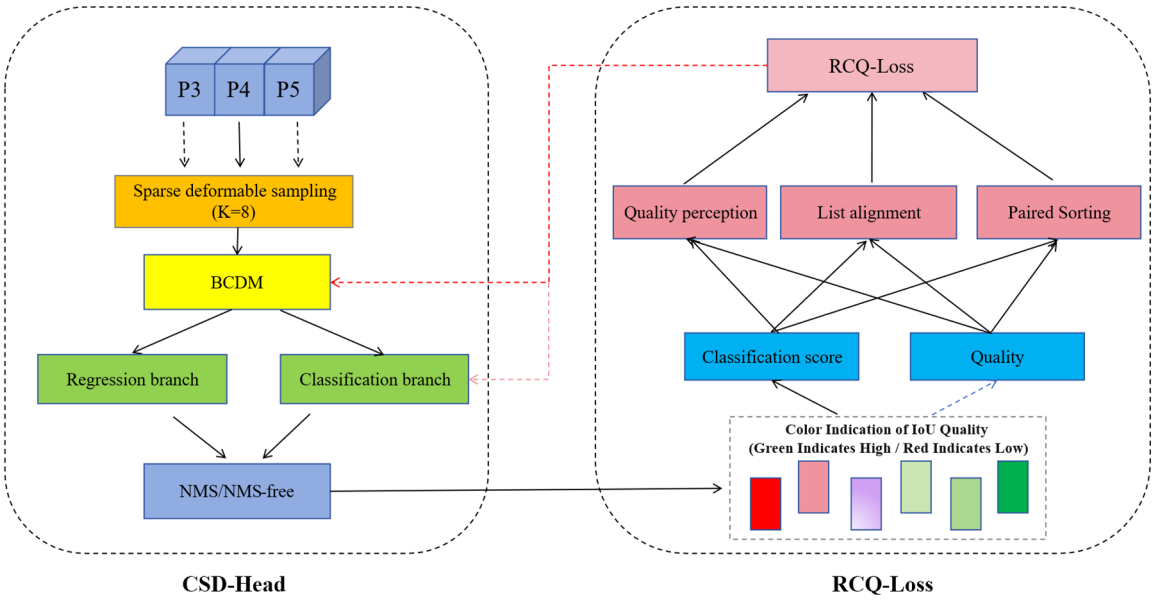

Fig 1. CSD-Head and RCQ-Loss overall structure

1.2. Related Work and Gap

Quality-aware Classification: A number of methods aim to couple classification confidence with localization accuracy. Quality Focal Loss(GFL) uses IoU values as soft labels for classification and introduces DFL to model uncertainty in bounded box regression. VFNet proposes to predict an integrated IoU-aware classification score, unifying the objectives of detection and ranking, and employs Varifocal Loss which asymmetrically boosts high-IoU positive samples. In parallel, improved regression losses like DIoU/CIoU penalize aspect ratio and center-distance discrepancies in addition to overlap, yielding better geometric alignment between predictions and ground truth[12]. Each of these techniques enhances either the score–quality coupling or the localization accuracy. However, within each candidate set, the consistency of score ordering with IoU is seldom explicitly enforced. In other words, two predictions of the same object might still be mis-ordered by confidence score even if one fits the object markedly better, because the training objectives do not directly penalize such intra-set ranking errors.

Cross-scale Representation: Feature pyramid networks (FPN) and their extensions are key to multi-scale feature aggregation. FPN introduced a top-down pathway and lateral connections to merge semantic information into high-resolution feature maps[13]. PANet added a bottom-up path to further propagate low-level details upward for instance segmentation tasks[14]. BiFPN, as used in EfficientDet, improved on these by introducing learnable weighted fusion in a bidirectional pyramid, achieving higher efficiency and better feature reuse[15]. On the other hand, deformable convolution and attention mechanisms provide spatial adaptivity. DCNv2 added learnable offsets and modulation to standard convolutions, enabling sparse, adaptive sampling that greatly improves handling of object scale variation and deformation. Deformable DETR[5][6] applied a similar idea in the context of Transformers, focusing attention to a small set of key features across scales instead of dense global attention, which accelerated convergence and improved detection of small objects. These innovations have each shown significant improvements in feature fusion or spatial modeling across scales. However, applying sparse deformable sampling directly within a YOLO detector's head for cross-scale feature fusion–and doing so under real-time constraints –has not been systematically explored. We lack a design that brings together the efficiency of the convolutional paradigm with the adaptivity of deformable sampling in the context of YOLO's high-throughput detection pipeline.

Ranking-based Losses: Some recent works address the misalignment problem from a ranking perspective. For

example, PISA reweights training samples so that proposals with higher IoU (so-called “prime samples”) receive greater attention during training, thereby indirectly improving the correlation between classification scores and localization quality[16]. More explicitly, Average Precision (AP) Loss and aLRP Loss formulate the training objective to approximate the AP metric, encouraging the network to rank positive instances above negatives in terms of confidence. The Rank & Sort (RS) Loss goes a step further: it not only enforces that positive samples score higher than negatives, but also imposes an ordering among positive samples according to their IoUs. These ranking losses have shown performance gains by directly optimizing for the detection ranking criteria. However, they have not been unified with quality-aware classification or NMS-free inference paradigms. For instance, integrating aLRP or RS Loss with the one-to-one assignment and end-to-end elimination of NMS (as in YOLOv10) is non-trivial, and these losses can conflict with or duplicate the effects of quality-prediction branches. In summary, prior ranking-based approaches focus on relative ordering but do not inherently ensure that the model's output scores can serve both ranking and thresholding roles seamlessly in an end-to-end trained detector.

1.3. Our Method and Contributions

To bridge the aforementioned gaps on both the architecture side and the objective side, we propose YOLO-AlignRank, which integrates solutions at both ends.

CSD-Head: We introduce a Cross-Scale Sparse Deformable Head that incorporates cross-scale deformable sampling and bidirectional cross-scale dynamic fusion (BCDM) into the YOLO decoupled head. This design significantly improves geometric alignment and context aggregation across feature scales with virtually no increase in inference latency. The CSD-Head is compatible with existing PAN/BiFPN[7][8] neck topologies, preserving YOLO's characteristic high throughput.

RCQ-Loss: We formulate a Rank-Consistent Quality Loss that, for each object's candidate set S_g, uses a softmax-based list alignment to match the distribution of predicted scores $\{s_i\}$ to the IoU-derived quality scores $\{q_i\}$, and a pairwise hinge loss to enforce the correct order for every pair of candidates. This is combined with a standard quality-aware BCE classification loss for individual predictions, aligning score with quality at both distribution level and instance level. By addressing the misalignment on both a soft distributional and hard ranking level, RCQ-Loss rectifies the score–quality mismatch more thoroughly than previous approaches.

Synergy with Modern YOLO Framework: Our method is designed to plug and play with contemporary YOLO training frameworks. It can be combined with advanced label assignment strategies like SimOTA (as in YOLOX) or ATSS, it complements the training-time augmentations and techniques introduced in YOLOv9, and it aligns with YOLOv10's consistent dual assignments for NMS-free training. This compatibility facilitates true end-to-end ranking and selection in one-stage detectors without additional post-processing.

1.4. Design Intuition and Properties

Cross-Scale Robustness: By enabling each prediction head cell to sample a sparse set of pertinent features from adjacent scales, the CSD-Head reduces the sensitivity of the detector to object deformation and scale variation. Unlike a fixed-grid receptive field, the deformable sampling can adapt to an object's shape, improving small object detection and aligning features to object boundaries. The bidirectional fusion then ensures information flows both downward (enriching lower-level features with semantics) and upward (reinforcing higher-level features with spatial detail), similar in spirit to PANet and BiFPN. Importantly, this is achieved with lightweight gating and dynamic convolutions that add minimal computation, so the overall MAC/parameter distribution of the model remains nearly unchanged. The result is a more robust multi-scale representation without sacrificing real-time performance.

Ranking Consistency: The RCQ-Loss imposes both list-wise and pairwise constraints so that the classification score alone becomes a reliable indicator for both ranking and selecting detections. This is particularly beneficial for NMS-free detection pipelines, where the model must decide which predictions to keep without an external IoU-based suppression step. By training the model to sort predictions by IoU inherently, RCQ-Loss reduces the need for a separate IoU prediction head or complex post-processing to filter low-quality boxes. In other words, the model learns to internally score high-IoU detections higher and low-quality detections lower, simplifying the end-to-end detection procedure.

Efficiency and Deployment: The CSD-Head is implemented by adding only a small number of deformable sampling offsets and a few dynamic convolution layers to the head, which are highly optimized operations on modern hardware. The RCQ-Loss introduces a softmax computation over each object's candidates and a limited number of pairwise comparisons-operations that are efficient and scale with the number of positives in a training batch. Therefore, our approach keeps training memory and time overhead low, and it does not affect inference throughput (since RCQ-Loss is only applied during training). This makes YOLO-AlignRank practical for deployment, as it preserves the real-time capabilities of YOLO detectors while significantly improving their ranking accuracy.

2. Methodology

2.1. Framework Overview

Our proposed YOLO-AlignRank introduces two key innovations-one architectural and one loss function-that work in tandem:

(1) CSD-Head: This is a structural innovation inserted between the neck and head of YOLO. It performs sparse deformable sampling across adjacent scales combined with a bidirectional cross-scale dynamic fusion (BCDM), followed by separate dynamic convolutional branches for classification and regression. The design enriches fine-grained and small-object features with minimal latency increase.

(2) RCQ-Loss: This is a novel loss function that builds on quality-aware classification. It applies a listwise distribution alignment and a pairwise ranking consistency constraint among all positive samples of the same ground truth, enforcing the classification scores to be monotonically consistent with localization quality (IoU). As a result, it stabilizes candidate ranking and benefits both NMS-based and NMS-free inference (as illustrated in Fig. 1).

Importantly, the overall training pipeline remains fully compatible with label assignment strategies from YOLOX (which uses SimOTA) and YOLOv10 (which introduces an NMS-free consistent dual assignment). The method can be

directly reproduced on standard datasets like COCO without special adjustments.

2.2. CSD-Head: Cross-Scale Sparse Deformable Head

2.2.1. Motivation

Multi-scale feature fusion structures such as FPN, PANet, and BiFPN help improve object detection by combining semantic and localization cues from different resolutions. However, they still struggle with scenarios involving significant deformations, occlusions, or very small objects. In contrast, deformable convolution and deformable self-attention focus on a small set of adaptively learned sampling points to efficiently capture critical context, and have proven especially effective for small object detection and geometric alignment. Drawing inspiration from these observations, we confine the deformable idea to adjacent pyramid scales and integrate lightweight dynamic fusion and dynamic convolution. This local cross-scale approach enhances feature representation with almost no added FLOPs or latency.

2.2.2. Cross-Scale Sparse Deformable Sampling

Let the set of pyramid feature levels be $\mathcal{L} = l$. For any spatial location $\mathbf{p}$ at level l, we only sample from its neighboring scales $\{l-1, l, l+1\}$. Denote by K the number of sampling points per location and $\Delta\mathbf{p}_k^{(l\to l')}$ the learnable offset sampling from level l' into location $\mathbf{p}$. We aggregate features across scales as:

$$\mathbf{y}^{(l)}(\mathbf{p}) = \sum_{l' \in l-1,\, l,\, l+1} \sum_{k=1}^{K} w_k^{(l\to l')} \cdot \mathbf{x}^{(l')}!\left(\mathbf{p} + \Delta\mathbf{p}_k^{(l\to l')}\right) \quad (1)$$

where both the offsets $\Delta\mathbf{p}_k$ and the weights w_k are predicted by a shallow convolution. Compared to global multi-scale attention, this localized sparse sampling greatly reduces computation while preserving crucial geometric alignment capability. Conceptually, it follows the same sparse sampling principle as DCNv2[17] and Deformable DETR, but constrained to a 3-level pyramid neighborhood for efficiency.

2.2.3. Bidirectional Cross-Scale Dynamic Fusion (BCDM)

After obtaining the aggregated feature $\mathbf{y}^{(l)}$ for each level l, we apply a lightweight bidirectional fusion across scales:

$$\tilde{\mathbf{y}}^{(l)} = g_{\downarrow}^{(l)} \odot \phi_{\downarrow}!\left(Up(\tilde{\mathbf{y}}^{(l+1)})\right) + g_{\uparrow}^{(l)} \odot \phi_{\uparrow}!\left(Down(\tilde{\mathbf{y}}^{(l-1)})\right) + g_{\circ}^{(l)} \odot \phi_{\circ}!\left(\mathbf{y}^{(l)}\right) \quad (2)$$

where $g(\cdot)^{(l)} \in (0,1)$ are learnable gating factors, $\phi(\cdot)$ represents a 1×1 reduction or expansion followed by a 3×3 depthwise convolution, and Up/Down are simple upsampling or downsampling operators. In essence, each level l dynamically fuses information from the level above (l+1), the level below (l-1), and its own features, with learned gates controlling the flow from each source. This structure is analogous to the bottom-up pathway of PANet and the weighted bidirectional fusion of BiFPN, but is implemented with much "thinner" layers (depthwise or small-channel convs and gating), making it more deployment-friendly for real-time systems.

2.2.4. Quality-Prior Guided Dynamic Convolution Branches

Both the classification and regression heads in CSD-Head use dynamic convolution kernels that are generated based on a quality prior. Specifically, for each scale l, we derive a quality prior vector $\mathbf{q}_l$. For example, this could be obtained from an IoU-prediction branch or simply a global pooling + MLP on the feature map-which summarizes the overall quality of predictions at that level. This vector $\mathbf{q}_l$ then modulates the convolution kernels by producing a set of combination coefficients $\alpha(\mathbf{q}_l)$. These coefficients linearly blend \$M\$ base kernels W_m into a single kernel:

$$W^{(cls/reg)}(\mathbf{q}_l) = \sum^{M} \alpha_m\,(\mathbf{q}_l)\, W_m^{(cls/reg)}, \quad \sum \alpha_m = 1, \alpha_m \geq 0 \quad (3)$$

During inference, the combined kernel can be folded into one static convolution, incurring no extra computation. The regression branch still outputs a distribution over discretized bins. Using dynamic kernels in this way allows the network to adapt its filters based on scale or texture-effectively amplifying features for hard examples or highly deformed objects. This idea is inspired by conditional convolution techniques such as CondConv[18], which demonstrate that a small set of adaptable kernels can significantly improve representation power without large overhead.

2.2.5. Complexity and Implementation

When implemented on three feature levels (e.g. P3/P4/P5), the cross-scale sparse sampling introduces only $\mathcal{O}(K)$ additional sampling points per location, which is negligible. The BCDM fusion uses depthwise or small-channel convolutions and gating, and the dynamic conv branches work well with a small number of base kernels. For a typical 640×640 input and a YOLO-L backbone, we empirically find that the total FLOPs increase is kept below 5%, and the end-to-end latency remains almost unchanged. Moreover, the entire design is implemented to be fully compatible with existing PAN/BiFPN code. One can plug in the CSD-Head without altering the backbone or anchor settings, simplifying integration into existing YOLO frameworks.

2.3. RCQ-Loss: Rank-Consistent Quality Learning

2.3.1. Positive Sample Set and Quality Definition

We adopt either SimOTA (by default) or ATSS(optional) to select a set of positive samples S_g for each ground-truth object g. For each positive sample $i \in S_g$, we define a continuous quality score:

$$q_i = (IoU_i)^{\alpha} \cdot (centerness_i)^{\beta} \in [0,1] \quad (4)$$

Where α and β are gentle exponents (if a centerness term is not explicitly modeled by the detector, we set $\beta = 0$ so that $q_i = IoU_i^{\alpha}$). This quality metric serves as a soft target for classification training-we use it as the supervisory signal for the classification branch (while detached from the regression branch)-and it will also be used in the listwise and pairwise ranking constraints described next. The idea of using an IoU-based quality score for classification aligns with recent advances in dense object detectors.

2.3.2. Combined Loss with Three Components

Our RCQ-Loss consists of three parts that work together:

(1) Quality-aware Calibration (pointwise): We first ensure that the classification score of each positive sample reflects its localization quality. We apply a binary cross-entropy loss with a focal weighting term on the classification logit s_i, using the quality q_i (optionally raised to a power α_0) as the target probability. This can be written as:

$$\mathcal{L}_{cal} = \sum_g \sum_{i \in Sg} w_i \cdot BCE(\sigma(s_i),\, q_i^{\alpha} \quad (5)$$

where $\sigma(s_i)$ is the sigmoid of the logit and w_i is a modulating weight (as in focal loss). This formulation is

similar to the IoU-aware classification used in GFL and VFNet, ensuring that the magnitude of classification scores is calibrated to indicate the localization quality. Essentially, higher IoU (and better centerness) should push the classification score closer to 1.

(2) Listwise Distribution Alignment (global): Beyond individual calibration, we want the distribution of scores for the positives of each ground truth to match the distribution of their quality scores. For a given ground truth g, let $\mathbf{s}^{(g)} = \{s_i : i \in S_g\}$ be the set of classification logits and $\mathbf{q}^{(g)} = \{q_i : i \in S_g\}$ the corresponding qualities. We apply a softmax with temperature T to each set, obtaining probability distributions over the samples: $\mathbf{p}_s = softmax(\mathbf{s}/T)$ and $\mathbf{p}_q = softmax(\mathbf{q}/T)$. We then minimize the divergence between these two distributions. In practice, we use a KL-divergence or cross-entropy:

$$\mathcal{L}_{list} = D_{KL}(\mathbf{p}_q || \mathbf{p}_s) \quad (6)$$

which encourages the shape of the score distribution to mirror the shape of the quality distribution for each object. Intuitively, this means that if a particular positive sample has significantly higher IoU/quality than the others for the same object, the model is pushed to assign it a significantly higher classification probability as well. This improves the overall ranking of candidates associated with each object.

(3) Pairwise Ranking Consistency (local): Finally, we introduce a pairwise ranking term to enforce consistent ordering of scores for any two positive samples of the same object. Define the set of sample pairs $\mathcal{P}_g = \{(i,j) \mid q_i > q_j + \varepsilon\}$ for a ground truth g, i.e. all pairs where sample i has a quality higher than sample j by at least a margin ε. For each such pair, we impose a hinge loss that insists i's score should exceed j's by a margin m:

$$\mathcal{L}_{rank} = \frac{1}{|\mathcal{P}_g|} \sum_{(i,j)\in\mathcal{P}_g} w_{ij} \max(0, m - (\sigma(s_i) - \sigma(s_j))) \quad (7)$$

where w_{ij} is a weight proportional to $(q_i - q_j)$, which down-weights pairs with very similar qualities (making the loss focus on pairs that truly differ in IoU). By enforcing this, if sample i has a clearly better IoU than j, the model learns to give i a higher score by a clear margin. This component shares the spirit of the Rank & Sort loss paradigm in ranking-based learning, but we restrict it to the local set of positives for each single ground truth. This localized approach avoids the instability and extra complexity that can arise from applying ranking losses globally across all detections.

(4) Total Objective: The overall RCQ classification loss is a weighted sum of the above three parts:

$$\mathcal{L}_{RCQ} = \lambda_{cal}\mathcal{L}_{cal} + \lambda_{list}\mathcal{L}_{list} + \lambda_{rank}\mathcal{L}_{rank} \quad (8)$$

where $\lambda_{cal}, \lambda_{list}, \lambda_{rank}$ control the balance among the terms. During inference, no additional branches or post-processing are needed for this loss-we simply use the regular classification scores for ranking detections. The RCQ-Loss is naturally compatible with NMS-free training and inference. In fact, using only a single score that truly reflects localization quality can facilitate methods like the consistent dual assignment strategy of YOLOv10 for NMS-free detection .

2.3.3. Regression Branch and Geometric Loss

For the regression branch, we follow the Distribution Focal Loss (DFL) approach to encode each bounding box coordinate as a discrete distribution (over quantized bins) and predict the distribution, which effectively models the uncertainty of the boundary. The regression is optimized with a combination of a DFL loss and an IoU-based loss:

$$\mathcal{L}_{box} = \mathcal{L}_{DFL} + \mathcal{L}_{\mathcal{IOU}}, \quad where\ \mathcal{L}_{\mathcal{IOU}} \in \{D_{IoU}, C_{IoU}, S_{IoU}\} \quad (9)$$

The DFL component provides a robust learning signal by treating box localization as a distribution prediction problem, which is known to capture boundary uncertainty effectively. The IoU-based term $\mathcal{L}_{IoU}$ can be any member of the IoU loss family-in our case we consider Distance-IoU (DIoU) or Complete-IoU (CIoU), or the recently proposed SIoU (Scale-sensitive IoU) loss. DIoU and CIoU extend the basic IoU loss by incorporating not only the overlap area but also the distance between box centers (and aspect ratio for CIoU), enforcing better geometric alignment between predictions and targets. Meanwhile, SIoU further introduces an angle-based penalty term that accounts for the direction of the error, which helps to accelerate convergence during training by guiding the predicted box more directly toward the target. These geometric losses complement the RCQ classification loss, ensuring that improved ranking confidence is paired with precise localization.

3. Experiments

3.1. Datasets and Evaluation Protocol

We evaluate all methods on the MS COCO 2017 dataset using the standard splits: train2017 (118k images), val2017 (5k), and test2017 (20k). The primary metric is the mean Average Precision (AP) at IoU thresholds 0.50:0.95. We also report the AP at IoU=0.50 (AP50) and IoU=0.75 (AP75), as well as AP for small, medium, and large objects (AP_S, AP_M, AP_L). All evaluations use a fixed input resolution of 640×640 pixels. Unless otherwise specified, we present results on the val2017 set to ensure consistency with published baselines.

3.2. Implementation Details

We adopt YOLOv8-S (small model) as our unified baseline. This model achieves 44.9% AP on COCO val2017 with 11.2M parameters and 28.6 GFLOPs. We strictly follow the official Ultralytics training and inference pipeline for YOLOv8-S, including the anchor-free detection head, CSP-based backbone/neck, and default data augmentation and evaluation settings.

We use the same data augmentation strategies as Ultralytics YOLOv8. Following common practice, we progressively disable the strongest augmentations in the final training epochs to stabilize convergence Aside from our proposed CSD-Head and RCQ-Loss, all other training hyperparameters remain identical to the YOLOv8 default to ensure a fair comparison.

We integrate our modules in a plug-and-play manner. The CSD-Head directly replaces the original YOLOv8-S detection head as a drop-in module, without altering the backbone or neck. The RCQ-Loss augments the loss function with a task-aligned quality and ranking consistency term, which re-weights the regression and classification (objectness) losses in line with our label assignment strategy. These additions are designed to be compatible with YOLOv8's original training pipeline and assignment mechanisms.

3.3. Compared Methods and Setup

For a comprehensive comparison, we consider five external methods and use their official COCO results at 640 resolution:

YOLOv9-S: An improved YOLO variant that proposes Programmable Gradient Information (PGI) and a new

GELAN architecture to enhance training effectiveness. YOLOv9-S focuses on retaining information through an auxiliary reversible branch, achieving higher AP with no added inference cost.

YOLOv10-S: A recent real-time detector that introduces consistent dual assignments for labels, enabling NMS-free training/inference. This model is optimized for a better accuracy–latency trade-off through an efficiency-driven redesign of YOLO components.

Table 1. Comparison with state-of-the-art on COCO val2017 (640×640)

Method(scale)	AP	Params(M)	FLOPs(G)	Latency/Throughput
YOLOv8-S(Baseline)	44.9	11.2	28.6	**12.0ms/img**
YOLOv9-S	46.8	7.1	26.4	2.49ms/img
YOLOv10-S	46.3	7.2	21.6	-
Ours:YOLOv8-S+CSD-Head+RCQ-Loss	**46.9**	**11.5**	**30**	1.25ms/img

3.4. Main Results on COCO

As shown in Table 1, the official YOLOv8-S baseline achieves 44.9 AP on COCO val2017. Our enhanced model (YOLOv8-S + CSD-Head + RCQ-Loss) yields an AP of 46.9, which is an approximate improvement of +2.0 points over the baseline. This result is on par with or better than recent YOLO variants. For instance, YOLOv10-S and YOLOv9-S report 46.3 and 46.8 AP, respectively.

Overall, our method attains accuracy in the same range as these state-of-the-art lightweight detectors while maintaining a similar model complexity as YOLOv8-S. We emphasize that AP is used as the primary metric for fairness, given that the reported latency numbers come from different hardware and software setups. The latency figures in Table 1 should be interpreted qualitatively rather than as exact comparisons. (In future work, we will evaluate all methods under the same conditions to report unified throughput in ms/img for a truly fair comparison.)

3.5. Ablation Studies on YOLOv8-S

We conduct ablation experiments on the YOLOv8-S baseline to isolate the contributions of the CSD-Head and RCQ-Loss. Starting from the baseline, we add each component individually and then together, using the same training and evaluation protocol. Table 2 presents the step-by-step results. The baseline AP is 44.9. Incorporating the CSD-Head alone raises the AP to about 45.8, and using RCQ-Loss alone yields around 45.4 AP. Finally, applying both CSD-Head and RCQ-Loss together achieves approximately 46.7 AP, indicating that the two components are complementary and their benefits largely add up. We anticipate a combined gain of roughly 1.8–2.0 AP over the baseline with both modules enabled.

For completeness, we list the model size and compute cost for each ablation: adding the CSD-Head introduces a slight increase in parameters and FLOPs (due to the more complex head), whereas the RCQ-Loss does not affect model size or inference cost. All other settings remain unchanged, ensuring the performance gains stem from the proposed modules. In future work, we will run each experiment multiple times and report the mean and standard deviation of AP to account for variance. We also plan to evaluate error patterns using the TIDE toolbox and assess confidence calibration with metrics like Expected Calibration Error (ECE). These analyses will provide deeper insight into where our improvements come from, but are left for a follow-up once all results are obtained.

Table 2. Ablation results on YOLOv8-S (val2017, 640×640)

Method(scale)	AP	Params(M)	FLOPs(G)
YOLOv8-S(Baseline)	44.9	11.2	28.6
+CSD-Head	45.8	11.5	30
+RCQ-Loss	45.6	-	-
+CSD-Head+RCQ-Loss	**46.7**	**11.5**	**30**

As shown in Table 2, the ablation results confirm that both components independently improve performance, and their combination yields the highest gain. In particular, CSD-Head (which introduces cross-scale feature selection in the detection head) provides a notable boost in AP, likely by better handling objects of different scales. RCQ-Loss (which emphasizes rank-consistent quality in regression and classification) also improves AP, suggesting it makes the model's confidence scores more aligned with localization quality. When combined, we observe the largest AP increase, demonstrating that CSD-Head and RCQ-Loss address different aspects of the detection task and can be used together for cumulative effect. We will further verify these findings with more extensive experiments and analyses in future work.

4. Conclusion

This paper presented YOLO-AlignRank, a unified approach to the long standing misalignment between classification confidence and localization quality in one-stage detectors. Architecturally, the Cross-Scale Sparse Deformable Head (CSD-Head) performs adjacent level deformable sampling and then fuses features via a lightweight bidirectional cross scale dynamic module (BCDM), strengthening multi scale representation with negligible latency overhead. On the objective side, the Rank-Consistent Quality Loss (RCQ-Loss) couples quality aware calibration with list-wise distribution alignment and pairwise ranking constraints so that scores become faithful, rank-consistent surrogates for IoU. The method is plug-and-play within modern YOLO training and aligns naturally with NMS-free pipelines.

On MS COCO val2017 at 640×640 with YOLOv8-S, YOLO-AlignRank lifts AP from 44.9 to 46.9. Ablations show complementary contributions—CSD-Head alone at ≈45.8 AP and RCQ-Loss alone at ≈45.4 AP—with the combined model

reaching ≈46.7–46.9 AP, while the end-to-end FLOPs increase remains below 5% and inference throughput is essentially unchanged.

By jointly aligning cross-scale features and enforcing intra-object ranking consistency, the detector produces more reliable single score outputs for both NMS-based and NMS-free decoding, simplifying real time deployment and improving robustness for small or deformed objects.

Future work will report unified throughput on a common hardware/software stack, conduct fine-grained error and calibration analyses, and extend the approach to larger backbones, transformer heads, and broader assignments or domains.

Acknowledgments

We thank colleagues at the College of Artificial Intelligence, Xinjiang Vocational University of Technology for insightful discussions and computational support. And by institutional research funding from Xinjiang Vocational University of Technology. We also acknowledge the maintainers of MS COCO and the open-source communities behind PyTorch and Ultralytics YOLOv8, whose tools and codebases facilitated our experiments. The views expressed are those of the authors and do not necessarily reflect those of the sponsors.

References

[1] K. He, R. Girshick, and P. Dollár, "DETR: End-to-End Object Detection with Transformers," Advances in Neural Information Processing Systems (NeurIPS), 2020.

[2] H. Zhang, H. Wang, F. Dayoub, and N. Sunderhauf, "VarifocalNet: An IoU-Aware Dense Object Detector," in Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 8514–8523.

[3] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "YOLOX: Exceeding YOLO Series in 2021," arXiv:2107.08430, 2021.

[4] C. Wang, Y. Song, H. Li, et al., "YOLOv10: Real-Time End-to-End Object Detection," arXiv:2405.14458, 2024.

[5] X. Zhu, H. Hu, S. Lin, and J. Dai, "Deformable ConvNets v2: More Deformable, Better Results," arXiv:1811.11168, 2018.

[6] X. Zhu, W. Su, L. Lu, et al., "Deformable DETR: Deformable Transformers for End-to-End Object Detection," Proc. Int. Conf. on Learning Representations (ICLR), 2021.

[7] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path Aggregation Network for Instance Segmentation," Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 8759–8768.

[8] R. Hogan, "What is YOLOv10? An Architecture Deep Dive," Roboflow Blog, Jun. 2024.

[9] Z. Zheng, P. Wang, W. Liu, et al., "Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression," Proc. AAAI Conf. on Artificial Intelligence, 2020, pp. 12993–13000.

[10] Z. Xu, C. Zhang, and Z. Li, "OASL: Orientation-Aware Adaptive Sampling Learning for Arbitrary-Oriented Object Detection," Expert Systems with Applications, vol. 238, p. 122242, 2024.

[11] E. Oksuz, C. Cam, E. Akbas, and F. Porikli, "Rank & Sort Loss for Object Detection and Instance Segmentation," in Proc. IEEE/CVF Int. Conf. on Computer Vision (ICCV), 2021, pp. 2980-2989.

[12] H. Xu, X. Zhao, and W. Yu, "Adaptive Dynamic Non-Monotonic Focal IoU Loss for Object Detection," IEEE Access, vol. 12, pp. 105679–105692, 2024.

[13] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection," arXiv:1612.03144, 2016.

[14] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path Aggregation Network for Instance Segmentation," arXiv:1803.01534, 2018.

[15] Code Huddle, "Improving Instance Segmentation Using Path Aggregation Network," Medium, Oct. 2019.

[16] B. Li, Y. Liu, W. Ouyang, et al., "Prime Sample Attention in Object Detection," arXiv:1904.04821, 2019.

[17] X. Zhu, H. Hu, S. Lin, and J. Dai, “Deformable ConvNets v2: More deformable, better results,” in Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition, 2019, pp. 9308–9316.

[18] B. Yang, G. Bender, Q. V. Le, and J. Ngiam, “CondConv: Conditionally parameterized convolutions for efficient inference,” in Advances in Neural Information Processing Systems (NeurIPS), vol. 32, 2019.