Long-Tailed Object Detection Pre-training: Dynamic Rebalancing Contrastive Learning with Dual Reconstruction

Chen-Long Duan[1], **Yong Li**[1], **Xiu-Shen Wei**[2]*, **Lin Zhao**[1]

[1]Nanjing University of Science and Technology
[2]School of Computer Science and Engineering, and Key Laboratory of New Generation Artificial Intelligence Technology and Its Interdisciplinary Applications, Southeast University

Abstract

Pre-training plays a vital role in various vision tasks, such as object recognition and detection. Commonly used pre-training methods, which typically rely on randomized approaches like uniform or Gaussian distributions to initialize model parameters, often fall short when confronted with long-tailed distributions, especially in detection tasks. This is largely due to extreme data imbalance and the issue of simplicity bias. In this paper, we introduce a novel pre-training framework for object detection, called Dynamic Rebalancing Contrastive Learning with Dual Reconstruction (2DRCL). Our method builds on a Holistic-Local Contrastive Learning mechanism, which aligns pre-training with object detection by capturing both global contextual semantics and detailed local patterns. To tackle the imbalance inherent in long-tailed data, we design a dynamic rebalancing strategy that adjusts the sampling of underrepresented instances throughout the pre-training process, ensuring better representation of tail classes. Moreover, Dual Reconstruction addresses simplicity bias by enforcing a reconstruction task aligned with the self-consistency principle, specifically benefiting underrepresented tail classes. Experiments on COCO and LVIS v1.0 datasets demonstrate the effectiveness of our method, particularly in improving the mAP/AP scores for tail classes.

1 Introduction

With the advancement of deep learning, computer vision has seen significant progress, particularly in the development of large-scale pre-training and fine-tuning optimization paradigms [55, 59, 63, 65]. Numerous pre-training methods capture domain-specific or task-relevant concepts, boosting downstream performance [7, 14, 15, 27, 30–32, 48, 52]. In the field of object detection, current methods typically leverage ImageNet [10] and COCO [35] for pre-training, allowing partial model components, such as the backbone, to achieve satisfactory pre-training. However, these pre-training paradigms leave some key detection components randomly initialized and tend to overlook the suboptimal performance issues caused by long-tailed distributions during pre-training process.

In the traditional supervised pre-training paradigm, models are constrained by the distribution of labeled data, making it difficult for them to perform well in long-tailed settings, for example, in

*Corresponding author. The first two authors contribute equally to this work. This work was supported by National Key R&D Program of China (2021YFA1001100), National Natural Science Foundation of China under Grant (62272231, 62172222), the Fundamental Research Funds for the Central Universities (4009002401), and the Big Data Computing Center of Southeast University.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

tasks of pipeline failure detection [1] and face recognition [3]. While self-supervised learning has demonstrated potential in enabling models to learn richer and more effective feature representations without relying on labeled data [12, 15, 22, 27, 52], significant challenges remain. An often-overlooked but crucial challenge in long-tailed object detection is simplicity bias [23, 43, 46, 53, 54], where deep neural networks tend to rely on simpler predictive patterns while overlooking complex features that are crucial for model generalization. This bias is especially problematic for tail classes, as their limited examples make them more likely to be ignored by models that prioritize simpler patterns. To address these challenges, this work aims not only to develop a pre-training strategy that aligns with the unique demands of object detection but also to ensure its effectiveness across both balanced and long-tailed data distributions.

Motivated by this, we propose a novel pre-training framework called Dynamic Rebalancing Contrastive Learning with Dual Reconstruction (2DRCL), specifically designed for long-tailed object detection pre-training. Our method incorporates Holistic-Local Contrastive Learning, which combines holistic and local feature learning to better align the pre-training process with the fine-tuning phase. To address the issues of long-tailed distributions during pre-training, 2DRCL integrates a dynamic rebalancing strategy that improves the accuracy of tail classes. Unlike traditional resampling methods, our dynamic rebalancing sampler considers instance-level imbalance, offering more precise control over class distribution and ensuring that tail classes are adequately represented. Additionally, by introducing Dual Reconstruction, our method effectively mitigates simplicity bias, enabling the model to capture both complex patterns and nuanced features that are essential for long-tailed object detection. This dual mechanism ensures that the model not only retains detailed visual information but also grasps deeper semantic relationships, which is particularly crucial for accurately recognizing and distinguishing tail classes with limited examples.

To evaluate the effectiveness of our method, we conduct extensive experiments on two benchmark datasets, i.e., COCO [35] and LVIS v1.0 [13]. Experiments on these datasets from both quantitative and qualitative perspectives validate the effectiveness of our proposed method.

2 Related Work

Pre-training for Object Detection. Pre-training is a critical step in object detection, often involving the use of large-scale datasets to learn transferable representations. Commonly, CNNs pre-trained on image classification datasets like ImageNet [10] are fine-tuned for object detection tasks. Self-supervised pre-training methods [4, 6, 7, 15] have gained traction in recent years. These methods do not require labeled data and aim to learn useful representations through contrastive learning. To bridge the gap between pre-training and fine-tuning, dense-level contrastive learning methods [8, 20, 28, 51, 52, 57] explored local feature similarities between views, enhancing target perception and feature learning. Recognizing the insufficiency of pre-training solely the backbone, SoCo [52] advocated pre-training additional modules like FPN to process intricate scene-level information. In object detection, methods like UP-DETR [9] and DETReg [2] pre-trained entire DETR-like detectors with region matching and feature reconstruction tasks, while AlignDet [27] froze a pre-trained backbone during detection pre-training, achieving satisfactory results with fewer epochs. Nonetheless, these approaches still struggled with effectively addressing long-tailed distribution challenges.

Long-tailed Object Detection. In the literature [59, 63], repeat factor sampling [13, 58] aims to balance the data distribution by sampling tail classes more frequently. In object detection and segmentation tasks, achieving sample balance solely through straightforward resampling strategies is challenging due to the complexity of the scenes. Special loss functions represent another technical direction for tackling the long-tailed problem. EQL [45] protected tail classes from being over-suppressed by ignoring negative gradients from head samples, while EQL v2 [44] balanced gradients from head and tail classes. Seesaw loss [47] rebalanced the positive and negative gradients of each class using two reweighting factors. ECM loss [24] provided a theoretical understanding of the long-tailed tracking detection problem and introduced a novel alternative objective that optimized the margin-based binary classification error. Beyond these loss functions, methods such as supervised contrastive learning [48, 66], decoupled training [11, 40] and expert-based classifier training [56, 62, 64] have also demonstrated effectiveness under long-tailed settings. While these methods often implicitly reshape decision boundaries to protect tail classes, their indirect nature may limit their effectiveness in more complex long-tailed scenarios.

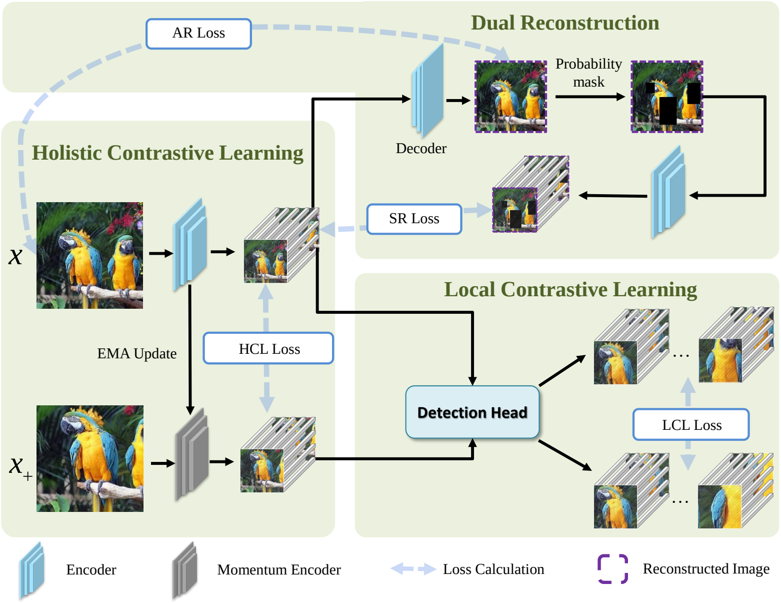


Figure 1: Illustration of the proposed Dynamic Rebalancing Contrastive Learning with Dual Reconstruction (2DRCL) method, which consists of the Holistic Contrastive Learning (Section 3.1.1), the Local Contrastive Learning (Section 3.1.2), and the Dual Reconstruction (Section 3.3). The whole network can be trained in an end-to-end manner.

3 Methodology

Our goal is to develop a pre-training approach tailored to the specific requirements of object detection, while maintaining robustness across both balanced and long-tailed data distributions. To this end, we introduce a novel method called Dynamic Rebalancing Contrastive Learning with Dual Reconstruction (2DRCL), specifically designed for pre-training in long-tailed object detection scenarios. In 2DRCL, we exploit a Holistic-Local Contrastive Learning (HLCL) paradigm to coordinate holistic and local feature learning to better align the pre-training with the fine-tuning phase. Building on this, a dynamic rebalancing strategy is incorporated, which emphasizes tail classes at both the image and instance (object proposal) levels to address data imbalance during pre-training. By integrating HLCL with this dynamic rebalancing strategy, we introduce a Dual Reconstruction component aimed at mitigating simplicity bias, enabling the model to concurrently capture both complex and subtle feature patterns essential for long-tailed object detection. Below, we present details of the three parts in 2DRCL.

3.1 Holistic-Local Contrastive Learning

In 2DRCL, the HLCL mechanism serves as the foundation for pre-training object detection models. The HLCL framework encompasses two key components: Holistic Contrastive Learning (HCL) and Local Contrastive Learning (LCL). HCL focuses on learning generic visual representations, enabling the backbone model to capture comprehensive image patterns and general semantic abstractions effectively. To integrate object-level representations into the pre-training process, LCL is introduced to guide both the backbone and the detection head toward object-level details within the image. By pre-training all network components used in object detectors, LCL ensures that the model is more precisely aligned with object detection tasks, while also enhancing its ability to capture fine-grained object-level features.

3.1.1 Holistic Contrastive Learning

We present HCL mechanism in Fig. 1. As illustrated, we follow the typical CL framework, i.e., MoCo [7, 15], to realize the holistic CL in our proposed 2DRCL framework. Typically, for an image $\mathcal{I}$, we apply different image views to obtain x and x_+ as inputs for the encoder and momentum encoder in HCL. Each view is randomly and independently augmented. Notice that the scale and location of the same object proposal are different across the augmented image views, which enables the model to learn translation-invariant and scale-invariant object-level representations in the following LCL part, which we will elaborate next.

Subsequently, x and x_+ are transformed via separate projectors, generating holistic-level representations, z and z_+, which are then ℓ_2-normalized. Subsequently, we employ the InfoNCE loss [15, 38] to drive the network training, formally:

$$\mathcal{L}_{HCL} = -\log \frac{\exp\left(z \cdot z_+/\tau\right)}{\exp\left(z \cdot z_+/\tau\right) + \sum_{i=1}^{K} \exp\left(z \cdot z_i/\tau\right)} \ , \tag{1}$$

where τ is a temperature hyper-parameter usually set as 0.2. For each input image, we use one positive and K negative samples for HCL, where K is fixed as 65,536. For the update of momentum encoder in Fig. 1, we use the same Exponential Moving Average (EMA) strategy as that in MoCo [7, 15]. Through HCL, the model is trained to effectively learn generic visual representations and capture comprehensive image patterns. However, solely relying on image-level pre-training may lead to an overemphasis on holistic representations, potentially neglecting features that are critical for object detection tasks.

3.1.2 Local Contrastive Learning

To introduce object-level representations into pre-training, we incorporate the LCL mechanism to bridge the gap between pre-training process and fine-tuning phase w.r.t object detection, as illustrated in Fig. 1. Specially, we employ a class-agnostic detector [26] to generate a series of proposals as bounding boxes $\mathcal{B} = \{b_1, b_2, \ldots, b_n\}$, where b_i denotes the i-th bounding box within the augmented input image x. The object-level representation of a proposal is then obtained via object detection heads (e.g., RoI [17]), denoted as z_{bb}. The LCL loss for the local-level representation can be formulated as:

$$\mathcal{L}_{LCL} = -\log \frac{\exp\left(z_{bb} \cdot z_{bb_+}/\tau\right)}{\exp\left(z_{bb} \cdot z_{bb_+}/\tau\right) + \sum_{i=1}^{K} \exp\left(z_{bb} \cdot z_{bb_i}/\tau\right)} \ . \tag{2}$$

where z_{bb_+} means a corresponding positive object proposal within another augmented input image x_+. The K negative proposals means any potential proposals within other unrelated images during training. To construct a dictionary comprising a large number of object proposals from different input images, we utilize a queue-based structure. Sequences from the current mini-batch are enqueued, while the oldest mini-batch sequences are dequeued, ensuring that the dictionary size is independent of the mini-batch size. LCL mechanism maximizes the similarity between object proposals across augmented views, enabling the model to learn comprehensive representations for diverse object proposals, thus enhancing its robustness in object detection tasks.

Finally, the objective w.r.t the HLCL mechanism can be formulated as:

$$\mathcal{L}_{HLCL} = \alpha_c \mathcal{L}_{HCL} + \beta_c \mathcal{L}_{LCL} \ , \tag{3}$$

where α_c and β_c are the weights of HCL and LCL loss, respectively.

3.2 Dynamic Rebalancing

To precisely control class distribution and ensure adequate representation of tail classes, we propose a dynamic resampling method that considers both images and object proposals. Unlike traditional resampling strategies, such as Repeat Factor Sampling (RFS) [13], which primarily emphasize class-balanced sampling, our approach aims to prioritize tail classes more effectively through resampling at both the image level and the object-proposal level. Given that object detection requires the identification and localization of specific objects, addressing instance-level imbalance in addition to image-level imbalance is expected to achieve a more balanced representation, particularly benefiting tail classes.

The proposed resampling method incorporates a dynamic adjustment mechanism, enabling the model to initially learn the overall distribution of the dataset and progressively shift its focus towards tail classes as pre-training advances. Specifically, for each category c, we calculate image-level and instance-level scores, denoted as f_c^{im} and f_c^{in}, respectively. Here, f_c^{im} indicates the proportion of images belonging to the c-th category in the entire dataset, while f_c^{in} represents the proportion of object proposals associated with the c-th category across the dataset. These two scores reflect the imbalance ratio for category c, following the approach used in RFS [13]. The combined score for each category, f_c, is then defined as the harmonic mean of these two scores:

$$f_c = \frac{f_c^{im} \cdot f_c^{in}}{\alpha_d f_c^{im} + (1 - \alpha_d)\, f_c^{in}} \,. \tag{4}$$

where the hyper-parameter α_d changes dynamically throughout pre-training, defined as $\alpha_d = \frac{T}{T_{max}}$, where T is the current epoch and T_{max} is the total number of pre-training epochs. As pre-training progresses, the value of α_d increases, gradually shifting the focus from image-level balancing to instance-level balancing, enabling the model to increasingly emphasize tail classes.

To achieve balanced sampling, we define the category-level repeat factor r_c based on the score f_c using the formula $r_c = \max\left(1, \sqrt{t/f_c}\right)$, where t is a fixed hyper-parameter set at 0.001. This repeat factor ensures that categories with lower scores (typically tail classes) are sampled more frequently during training. The dynamic resampling strategy effectively addresses data imbalance at both the image and instance levels, enhancing focus on tail classes while mitigating the risk of overfitting due to excessive repetition of rare instances.

3.3 Dual Reconstruction

Building on the HLCL and dynamic resampling mechanisms, which provide the foundation for pre-training object detection and mitigate instance imbalance, respectively, our proposed 2DRCL framework introduces a Dual Reconstruction component to address simplicity bias. This component enables the model to concurrently capture both complex and subtle feature patterns, which are vital for effective long-tailed object detection.

3.3.1 Simplicity Bias

Simplicity bias [23, 43, 46] is a phenomenon where models tend to favor simpler predictive patterns, often neglecting complex features that are critical for effective generalization. This issue is particularly prevalent in long-tailed distributions, where it significantly affects the performance on tail classes. In such scenarios, models struggle to learn the intricate and unique characteristics of these tail classes, further exacerbating the class imbalance problem.

To bridge this gap, we propose a Dual Reconstruction (DRC) component aimed at mitigating simplicity bias by enhancing feature discrimination for both head and tail classes. As shown in Fig. 1, DRC comprises two key elements: Appearance Reconstruction (AR) and Semantic Reconstruction (SR). The AR component enforces pixel-level reconstruction, compelling the model to capture as many subtle details as possible for each input image. In contrast, the SR component ensures semantic consistency between the features of the original input image and those of a corresponding randomly occluded image. We hypothesize that the effective implementation of DRC will enable the model to retain fine-grained visual information while also capturing deeper semantic relationships. This capability is particularly important for accurately recognizing and distinguishing tail classes, which often have limited training examples. This strategy ensures accurate visual representation while promoting a deeper semantic focus, enabling the model to better handle tail classes in long-tailed object detection.

3.3.2 Appearance Reconstruction

To enforce appearance consistency, we utilize an auto-encoding structure specifically designed for high-fidelity reconstruction of input images. The encoder f, parameterized by θ, maps an input image x into a dense feature space, represented as $z = f\left(x\right)$. A generator g, parameterized by η, then attempts to invert this mapping, producing a reconstructed version: $\hat{x} = g\left(f\left(x\right)\right)$. Through pixel-wise image reconstruction, the Appearance Reconstruction (AR) component compels the latent features, $f\left(x\right)$, to capture as many subtle details as possible for each input image.

AR is not merely replicating the input image; rather, it acts as an auxiliary regularization mechanism that focuses on distilling discriminative visual features relevant to the primary object detection task for each input image. By enforcing image reconstruction, AR enables the model to effectively capture both prominent and nuanced details present in the input data. The AR loss is formulated as a pixel-wise mean-squared error (MSE), expressed as:

$$\mathcal{L}_{AR} = \|x - g(f(x))\|_2^2 \ . \tag{5}$$

3.3.3 Semantic Reconstruction

While AR ensures that the model captures fine-grained visual details essential for accurately representing and distinguishing between different objects, especially in cases with limited examples of tail classes, it is equally important to maintain semantic integrity in the reconstructed images. This semantic consistency allows the model to focus on the underlying meaning and context of the image, rather than merely surface-level details, thereby promoting a more robust and generalized understanding of each input.

To address this need, we introduce Semantic Reconstruction (SR), which incorporates controlled perturbations during the reconstruction process. SR is designed to preserve the semantic content of the original image while allowing the model to learn to recognize and reconstruct meaningful features even when certain parts of the image are altered or obscured. This approach ensures that the model develops a deeper understanding of each input's inherent structure and context.

Specifically, we apply a mask to a fixed percentage (e.g., 25%) of an object proposal within the reconstructed image $g(f(x))$, resulting in a masked version, denoted as $\mathcal{M}(g(f(x)))$, where $\mathcal{M}$ means image masking operation. This masked image is then re-encoded by the encoder to generate the corresponding latent features, $\hat{z} = f(\mathcal{M}(g(f(x))))$. The Semantic Reconstruction (SR) loss is computed by measuring the congruence between the feature representations of the vanilla images and those of the masked reconstructed images, evaluated across multiple layers of the network. This approach ensures that the model maintains semantic consistency while learning to recognize and reconstruct meaningful features.

$$\mathcal{L}_{SR} = \sum_{p=1}^{P} \|f(x) - f(\mathcal{M}(g(f(x))))\|_2^2 \ , \tag{6}$$

where P represents the number of feature layers considered, and the SR loss, $\mathcal{L}_{SR}$, is defined as the Euclidean distance between the original (vanilla) features and the reconstructed features across these layers. The SR component ensures that even in the presence of visual disruptions, the essential semantic features are preserved, allowing the model to learn robust, invariant features that go beyond superficial visual similarities. This approach enhances the model's ability to generalize by focusing on meaningful semantic information rather than just appearance.

Conclusively, our proposed DRC leverages both appearance and semantic consistency to address simplicity bias, encouraging the model to learn rich and complex feature representations essential for accurate and robust detection of tail classes. The interplay between the two reconstruction losses enhances the model's sensitivity to both fundamental visual details and higher-level semantic features, leading to a more versatile and effective detection paradigm. This combined approach ensures that the model not only captures detailed visual information but also grasps abstract semantic relationships, improving its overall performance in long-tailed object detection tasks.

The total loss for the Dual Reconstruction combines the AR and SR losses can be formulated as:

$$\mathcal{L}_{DRC} = \alpha_r \mathcal{L}_{AR} + (1 - \alpha_r)\mathcal{L}_{SR} \ , \tag{7}$$

where α_r balances the trade-off between visual fidelity and semantic accuracy. This dual-focus strategy force the model to reconstruct the image/features for both the head and the tail classes. This, DRC enhances the model's ability to represent and detect tail classes effectively.

Overall, the final loss function of our method is optimized by:

$$\mathcal{L} = \mathcal{L}_{HLCL} + \mathcal{L}_{DRC} + \mathcal{L}_{det} \ , \tag{8}$$

where $\mathcal{L}_{det}$ denotes the loss of object detection that makes the pre-training consistent with the task. For simplicity, the weights of all losses in $\mathcal{L}$ are set to 1.

4 Experiments

In this section, we outline the experimental settings, implementation details, and main results. Additionally, a comprehensive description of the experimental settings is provided in Section A.1 of the Appendix.

Table 1: Comparisons with state-of-the-art methods on COCO (Mask R-CNN with R50-FPN).

Backbone Initialization	Methods	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
From scratch	DenseCL [51]	39.6	59.3	43.3	-	-	-
	Self-EMD [36]	40.4	61.1	43.7	37.4	56.5	**39.7**
	SoCo [52]	40.6	61.1	44.4	-	-	-
	SlotCon [57]	41.0	61.1	45.0	-	-	-
ImageNet pre-trained backbone	Surpervised	38.3	58.0	42.1	34.3	54.9	36.6
	AlignDet [27]	39.4	59.2	43.2	35.3	56.1	37.7
	Ours	**41.4**	**61.3**	**45.8**	**37.4**	**57.2**	39.4

4.1 Experimental Configurations

Datasets. We conduct experiments on two representative datasets: COCO [35] and LVIS v1.0 [13]. The COCO dataset is a standard benchmark for object detection, segmentation, and captioning tasks, comprising 80 classes with a relatively balanced distribution, including 118k training images and 5k validation images. Given the balanced nature of the class distribution in COCO, we use this dataset to evaluate the performance of the proposed 2DRCL under balanced settings. In addition, we utilize the LVIS v1.0 dataset to benchmark long-tailed object detection scenarios. LVIS features 1,203 classes with a highly imbalanced distribution, containing 100k training images and 19.8k validation images. The classes in LVIS are categorized into three groups based on their frequency of occurrence [13]: rare (1~10 instances), common (11~100 instances), and frequent (>100 instances). This categorization allows for a comprehensive assessment of 2DRCL's performance under long-tailed data distributions.

Implementation Details. Experiments are conducted with both Faster R-CNN and Mask R-CNN frameworks. For a comprehensive comparison, we use both ResNet-50 and ResNet-101 backbones. All models are implemented using the MMDetection toolbox [5]. We pre-train the models on 8 RTX3090 GPUs with a batch size of 16. Unless otherwise specified, pre-training follows the 1× schedule (12 epochs), starting with an initial learning rate of 0.02, which is reduced by a factor of 10 after the 8th and 11th epochs. For 2× schedule, models are trained with 24 epochs, and the learning rate decays at the end of epoch 16 and 22. In our experiments, the hyper-parameters are set as follows: α_c is set to 0.1, β_c is set to 0.05, α_r is set to 0.1. When conducting experimental comparisons on the LVIS v1.0 dataset in Table 3, we first use our 2DRCL for pre-training, followed by the application of existing long-tailed methods for fine-tuning to further enhance performance. Finally, we select 'ECM [24]+2DRCL' as 'Ours' for comparison with state-of-the-art methods.

4.2 Quantitative Results

Table 2: Comparisons with pre-trained methods on LVIS v1.0 with a 1× scheduler using Mask R-CNN.

Method	AP^{bb}	AP^{bb}_{r}	AP^{bb}_{c}	AP^{bb}_{f}
MoCo v2 [7]	14.5	3.9	12.4	21.6
SimCLR [6]	19.9	8.0	18.1	27.1
BYOL [12]	15.3	5.4	13.2	21.9
SoCo [52]	17.6	5.3	15.9	24.9
AlignDet [27]	22.6	10.3	20.8	29.9
Ours	**23.9**	**11.9**	**22.3**	**31.0**

Mask R-CNN with R50-FPN on COCO dataset. Table 1 presents the comparison, where all methods are pre-trained on the COCO training dataset and evaluated on the COCO validation dataset. Typically, the backbone can be initialized either from scratch or using an ImageNet pre-trained model. Methods such as DenseCL [51], Self-EMD [36], and SoCo [52], which are initialized from scratch, achieve an AP^{bb} ranging from 39.6% to 41.0%. Notably, these methods rely on 200~800 epochs for training. As a comparison, methods that utilize an ImageNet pre-trained style, including AlignDet and our proposed 2DRCL, require only 12 epochs for pre-training. Among the compared methods in Table 1, our 2DRCL demonstrates superior object detection performance, achieving the highest AP^{bb} of 41.4% and AP^{mk} of 37.3%, significantly outperforming both AlignDet and the supervised baseline. This improvement can be attributed to 2DRCL's capability to narrow the gap between pre-training and fine-tuning. By effectively bridging this gap, 2DRCL is able to leverage the benefits of the pre-trained model more efficiently for object detection tasks.

Table 3: Comparisons with state-of-the-art methods on LVIS v1.0 with a 2× schedule.

(a) Faster R-CNN with R50-FPN.

Method	AP^{bb}	AP_r^{bb}	AP_c^{bb}	AP_f^{bb}
BCE [42]	19.5	1.6	16.6	30.6
RFS [13]	24.2	14.2	22.3	30.6
DropLoss [21]	21.8	5.2	21.8	29.1
PCB [19]	23.0	6.2	21.5	32.2
EQLv2 [44]	25.4	15.8	23.5	31.7
Seesaw [47]	26.4	16.8	25.1	32.2
BAGS [29]	23.7	14.2	22.2	29.6
ACSL [50]	22.2	9.9	21.3	28.5
LOCE [11]	25.1	15.7	24.2	30.1
BACL [40]	26.1	16.0	25.7	30.9
ECM [24]	26.7	17.5	25.7	32.2
Ours	**27.3**	**18.6**	**25.8**	**32.6**

(b) Mask R-CNN with ResNet-50/101.

Backbone	Method	AP	AP_r	AP_c	AP_f	AP^{bb}
R50-FPN	CE	18.7	0.4	16.5	29.3	19.7
	RFS [13]	23.7	14.2	22.9	29.3	24.7
	EQLv2 [44]	25.2	17.4	24.1	29.9	26.0
	LOCE [11]	26.6	18.5	26.2	30.7	27.4
	SeeSaw [47]	26.9	19.6	26.8	30.5	27.3
	ECM [24]	27.4	19.7	27.0	31.1	27.9
	Ours	**27.7**	**20.4**	**27.1**	**31.4**	**28.3**
R101-FPN	CE	25.5	16.6	24.5	30.6	26.6
	EQLv2 [44]	27.2	20.6	25.9	31.4	27.9
	SeeSaw [47]	28.2	20.3	28.1	31.8	29.0
	ECM [24]	28.7	**21.9**	28.4	32.2	29.4
	Ours	**28.8**	21.1	**28.7**	**32.3**	**29.6**

Comparisons with Pre-trained Methods on LVIS v1.0. In Table 2, we present a comparison of our method with several state-of-the-art pre-trained methods on the LVIS v1.0 dataset using the Mask R-CNN framework with a 1× scheduler. The results obviously illustrate the limitations of existing pre-training approaches in addressing the challenges posed by long-tailed distributions w.r.t the object detection tasks. Specifically, traditional pre-training methods consistently demonstrate inferior performance on tail classes, as evidenced by their relatively low AP_r^{bb} scores. For instance, MoCo v2 [7] and BYOL [12] achieve AP_r^{bb} scores of 3.9% and 5.3%, respectively, indicating a obvious deficiency in precisely detect the target objects w.r.t the long-tailed classes. Our proposed 2DRCL is specially designed for long-tailed object detection pre-training and shows consistent superiority in Table 2. By dynamically rebalancing the data distribution and incorporating Dual Reconstruction mechanisms, 2DRCL effectively captures object-level characteristics for both head and tail classes. The superior performance of the proposed 2DRCL highlights its efficacy in addressing long-tailed object detection challenges, demonstrating a strong capability to focus on tail classes and alleviate the inherent imbalance issues in such datasets.

Comparisons with State-of-the-art Long-tailed Object Detection Methods on LVIS v1.0. To evaluate the effectiveness of our method for long-tailed object detection, we compare 2DRCL with state-of-the-art techniques across different object detection frameworks (Faster R-CNN and Mask R-CNN) and backbone networks (ResNet-50 and ResNet-101) on the LVIS v1.0 dataset. As shown in Table 3, our method achieves the highest accuracy in both AP^{bb} and AP. Specifically, for the Faster R-CNN framework, our pre-training technique outperforms all the competitors, particularly in AP^{bb} and AP_r^{bb}. This advantage is consistently observed with the Mask R-CNN framework as well. We attribute this improved performance, particularly for tail classes, to 2DRCL's dynamic rebalancing of the data distribution and the introduction of Dual Reconstruction mechanisms. The effectiveness of 2DRCL stems from its ability to significantly mitigate extreme imbalance and simplicity bias for tail classes during the pre-training phase. We will investigate the contribution for each of components in 2DRCL in Section 4.2.

Discussions. To address concerns that the performance gains might be attributed to longer training durations, we evaluate the COCO dataset using various fine-tuning schedules, with the results presented in Table 4a. The findings demonstrate that even with 1× fine-tuning (12 epochs), our method surpasses the baseline trained with 4× fine-tuning, indicating that the observed performance improvements of 2DRCL are not merely due to extended training epochs. Furthermore, Table 4b shows results on the LVIS v1.0 dataset, where all methods are compared under the same total number of training epochs for a fair evaluation. Specifically, while long-tailed methods are fine-tuned for 12 epochs, our 2DRCL employs 6 epochs pre-training followed by 6 epochs fine-tuning, maintaining an equivalent overall training duration. The results reveal an average improvement of 0.5% in AP^{bb} and 1.3% in AP_r^{bb} with 2DRCL's pre-training strategy, underscoring the effectiveness of 2DRCL in enhancing long-tailed object detection performance.

Table 4: Comparisons w.r.t different training/fine-tuning epochs.

(a) Comparisons under different fine-tuning epochs on COCO. The preceding four methods exploit ImageNet pre-trained backbone.

Fine-tuning Schedule	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}
1×	38.3	58.0	42.1
2×	38.8	58.4	42.4
3×	39.0	58.7	42.9
4×	39.2	59.5	42.9
1× (Ours)	41.4	61.3	45.8

(b) Results on LVIS v1.0 with same training epochs.

Methods	AP^{bb}	AP^{bb}_{r}	AP^{bb}_{c}	AP^{bb}_{f}
RFS	22.7	9.1	21.5	30.0
+Ours	**23.3**	**10.3**	**21.7**	**30.3**
EQL	24.9	14.8	24.1	**30.4**
+Ours	**25.2**	**15.9**	**24.3**	30.3
Seesaw	24.7	14.7	23.6	30.4
+Ours	**25.2**	**15.6**	**24.2**	**30.5**
ECM	26.5	17.0	25.4	31.7
+Ours	**27.0**	**19.0**	**25.8**	**31.9**

Ablation Analysis across 2DRCL Components. To investigate the contribution for each component in 2DRCL, we evaluate our 2DRCL on the LVIS v1.0 dataset. As shown in Table 5, incorporating HLCL, which combines HCL and LCL, results in a 1.7% improvement in AP^{bb}_{r} over the baseline. HCL focuses on learning generic visual representations, enabling the backbone to capture comprehensive image patterns and semantic abstractions, while LCL ensures precise alignment with object detection tasks and enhances the capture of fine-grained object-level features. Additionally, DRB dynamically rebalances the data distribution and helps to significantly boosts performance for tail classes, leading to a 3.5% improvement in APr^{bb}. The Dual Reconstruction (DRC) mechanism, comprising AR and SR, brings the total improvement to 6.1%. AR enforces pixel-level reconstruction, compelling the model to capture subtle visual details, while SR ensures semantic consistency between the original and occluded images. This combination allows the model to retain intricate visual information while capturing deeper semantic relationships, resulting in enriched and coherent feature representations.

Table 5: Ablations for various components in our 2DRCL on LVIS v1.0.

HCL	LCL	DRB	AR	SR	AP^{bb}	AP^{bb}_{r}	AP^{bb}_{c}	AP^{bb}_{f}
✗	✗	✗	✗	✗	22.7	9.1	21.5	30.0
✓	✗	✗	✗	✗	22.5	10.5	21.0	29.3
✗	✓	✗	✗	✗	21.9	9.8	20.8	28.7
✓	✓	✗	✗	✗	22.4	10.8	21.1	29.0
✓	✓	✓	✗	✗	23.8	14.3	22.3	30.1
✓	✓	✓	✓	✗	24.2	14.9	22.6	30.3
✓	✓	✓	✓	✓	**24.4**	**15.2**	**22.7**	**30.3**

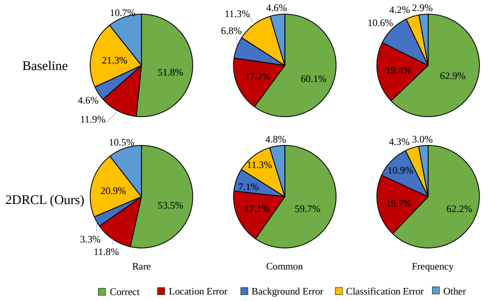


Figure 2: Error analyses comparisons. 2DRCL achieves superior performance on tail classes without significantly compromising accuracy for the more frequent classes.

4.3 Further Analysis

In this section, we conduct a thorough analysis of our proposed 2DRCL, emphasizing its role in mitigating simplicity bias and enhancing feature representation through DRC mechanism.

Error Analyses. To determine which error types our 2DRCL method effectively mitigates, we conducted an error analysis experiment. Following the error categorization paradigm from YOLO [41], we classify the top N predictions for each class into five error types. The pie charts in Figure 2 show the distribution of these errors for rare, common, and frequent classes on the LVIS v1.0 validation set. As shown in Figure 2, our 2DRCL shows noticeable improvements for rare object classes, with correct predictions increasing from 51.8% in the baseline to 53.5%, alongside a reduction in both non-background classification errors and background prediction errors. This suggests that our 2DRCL enhances the model's ability to accurately classify the rare objects and accurately distinguish them from the background. Although there is a slight accuracy decrease for common and frequent classes, this trade-off is minimal, with the gains in rare class detection outweighing these minor losses. This demonstrates that our method effectively addresses long-tailed object detection challenges by improving performance on tail classes without obviously compromising accuracy for other frequent classes across the dataset.

Simplicity Bias Analyses. To explicitly illustrate how our method addresses simplicity bias, we present a visualization of the activations corresponding to randomly sampled test images from the LVIS v1.0 dataset in Figure 3. The results demonstrate that 2DRCL effectively mitigates simplicity bias in long-tailed object detection by learning more comprehensive patterns that encompass informative regions, particularly for images belonging to tail classes. In comparison, 2DRCL consistently identifies more critical regions than ECM [24], highlighting the superiority of our approach in addressing simplicity bias. The comparisons presented in the fourth and fifth rows underscore the effectiveness of the proposed DRC mechanism, revealing that the introduction of the DRC mechanism significantly enhances feature attention on tail classes while reducing background interference. This finding further indicates that the DRC plays a crucial role in mitigating simplicity bias, enabling the model to retain intricate visual details and capture deeper semantic relationships, thereby producing enriched and coherent feature representations.

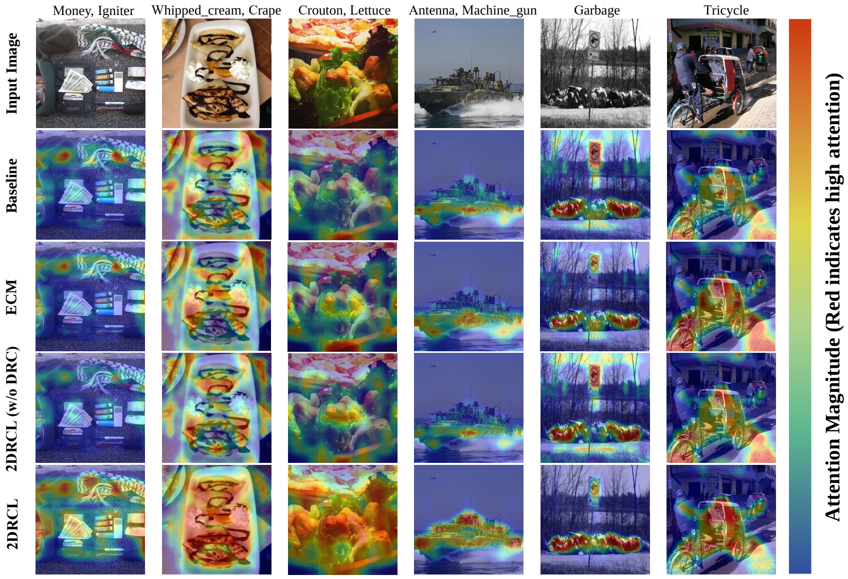


Figure 3: Attention map comparisons w.r.t Baseline [13], ECM [24], 2DRCL (w/o DRC) and 2DRCL (our method) on LVIS dataset. The top row shows the corresponding class names of the input images. Best viewed in color.

5 Conclusions and Limitations

We proposed Dynamic Rebalancing Contrastive Learning with Dual Reconstruction (2DRCL) to address the challenges posed by long-tailed distributions in object detection pre-training. By integrating holistic and local contrastive learning with dynamic rebalancing and dual reconstruction, 2DRCL aligned the pre-training strategy with the specific demands of object detection, ensuring effectiveness across both balanced and long-tailed data. It successfully mitigated simplicity bias for tail classes, enhancing their feature representations and overall performance. Experiments demonstrated significant improvements in attention to tail classes and reduced background errors, as confirmed by both quantitative and qualitative analyses. However, our method had limitations, particularly in its relatively high computational costs. Future work will focus on optimizing computational efficiency.

References

[1] Mohammad H. Alobaidi, Mohamed A. Meguid, and Tarek Zayed. Semi-supervised learning framework for oil and gas pipeline failure detection. *Scientific Reports*, 12(13758), 2022.

[2] Amir Bar, Xin Wang, Vadim Kantorov, Colorado J Reed, Roei Herzig, Gal Chechik, Anna Rohrbach, Trevor Darrell, and Amir Globerson. DETReg: Unsupervised pretraining with region priors for object detection. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 14585–14595, 2022.

[3] Dong Cao, Xiangyu Zhu, Xingyu Huang, Jianzhu Guo, and Zhen Lei. Domain balancing: Face recognition on long-tailed domains. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 5670–5678, 2020.

[4] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 9912–9924, 2020.

[5] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, Zheng Zhang, Dazhi Cheng, Chenchen Zhu, Tianheng Cheng, Qijie Zhao, Buyu Li, Xin Lu, Rui Zhu, Yue Wu, Jifeng Dai, Jingdong Wang, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. MMDetection: Open MMLab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019.

[6] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big self-supervised models are strong semi-supervised learners. *arXiv preprint arXiv:2006.10029*, 2020.

[7] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020.

[8] Zeren Chen, Gengshi Huang, Wei Li, Jianing Teng, Kun Wang, Jing Shao, Chen Change Loy, and Lu Sheng. Siamese DETR. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 15722–15731, 2023.

[9] Zhigang Dai, Bolun Cai, Yugeng Lin, and Junying Chen. UP-DETR: Unsupervised pre-training for object detection with transformers. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 1601–1610, 2021.

[10] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 248–255, 2009.

[11] Chengjian Feng, Yujie Zhong, and Weilin Huang. Exploring classification equilibrium in long-tailed object detection. In *Proc. IEEE Int. Conf. Comput. Vis.*, pages 3417–3426, 2021.

[12] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Remi Munos, and Michal Valko. Bootstrap your own latent - a new approach to self-supervised learning. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 21271–21284, 2020.

[13] Agrim Gupta, Piotr Dollar, and Ross Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 5356–5364, 2019.

[14] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 15979–15988, 2022.

[15] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 9729–9738, 2020.

[16] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *Proc. IEEE Int. Conf. Comput. Vis.*, pages 2961–2969, 2017.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 37(9):1904–1916, 2015.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 770–778, 2016.

[19] Yin-Yin He, Peizhen Zhang, Xiu-Shen Wei, Xiangyu Zhang, and Jian Sun. Relieving long-tailed instance segmentation via pairwise class balance. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 7000–7009, 2022.

[20] Olivier J Hénaff, Skanda Koppula, Jean-Baptiste Alayrac, Aaron van den Oord, Oriol Vinyals, and João Carreira. Efficient visual pretraining with contrastive detection. In *Proc. IEEE Int. Conf. Comput. Vis.*, pages 10086–10096, 2021.

[21] Ting-I Hsieh, Esther Robb, Hwann-Tzong Chen, and Jia-Bin Huang. DropLoss for long-tail instance segmentation. In *Proc. AAAI Conf. Artif. Intell.*, pages 1549–1557, 2021.

[22] Feiran Hu, Chenlin Zhang, Jiangliang Guo, Xiu-Shen Wei, Lin Zhao, Anqi Xu, and Lingyan Gao. An asymmetric augmented self-supervised learning method for unsupervised fine-grained image hashing. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 17648–17657, 2024.

[23] Minyoung Huh, Hossein Mobahi, Richard Zhang, Brian Cheung, Pulkit Agrawal, and Phillip Isola. The low-rank simplicity bias in deep networks. *arXiv preprint arXiv:2103.10427*, 2021.

[24] Jang Hyun Cho and Philipp Krähenbühl. Long-tail detection with effective class-margins. In *Proc. Eur. Conf. Comput. Vis.*, pages 698–714, 2022.

[25] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proc. Int. Conf. Mach. Learn.*, pages 448–456, 2015.

[26] Dahun Kim, Tsung-Yi Lin, Anelia Angelova, In So Kweon, and Weicheng Kuo. Learning open-world object proposals without learning to classify. *IEEE Trans. Robot. Autom.*, 7(2):5453–5460, 2022.

[27] Ming Li, Jie Wu, Xionghui Wang, Chen Chen, Jie Qin, Xuefeng Xiao, Rui Wang, Min Zheng, and Xin Pan. AlignDet: Aligning pre-training and fine-tuning in object detection. In *Proc. IEEE Int. Conf. Comput. Vis.*,

pages 6866–6876, 2023.

[28] Yong Li and Shiguang Shan. Contrastive learning of person-independent representations for facial action unit detection. *IEEE Trans. Image Process.*, 32:3212–3225, 2023.

[29] Yu Li, Tao Wang, Bingyi Kang, Sheng Tang, Chunfeng Wang, Jintao Li, and Jiashi Feng. Overcoming classifier imbalance for long-tail object detection with balanced group softmax. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 10988–10997, 2020.

[30] Yong Li, Jiabei Zeng, and Shiguang Shan. Learning representations for facial actions from unlabeled videos. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(1):302–317, 2022.

[31] Yong Li, Jiabei Zeng, Shiguang Shan, and Xilin Chen. Occlusion aware facial expression recognition using cnn with attention mechanism. *IEEE Trans. Image Process.*, 28(5):2439–2450, 2019.

[32] Yong Li, Jiabei Zeng, Shiguang Shan, and Xilin Chen. Self-supervised representation learning from videos for facial action unit detection. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 10916–10925, 2019.

[33] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 2117–2125, 2017.

[34] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proc. IEEE Int. Conf. Comput. Vis.*, pages 2980–2988, 2017.

[35] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *Proc. Eur. Conf. Comput. Vis.*, pages 740–755, 2014.

[36] Songtao Liu, Zeming Li, and Jian Sun. Self-EMD: Self-supervised object detection without imagenet. *arXiv preprint arXiv:2011.13677*, 2020.

[37] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proc. Int. Conf. Mach. Learn.*, pages 807–814, 2010.

[38] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[39] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 8026–8037, 2019.

[40] Tianhao Qi, Hongtao Xie, Pandeng Li, Jiannan Ge, and Yongdong Zhang. Balanced classification: A unified framework for long-tailed object detection. *IEEE Trans. Multimedia*, 26:3088–3101, 2024.

[41] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 779–788, 2016.

[42] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(6):1137–1149, 2016.

[43] Harshay Shah, Kaustav Tamuly, Aditi Raghunathan, Prateek Jain, and Praneeth Netrapalli. The pitfalls of simplicity bias in neural networks. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 9573–9585, 2020.

[44] Jingru Tan, Xin Lu, Gang Zhang, Changqing Yin, and Quanquan Li. Equalization loss v2: A new gradient balance approach for long-tailed object detection. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 1685–1694, 2021.

[45] Jingru Tan, Changbao Wang, Buyu Li, Quanquan Li, Wanli Ouyang, Changqing Yin, and Junjie Yan. Equalization loss for long-tailed object recognition. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 11662–11671, 2020.

[46] Damien Teney, Ehsan Abbasnejad, Simon Lucey, and Anton Van den Hengel. Evading the simplicity bias: Training a diverse set of models discovers solutions with superior ood generalization. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 16761–16772, 2022.

[47] Jiaqi Wang, Wenwei Zhang, Yuhang Zang, Yuhang Cao, Jiangmiao Pang, Tao Gong, Kai Chen, Ziwei Liu, Chen Change Loy, and Dahua Lin. Seesaw loss for long-tailed instance segmentation. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 9695–9704, 2021.

[48] Peng Wang, Kai Han, Xiu-Shen Wei, Lei Zhang, and Lei Wang. Contrastive learning based hybrid networks for long-tailed image classification. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 943–952, 2021.

[49] Tao Wang, Yu Li, Bingyi Kang, Junnan Li, Junhao Liew, Sheng Tang, Steven Hoi, and Jiashi Feng. The devil is in classification: A simple framework for long-tail instance segmentation. In *Proc. Eur. Conf. Comput. Vis.*, pages 728–744, 2020.

[50] Tong Wang, Yousong Zhu, Chaoyang Zhao, Wei Zeng, Jinqiao Wang, and Ming Tang. Adaptive class suppression loss for long-tail object detection. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 3103–3112, 2021.

[51] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Dense contrastive learning for self-supervised visual pre-training. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 3024–3033, 2021.

[52] Fangyun Wei, Yue Gao, Zhirong Wu, Han Hu, and Stephen Lin. Aligning pretraining for detection via object-level contrastive learning. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 22682–22694, 2021.

[53] Xiu-Shen Wei, Yang Shen, Xuhao Sun, Peng Wang, and Yuxin Peng. Attribute-aware deep hashing with self-consistency for large-scale fine-grained image retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(11):13904–13920, 2023.

[54] Xiu-Shen Wei, Xuhao Sun, Yang Shen, Anqi Xu, Peng Wang, and Faen Zhang. Delving deep into simplicity bias for long-tailed image recognition. *arXiv preprint arXiv:2302.03264*, 2023.

[55] Xiu-Shen Wei, He-Yang Xu, Zhiwen Yang, Chen-Long Duan, and Yuxin Peng. Negatives make a positive: An embarrassingly simple approach to semi-supervised few-shot learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 46(4):2091–2103, 2024.

[56] Xiu-Shen Wei, Shu-Lin Xu, Hao Chen, Liang Xiao, and Yuxin Peng. Prototype-based classifier learning for long-tailed visual recognition. *Sci. China Inf. Sci.*, 65(6):160105, 2022.

[57] Xin Wen, Bingchen Zhao, Anlin Zheng, Xiangyu Zhang, and Xiaojuan Qi. Self-supervised visual representation learning with semantic grouping. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 16423–16438, 2022.

[58] Burhaneddin Yaman, Tanvir Mahmud, and Chun-Hao Liu. Instance-aware repeat factor sampling for long-tailed object detection. *arXiv preprint arXiv:2305.08069*, 2023.

[59] Lu Yang, He Jiang, Qing Song, and Jun Guo. A survey on long-tailed visual recognition. *Int. J. Comput. Vis.*, 130(7):1837–1872, 2022.

[60] Shaoyu Zhang, Chen Chen, and Silong Peng. Reconciling object-level and global-level objectives for long-tail detection. In *Proc. IEEE Int. Conf. Comput. Vis.*, pages 18982–18992, 2023.

[61] Shifeng Zhang, Cheng Chi, Yongqiang Yao, Zhen Lei, and Stan Z. Li. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 9759–9768, 2020.

[62] Yifan Zhang, Bryan Hooi, Lanqing Hong, and Jiashi Feng. Self-supervised aggregation of diverse experts for test-agnostic long-tailed recognition. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 34077–34090, 2022.

[63] Yifan Zhang, Bingyi Kang, Bryan Hooi, Shuicheng Yan, and Jiashi Feng. Deep long-tailed learning: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(9):10795–10816, 2023.

[64] Boyan Zhou, Quan Cui, Xiu-Shen Wei, and Zhao-Min Chen. BBN: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 9719–9728, 2020.

[65] Beier Zhu, Kaihua Tang, Qianru Sun, and Hanwang Zhang. Generalized logit adjustment: Calibrating fine-tuned models by removing label bias in foundation models. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 64663–64680, 2023.

[66] Jianggang Zhu, Zheng Wang, Jingjing Chen, Yi-Ping Phoebe Chen, and Yu-Gang Jiang. Balanced contrastive learning for long-tailed visual recognition. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 6898–6907, 2022.

A Appendix / supplemental material

In the supplementary materials, we present further information about the proposed 2DRCL pre-training framework, including: 1) More detailed experimental settings, including the specifics of pre-training and downstream fine-tuning, as well as the setup for error analysis; 2) Additional experimental results for further analysis.

A.1 Implementation Details

Pre-training Settings. First, we generate a series of high-quality bounding boxes using a class-agnostic detector [26]. Then, we randomly select 8 bounding boxes from this set for subsequent pre-training. Through the introduction of object proposals, the architectural discrepancy is reduced between pre-training and downstream detection fine-tuning. Faster R-CNN [42] and Mask R-CNN [16] are commonly adopted frameworks to evaluate transfer performance. We employ MMDetection [5] as our detection framework to conduct our experiment. Both the projection network and prediction network are 2-layer MLPs which consist of a linear layer with output size 256 followed by batch normalization [25], rectified linear units (ReLU) [37], and a final linear layer with output dimension 256. Once all views are constructed, we employ the data augmentation pipeline of MoCo [7, 15]. Our generator architecture consists of four deconvolutional layers with dimensions (2048, 512), (512, 256), (256, 64), and (64, 3), respectively, and each layer uses a kernel size of 4. ReLU is used for non-linear activation between the layers. Specifically, we apply random horizontal flip, random crop, color distortion, Gaussian blur, and the solarization operation. The models are trained with a total batch size of 16 on 8 GPUs (RTX3090 with 24 GB VRAM). Unless otherwise specified, all pre-training follows the default $1\times$ (12 epochs) schedule. In each stage, the learning rate starts at 0.02 and decreases by 0.1 after 8 and 11 epochs, respectively. If not specified, the supervised pre-trained ResNet [18] in PyTorch [39] is used by default for both the pre-training and fine-tuning stages.

Training Details. We reproduce multiple methods with different paradigms as our baselines, including end-to-end and decoupled methods, such as RFS [13], SeeSaw [47], ECM [24], ROG [60], LOCE [11] and BACL [40], following their default experiment settings. In terms of the model architecture, we opt for the popular ResNet [18] with FPN [33] as the backbone and train detection models of Faster-RCNN and Mask-RCNN for $1\times$ or $2\times$ scheduler. We trained the models using SGD with 0.9 momentum. The batch size and learning rate are set as 16 and 0.02, and the data augmentation strictly follows previous long-tailed detection methods [24, 47]. For $1\times$ schedule with 12 training epochs, the learning rate is initialized as 0.02, and then decays by 0.1 at the end of epoch 8 and 11. For $2\times$ schedule, models are trained with 24 epochs, and the learning rate decays at the end of epoch 16 and 22. We evaluated our models using both COCO and LVIS metrics. For COCO, we report object detection metrics including average precision for bounding boxes (AP^{bb}), AP with an IoU threshold of 50% (AP^{bb}_{50}), and AP with an IoU threshold of 75% (AP^{bb}_{75}). For instance segmentation, we report AP^{mk} (AP for masks), AP^{mk}_{50}, and AP^{mk}_{75}. The LVIS evaluation includes mean average precision (mAP), AP at an IoU of 50% (AP_{50}), AP at an IoU of 75% (AP_{75}), as well as AP for rare (AP_r), common (AP_c), and frequent classes (AP_f). For Mask R-CNN, we report AP for instance segmentation and AP^{bb} for object detection.

The Setting of Error Analyses. Following the settings of [41], we choose the top N predictions for each category during inference time. Each prediction is classified based on the type of error:

- Correct: correct class and IOU > 0.5
- Location Error: correct class and 0.1 < IOU < 0.5
- Background Error: IOU < 0.1 for any object
- Classification Error: class is wrong and IOU > 0.5
- Other: class is wrong and 0.1 < IOU < 0.5

A.2 Additional Experiment

Consistent Improvements. We evaluate the effectiveness of our method on the LVIS v1.0 dataset by combing the proposed 2DRCL method with existing long-tailed object detection methods. As

Table A.1: Experiments on **LVIS v1.0**. We combine eight existing methods with our method '2DRCL'. The ResNet-50-FPN and ResNet-101-FPN are adopted as backbones for Mask R-CNN. We reproduced all methods using their official code and trained with a 1× schedule, totaling 12 epochs.

Strategy	Schedules	Methods	+Ours	LVIS v1.0 (ResNet-50-FPN)				LVIS v1.0 (ResNet-101-FPN)			
				AP^{bb}	AP_r^{bb}	AP_c^{bb}	AP_f^{bb}	AP^{bb}	AP_r^{bb}	AP_c^{bb}	AP_f^{bb}
End-to-end	12 epochs	RFS [13]	*no*	22.7	9.1	21.5	30.0	24.8	12.1	23.4	31.9
			yes	**23.9**	**11.9**	**22.3**	**31.0**	**25.1**	**12.7**	**23.5**	**32.4**
		IRFS [58]	*no*	24.4	14.3	22.6	30.8	26.3	16.5	24.5	32.5
			yes	**24.7**	**14.3**	**22.9**	**31.3**	**26.5**	**16.7**	**24.6**	**32.8**
		EQLv2 [44]	*no*	24.9	14.8	24.1	30.4	26.3	17.7	24.4	31.2
			yes	**25.7**	**16.5**	**24.5**	**31.0**	**26.9**	**18.9**	**25.1**	**32.5**
		SeeSaw [47]	*no*	24.7	14.7	23.6	30.4	26.3	15.1	25.4	32.2
			yes	**26.2**	**17.5**	**25.0**	**31.5**	**27.0**	**17.6**	**25.6**	**32.6**
		ECM [24]	*no*	26.5	17.0	25.4	31.7	27.9	19.2	**26.5**	33.5
			yes	**27.3**	**19.2**	**25.9**	**32.5**	**28.0**	**19.5**	26.2	**33.7**
		ROG [60]	*no*	25.7	16.4	24.4	31.2	27.3	18.5	26.2	32.5
			yes	**26.2**	**16.9**	**24.8**	**31.8**	**27.6**	**18.8**	**26.3**	**32.9**
Decoupled	24+6 epochs	LOCE [11]	*no*	27.2	18.7	25.7	32.6	28.5	19.0	**27.0**	34.3
			yes	**27.6**	**18.9**	**26.5**	**33.0**	**28.7**	**20.2**	26.8	**34.4**
	12+12 epochs	BACL [40]	*no*	26.1	16.0	25.7	30.9	27.2	16.7	26.8	32.3
			yes	**27.0**	**17.5**	**25.9**	**32.5**	**28.4**	**18.9**	**27.3**	**33.7**

Table A.2: One-stage object detection results on LVIS v1.0 validation set. We compare different methods with ResNet-50 backbone on 2× schedule using ATSS.

Methods	AP^{bb}	AP_r^{bb}	AP_c^{bb}	AP_f^{bb}
Focal Loss [34]	25.6	14.5	24.3	**31.8**
ECM [24]	26.1	16.6	25.2	31.3
Ours	**26.4**	**17.7**	**25.4**	30.8

shown in Table A.1, using 2DRCL leads to consistent AP^{bb} improvement over existing classification-based methods, surpassing all of them with large margins. Interestingly, combining our methods can be observed further growth in multiple paradigms. The method 'ECM+2DRCL' (which trained with a 1× schedule) can almost achieve the same rare object detection accuracy as the LOCE [11] method, and surpasses BACL [40] for about 1.0% AP_r^{bb}. Therefore, we speculate that by using 2DRCL during training, the model can generate more balanced feature representations, allowing it to achieve comparable results to the decoupled method with minimal training when combined with end-to-end approaches.

Comparison on ATSS Framework. Table A.2 presents the performance comparison of our method against the baseline Focal Loss [34] and ECM Loss [24] on the ATSS [61] detection framework. Our method achieves the highest overall average precision (AP^{bb}) at 26.4%, outperforming both Focal Loss and ECM Loss. Notably, for rare classes, our method significantly improves performance with an AP_r^{bb} of 17.7%, compared to 14.5% for Focal Loss and 16.6% for ECM Loss. Our method also shows consistent improvement for common classes, surpassing both Focal Loss and ECM Loss. Although Focal Loss achieves the highest precision for frequent classes, our method maintains competitive performance across all categories.

Results on COCO-LT. To further verify the generalization ability of our 2DRCL, we construct a long-tailed distribution dataset COCO-LT by sampling images and annotations from COCO [35] train 2017 split. Following [49], we divide 80 classes into 4 groups with < 20, 20-400, 400-8000, and >= 8000 training instances and report the accuracy for each group as AP_1, AP_2, AP_3, AP_4. In Table A.3, we compare our 2DRCL method with the baseline model and several state-of-the-art long-tailed detection methods on the COCO-LT dataset. The results demonstrate that 2DRCL consistently

Table A.3: Results on COCO-LT dataset. All experiments were conducted using the Mask R-CNN framework with a ResNet-50-FPN backbone and a 1× training schedule.

Methods	AP	AP_1	AP_2	AP_3	AP_4
CE	18.7	0.0	8.2	24.4	26.0
BAGS [29]	21.5	13.4	17.7	22.5	26.0
EQLv2 [44]	23.1	3.8	17.4	25.8	29.4
Seesaw [47]	22.7	3.4	15.5	**26.2**	28.5
ECM [24]	22.9	11.0	18.7	25.7	28.7
Ours	**24.4**	**14.4**	**20.2**	26.1	**29.4**

Table A.4: Efficiency evaluation of Mask R-CNN with ResNet50-FPN.

Methods	VRAM	Training Time	AP^{bb}	AP_r^{bb}
ECM (12 epochs)	**94 GB**	**16.1 h**	26.5	17.0
+AlignDet (6+6 epochs)	103 GB	19.5 h	26.2	16.7
+Ours (6+6 epochs)	106 GB	20.6 h	**27.0**	**19.0**

outperforms the baseline model by a notable 5.7% in overall AP. Notably, 2DRCL achieves the highest AP across both group 1 and group 2, outperforming the closest competitor, ECM, by 3.4% and 1.5% AP, respectively. We attribute these improvements to 2DRCL's ability to mitigate the simplicity bias toward tail classes during pre-training, which contrasts with methods such as Seesaw and ECM that primarily address the issue of unequal competition between foreground classes without sufficiently addressing the inherent bias in representation learning. By directly confronting these biases, our method demonstrates substantial gains in tail category performance while maintaining strong results across the full distribution.

Efficiency Evaluations. Table A.4 compares the VRAM usage, training time, and performance of different methods for long-tailed object detection. Our method, which includes 6 epochs of pre-training followed by 6 epochs of fine-tuning, utilizes slightly more VRAM and training time compared to AlignDet and ECM. Despite the increased computational cost, our method delivers a notable performance boost, achieving the highest AP^{bb} of 27.0% and AP_r^{bb} of 19.0%, demonstrating a clear performance-cost trade-off. These results indicate that while our pre-training strategy demands more resources, it does not negatively impact fine-tuning performance and offers significant improvements in long-tailed detection.

Qualitative results. We present additional visualization results on the LVIS dataset [13]. For simplicity, we use RFS [13] with Cross-Entropy (CE) loss as the baseline and combine it with our 2DRCL method. As shown in Figure A.1, both the baseline and our 2DRCL detect most objects in an image, but 2DRCL generally captures more details. For instance, 2DRCL helps discover missed boxes, such as the horse carriage in the center-left images. Additionally, 2DRCL generates more accurate bounding box predictions, like correctly identifying boats on the ground. While using a basic method still results in numerous classification errors, this issue can be mitigated by carefully designing the loss function.

Bias Analyses. We visualize how 2DRCL mitigates the weight norm bias induced by long-tailed distributions. Figure A.2a shows the classifier weight norm distribution across classes for models trained with the LVIS v1.0 training split. The transparent lines represent actual weight norms, while the solid lines show polynomial fits, providing a smoothed interpretation of trends. Our 2DRCL method achieves weight norm performance comparable to ECM [24], indicating effective bias mitigation. Figures A.2b and A.2c further illustrate that each component of 2DRCL contributes to reducing weight norm bias. Despite these gains, some advantages achieved at the feature level are slightly diminished during downstream fine-tuning, limiting overall progress in final outcomes.

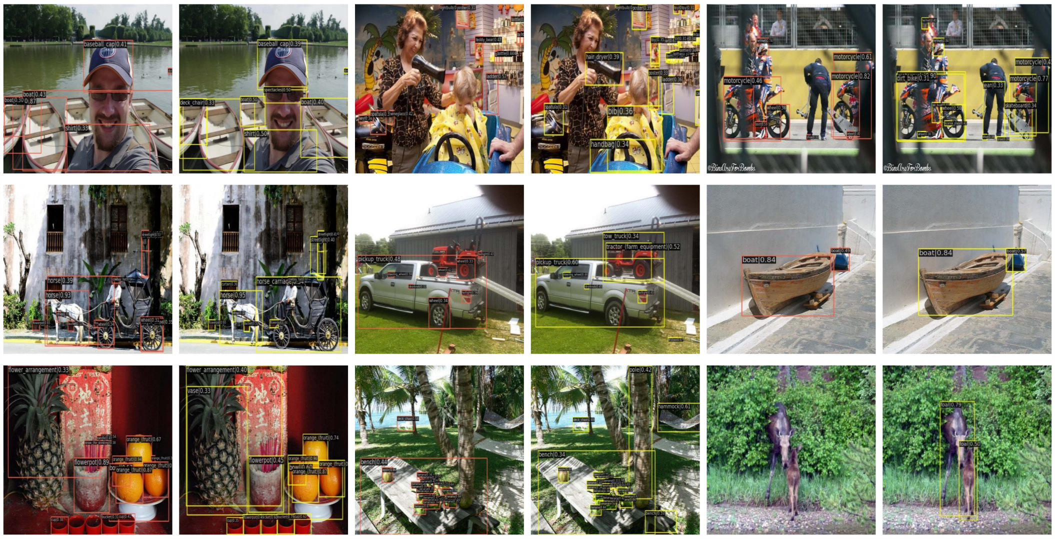
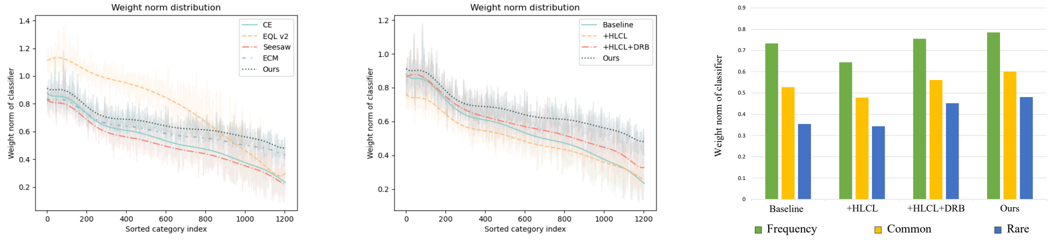


Figure A.1: Visualizations of detection results before (in the left of each group) and after (in the right) using our 2DRCL. We adopted RFS [13] as the baseline in LVIS and combined it with our 2DRCL pre-training method. In comparison, the proposed method is good at detecting missing objects and rectifying bounding box predictions. This figure needs to be viewed in color.



(a) (b) (c)

Figure A.2: (a) and (b) are classifiers' weight norm distribution across different classes in Mask R-CNN models trained with the LVIS v1.0 training split [13]. The X-axis represents the sorted category index based on category frequency. The Y-axis shows the weight norm. Transparent lines depict the actual weight norms for each category, providing a raw look at the data distribution. The solid lines represent polynomial curves fitted to the transparent data, offering a smoothed interpretation of trends across classes. (a) represents the comparison with the state-of-the-art methods, while (b) represents the comparison with the proposed components in this paper. (c) represents the average weight norm of the classifiers for each frequency category.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims presented in the abstract and introduction accurately reflect the contributions and scope of the paper.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses the limitations of the work performed by the authors.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: In our paper, we attribute the bias induced by the long-tailed distribution to biases in classifier weights and feature representations (such as simplicity bias). We provide relevant illustrations in the paper to support this claim.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Due to page limitations, we have included more detailed experimental information in the appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
    (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
    (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
    (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
    (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: We will continue to conduct further research based on this work. However, we can consider releasing the main checkpoints for public use.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: Detailed specifics are provided in the appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: We have demonstrated the effectiveness of 2DRCL on the LVIS v1.0 dataset, in particular significantly improving performance in the detection of rare classes. Although we demonstrate the performance improvement in graphs and experiments, details of error bars or statistical significance tests are not explicitly mentioned in Sections 4.2 and 4.3.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper specifies the use of 8 RTX3090 GPUs with 24 GB VRAM each, and provides details on the training schedule and learning rates, ensuring reproducibility of the experiments .

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in the paper adheres to the NeurIPS Code of Ethics. All experimental procedures, data handling, and reporting practices were conducted with full compliance to ethical standards, ensuring transparency, integrity, and respect for all stakeholders involved.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [No]

Justification: The proposed Dynamic Rebalancing Contrastive Learning with Dual Reconstruction (2DRCL) method significantly improves object detection accuracy, especially for rare objects. In autonomous systems, such as self-driving cars and drones, this improvement can enhance the detection of critical but infrequent objects, leading to safer navigation and operation.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: The paper does not release any data or models that pose a high risk for misuse. The focus of the paper is on the development and evaluation of the 2DRCL method for object detection, which does not involve the release of pre-trained models, language models, image generators, or scraped datasets with high misuse potential.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The paper provides appropriate credits to the creators or original owners of the assets used, including references to code packages, datasets, and models. However, it does not explicitly mention the licenses and terms of use for these assets. Including this information would ensure proper respect for intellectual property and compliance with usage terms .

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [No]

    Justification: The paper does not introduce new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing experiments or research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing experiments or research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.