Fractal Calibration for long-tailed object detection

Konstantinos Panagiotis Alexandridis[1,*], Ismail Elezi[1], Jiankang Deng[2], Anh Nguyen[3] and Shan Luo[4]
[1]Huawei Noah's Ark Lab; [2]Imperial College London; [3]University of Liverpool; [4]King's College London
*konstantinos.alexandridis@huawei.com

Abstract

*Real-world datasets follow an imbalanced distribution, which poses significant challenges in rare-category object detection. Recent studies tackle this problem by developing re-weighting and re-sampling methods, that utilise the class frequencies of the dataset. However, these techniques focus solely on the frequency statistics and ignore the distribution of the classes in image space, missing important information. In contrast to them, we propose **FRA**ctal **CAL**ibration (FRACAL): a novel post-calibration method for long-tailed object detection. FRACAL devises a logit adjustment method that utilises the fractal dimension to estimate how uniformly classes are distributed in image space. During inference, it uses the fractal dimension to inversely downweight the probabilities of uniformly spaced class predictions achieving balance in two axes: between frequent and rare categories, and between uniformly spaced and sparsely spaced classes. FRACAL is a post-processing method and it does not require any training, also it can be combined with many off-the-shelf models such as one-stage sigmoid detectors and two-stage instance segmentation models. FRACAL boosts the rare class performance by up to* 8.6% *and surpasses all previous methods on LVIS dataset, while showing good generalisation to other datasets such as COCO, V3Det and OpenImages. We provide the code at https://github.com/kostas1515/FRACAL.*

1. Introduction

In recent years, there have been astonishing developments in the field of object detection [8, 10, 57]. Most of these works utilise vast, balanced, curated datasets such as ImageNet1k [15], or MS-COCO [48] to learn efficient image representations. However, in the real world, data are rarely balanced, in fact, they follow a long-tailed distribution [55]. When models are trained with long-tailed data, they perform well for the frequent classes of the distribution but they perform inadequately for the rare classes [46, 68, 78]. This problem poses significant challenges to the safe deployment of detection and instance segmentation models in real-world safe-critical applications such as autonomous vehicles, medical applications, and industrial applications, scenarios where rare class detection is paramount.

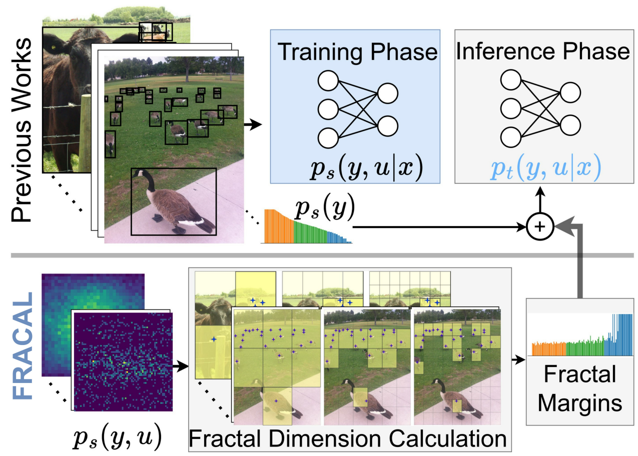

Figure 1. Previous works used class information $p_s(y)$ to align the learned source distribution $p_s(y,u|x)$ with the balanced target distribution $p_t(y,u|x)$, without considering the space u and class y relationship i.e. $p_s(y,u)$. FRACAL captures $p_s(y,u)$, using the fractal dimension, and embeds fractal margins during inference, aligning the learned distribution $p_s(y,u|x)$ with the target $p_t(y,u|x)$ better than previous works.

Many approaches address the long-tailed detection problem by employing adaptive re-weighting or data resampling techniques to handle imbalanced distributions [76, 79, 90]. However all these methods require training. In contrast, in long-tailed image classification, alternative methods focus on mitigating class imbalance during inference through a post-calibrated softmax adjustment (PCSA) [3, 26, 68]. PCSA boasts strong performance, good compatibility with many methods like data augmentation, masked image modeling, contrastive learning, and does not necessitate specialized loss function optimization, making it more user friendly [11, 86, 102].

However, current PCSA methods utilise solely the train set's class frequency $p_s(y)$ as shown in Fig.1-top, overlooking the significance of the classes' dependence on the location distribution $p_s(y,u)$. This is a significant limitation of previous PCSA methods because the location information is a critical indicator considering the correlation between classes y and their respective locations u.

Motivated by the class-location dependence [34], in this work, we investigate a novel way to incorporate location information into post-calibration for imbalanced object detection to boost the performance of rare classes by fully exploiting dataset statistics. We empirically show that naively injecting location statistics results in inferior performance because the location information is sparse for the rare classes. To overcome this, we propose FRACAL (FRActal CALibration), a novel post-calibration method based on the fractal dimension, as shown in Fig.1-bottom. Our method aggregates the location distribution of all objects in the training set, using the box-counting method [70]. This resolves the sparsity problem and significantly enhances the performance of both frequent and rare classes as shown in our experiments.

Our method comes with several advantages. First, it performs an effective class calibration, suitable for the object detection task, using the dataset's class frequencies. Secondly, it captures the class-location dependency [34], using the fractal dimension, and it fuses this information into class calibration. This results in a better and unique space-aware logit-adjustment technique that complements the frequency-dependent class calibration method and achieves higher overall performance compared to previous PCSA techniques.

FRACAL can be easily combined with both one-stage and two stage detectors, Softmax and Sigmoid-based models, various instance segmentation architectures, various backbones, sampling strategies, and largely increase the performance during the inference step. FRACAL significantly advances the performance on the challenging LVISv1 benchmark [20] with no training, or additional inference cost by 8.6% rare mask average precision (AP_r^m).

Our **contributions** are as follows:

- For the first time, we show the importance of the class-location dependence in post-calibration for long-tailed object detection.
- We capture the location-class dependence via a space-aware long-tailed object detection calibration method based on the fractal dimension.
- Our method performs remarkably on various detectors and backbones, on both heavily imbalanced datasets such as LVIS and less imbalanced datasets such as COCO, V3DET and OpenImages, outperforming the state-of-the-art by up to 8.6%.

2. Related Work

General Object Detection. General object detection [8, 10, 47, 50, 52, 67, 69, 71, 104] and instance segmentation [5, 6, 9, 22, 31, 47, 75] have witnessed tremendous advancements. Recently, transformer-based detectors were proposed which use self-attention to directly learn object proposals [8, 104], or diffusion-based methods which use a de-noising process to learn bounding boxes [10] and segmentation masks [19]. However, all of these methods struggle to learn the rare classes when trained with long-tailed data [20, 62] due to the insufficient rare samples. To this end, FRACAL enhances the rare class performance using a space-aware logit adjustment that can be easily applied during inference.

Table 1. Post-calibration techniques in long-tailed tasks. τ and γ are hyper-parameters, bg is the background class, μ_y and ς_y are estimated class mean and standard deviation respectively. Compared to past works, FRACAL uses both frequency (**F**) and space (**S**) information, as shown in Section 3.

Method	Use	Adjustment
LA. [60]	**F**	$z'_y = z_y - \tau\log(p_s(y))$
IIF [3]	**F**	$z'_y = -z_y \cdot \log(p_s(y))$
PCSA [26]	**F**	$z'_y = z_y - \log(p_s(y)) + \log(p_t(y))$
Norcal [63]	**F**	$p'_y = \frac{p_y/n_y^\gamma}{p_{bg}+\sum p_y/n_y^\gamma}, y \notin bg$
LogN [97]	**F**	$z'_y = \frac{z_y-(\mu_y-\min_y(\mu_y))}{\varsigma_y}, y \notin bg$
FRACAL	**S+F**	$z'_y = \mathrm{S}(\mathrm{C}(z_y))/\sum_{j=1}^{C+1}\mathrm{S}(\mathrm{C}(z_y))$

Long-tailed image classification. In the past years, the long-tailed image recognition problem has received great attention, as demonstrated by many recent surveys [62, 87, 96] and newly created benchmarks [18, 74, 88]. In long-tailed classification, the works could be split into two groups, representation learning and classifier learning. Representation learning techniques aim to efficiently learn rare class features using oversampling [27, 66, 90], contrastive learning [13, 44, 102], using ensemble or fusion models [1, 12, 39, 41, 82], knowledge distillation [24, 41, 43], knowledge transfer [55, 65, 103], sharpness aware minimisation [59, 100, 101] and neural collapse [42, 53, 99]. Classifier learning techniques aim to adjust the classifier in favour of the rare classes via decoupled training [29, 33, 94], margin adjustment [3, 7, 26, 32, 60, 68, 89, 98] and cost-sensitive learning [14, 35, 83]. Among these works, the Post-Calibrated Softmax Adjustment (PCSA) method [26, 58, 60] distinguishes itself through both its strong performance and the absence of any training requirements. However, most of the classifier and representation learning techniques are hard to adopt in long-tailed object detection. This difficulty arises from the larger imbalance inherent in this task, amplified by the presence of the background class [61, 87]. Moreover, the optimisation of models for this task becomes more complex due to multiple sources of imbalance such as batch imbalance, class imbalance and task imbalance as outlined in this survey [62]. For this reason, we develop FRACAL, which is a post-calibration method tailored to the long-tailed object detection task. Different from post-calibration classification methods [26, 60], FRACAL enhances the detection performance by leveraging class-

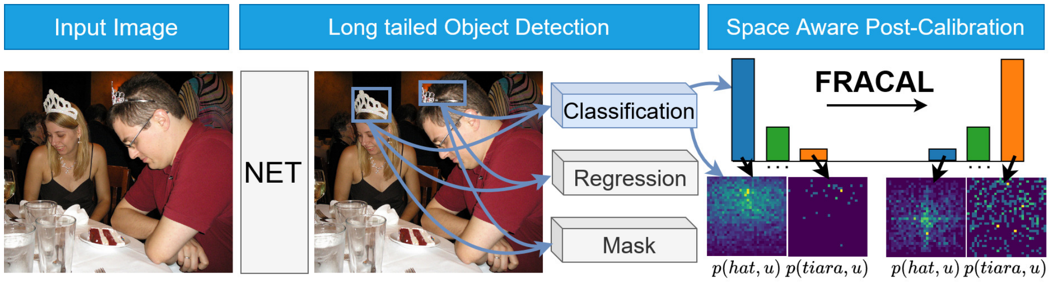

Figure 2. During imbalanced object detection, the model makes more frequent class detections like *hat* and less rare class detections like *tiara* both of which have strong upper location bias. FRACAL utilises fractal dimension and debiases the logits both in the frequency and space axes, making fewer *hat* detections and more *tiara* detections that are both evenly spread in image space.

dependent space information derived from the fractal dimension. Through space-aware logit-adjustment, FRACAL mitigates biases in both the detection's location and classification axes.

Long-tailed object detection. The most prevalent technique is adaptive rare class re-weighting, which could be applied using either the statistics of the mini-batch [28, 72, 79] or the statistics of the gradient [40, 73]. Other works use adaptive classification margins based on the classifier's weight norms [38, 80], classification score [17, 25, 76], activation functions [2, 4], group hierarchies [46, 85] and ranking loss [95]. Many works use data resampling techniques [17, 20, 33, 85, 90] or external rare class augmentation [91, 92]. All these works optimise the model on the long-tailed distribution and require the construction of a complicated and cumbersome training pipeline. In contrast, our method operates during the model's inference stage thus it is easier to use and less evasive to the user's codebase.

Norcal [63] was the first method to apply a post-calibration technique in imbalanced object detection, achieving promising results without training the detector. They proposed to calibrate only the foreground logits using the train-set's label statistics and applied a re-normalisation step. LogN [97] proposed to use the model's own predictions to estimate the class statistics and applied standardisation in the classification layer. However LogN, requires forward-passing the whole training set through the model to estimate the weights, thus it is slower than FRACAL, which is not model-dependent. Also, both methods do not utilise the spatial statistics of the classes which are valuable indicators since the classes and their location are correlated [34]. To this end, FRACAL balances the detectors using both class and space information, largely surpassing the performance of the previous methods. FRACAL can be easily combined with two-stage softmax-based models like MaskRCNN [22], or one-stage sigmoid detectors such as GFLv2 [45] achieving great results without training or additional inference cost.

Relation to previous works. In Table 1, we contrast our work to previous post-calibration methods used in classification and object detection. As the Table suggests, all prior methods use only frequency information and none of them considers the space information.

3. Methodology

We show the overview of our approach in Fig.2. FRACAL is essentially a post-processing method that calibrates the classification logits of the detector using precomputed weights based on the class and space statistics of the train-set. The FRACAL weights can be stored in the memory, thus during inference our method has insignificant overhead. Its effects on the detector are twofold, on the frequency axis, it decreases frequent class detections like *hat* and increases rare class detections like *tiara*. On the space axis, it produces more uniformly spaced detections for all classes, by forcing e.g. both *hats* and *tiaras* to appear in all locations and not just the top. Next, we analyse our method in detail.

3.1. Background: Classification Calibration

Let $f_y(x;\theta) = z$ be a classifier parameterised by θ, x the input image, y the class, z the logit, $\bar{y}$ is the model's prediction and $p_s(y)$ and $p_t(y)$ the class priors on the train and test distributions respectively. The post-calibration equation is:

$$\bar{y} = \arg\max_y (f_y(x;\theta) + \log(p_t(y)) - \log(p_s(y))). \quad (1)$$

This has been numerously analysed in previous literature [3, 26, 51, 60, 68] and we derive it in Appendix. In short, this shows that to get better performance, one can align the model's predictions with the test distribution, by subtracting $\log(p_s(y))$ and adding $\log(p_t(y))$ in the logit space. We now extend it to object detection.

3.2. Classification Calibration for Object Detection

In classification, $p(y)$ can be easily defined using the dataset's statistics, by using instance frequency n_y, i.e. $p(y) = \frac{n_y}{\sum_j^C n_j}$. In object detection, this is not the case because $p(y)$ is affected by the location and the object class.

Following [2], we define the class priors as:

$$p(y, o, u) = p(y|o, u) \cdot p(o, u) = p(y, u) \cdot p(o, u), \quad (2)$$

where o is an object, irrespective of class, and u is the location inside the image. Substituting Eq.2 in Eq.1, $\bar{y}$ becomes:

$$\bar{y} = \arg\max_y (f_y(x; \theta) + \log(\frac{p_t(y, u) \cdot p_t(o, u)}{p_s(y, u) \cdot p_s(o, u)}). \quad (3)$$

The term $p(o, u)$ in Eq.3 cannot be calculated apriori as it depends on the model's training (e.g., the IoU sampling algorithm, how the object class is encoded etc[1]). Despite this, $p_s(o, u) \approx p_t(o, u)$, as we show in the Appendix, which means that the object distributions of the train and the test set remain the same and only the foreground class distribution changes. As a result:

$$\bar{y} = \arg\max_y (f_y(x; \theta) + \log(p_t(y, u)) - \log(p_s(y, u))). \quad (4)$$

Next, we show how the location parameter u affects Eq. 4.

3.2.1. Location-class independence.

We consider the case where the location u does not give any information. In this case, u and y are independent variables, thus $p(y, u) = p(y) \cdot p(u)$ and we rewrite Eq. 4 as:

$$\begin{aligned} \bar{y} &= \arg\max_y (f_y(x; \theta) + \log(\frac{p_t(y) \cdot p_t(u)}{p_s(y) \cdot p_s(u)}) \\ &= \arg\max_y (f_y(x; \theta) + \log(p_t(y)) - \log(p_s(y))), \end{aligned} \quad (5)$$

where $p(u)$ is the probability of a random location in the image space and it has been simplified because it is the same in both source and target distributions, i.e., $p_s(u) = p_t(u)$.

In theory $p_t(y)$ is unknown, thus Eq.5 cannot be applied. Despite that, we found that setting $p_t(y) = \frac{1}{C}$ works well, because it forces the model to do balanced detections on the test set. In practice, this maximises average precision because this metric independently evaluates all classes and it rewards balanced detectors [16]. Accordingly, the Classification (C) calibration of the logit z_y is:

$$\mathrm{C}(z_y) = \begin{cases} z_y - \log_\beta(\frac{n_y}{\sum_i^C n_i}) + \log_\beta(\frac{1}{C}), & y \in \{1, ..., C\} \\ z_y, & y = \mathrm{bg}, \end{cases} \quad (6)$$

where β is the base of the logarithm that we optimise through hyperparameter search. The background logit remains unaffected because of the assumption that the object distribution is the same in train and test set $p_s(o, u) \approx p_t(o, u)$, (this assumption is taken from [63, 97]).

To this end, Eq. 6 can get good performance as shown in our ablation study but it is limited because the assumption that $p(y, u) = p(y) \cdot p(u)$ is not correct. In the real world, the object detection distribution has a strong center bias, as shown in Fig.3 and discussed in [62]. Furthermore, the location is correlated with the class [34], therefore, $p(y, u) \neq p(y) \cdot p(u)$. As we show, the location provides valuable information for the long-tailed detection task and we enhance Eq. 6 by fusing location information.

[1]Typically object detectors use an extra background logit *bg* to implicitely learn $p(o, u)$.

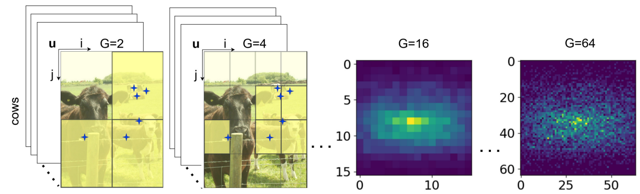

Figure 3. Different grid sizes affect the object distribution estimation. When the grid is coarse, e.g., 1×1 or 2×2, there is no or little location information. When it is finer, e.g., 64×64, the probability is sparse, giving noisy estimates for the rare classes.

3.2.2. Location-class dependence.

One way to compute $p(y, u)$ is by counting the class occurrences $n_y(\mathbf{u})$ along locations that fall inside the cell $\mathbf{u} = [i, j]$ as shown in Fig. 3-left. To do so, we discretise the space of various image resolutions into a normalised square grid $U_{G \times G}$ of fixed size $G \in \mathbf{N}$ and count class occurrences inside every grid cell. Accordingly, the grid dependent calibration is defined as:

$$\mathrm{C}_G(z_{y,\mathbf{u}}) = \begin{cases} z_{y,\mathbf{u}} - \log_\beta(p_s(y, \mathbf{u})) + \log_\beta(p_t(y, \mathbf{u})) \\ z_{y,\mathbf{u}}, \quad if \quad y = \mathrm{bg}, \end{cases} \quad (7)$$

where $z_{y,\mathbf{u}}$ is the predicted logit whose center falls inside the discrete cell $\mathbf{u} = [i, j]$ and $p_t(y, \mathbf{u})$ is uniform, i.e., $p_t(y, \mathbf{u}) = \frac{1}{C} \cdot \frac{1}{G^2}$.

However, the choice of the grid size G largely affects the estimation of $p(y, u)$, as shown in Fig.3-right. For example, if we use smaller G, the generic object distribution becomes denser and little location information is encoded. If we use larger G, the distribution becomes sparse. This is problematic for the rare classes because they are already sparse and their location information is noisy. In Table 5-e, we show that this baseline shows limited performance.

3.3. Calibration using fractals

To solve the sparsity problem introduced by the grid-size, we use the fractal dimension Φ [64], which is a metric independent of the grid size G. To calculate Φ, we use the box-counting method [70]:

$$\Phi(y) = \lim_{G \to \infty} \frac{\log \sum_{j=0}^{G-1} \sum_{i=0}^{G-1} \mathbb{1}(n_y(\mathbf{u}))}{\log(G)}, \quad (8)$$

where $\mathbb{1}$ is the indicator function. For objects in 2D images, as in our case, $\Phi(y) \in [0, 2]$, where 0 is only one object, 1 shows that the objects lie across a line and 2 shows that they are located uniformly across the image space.

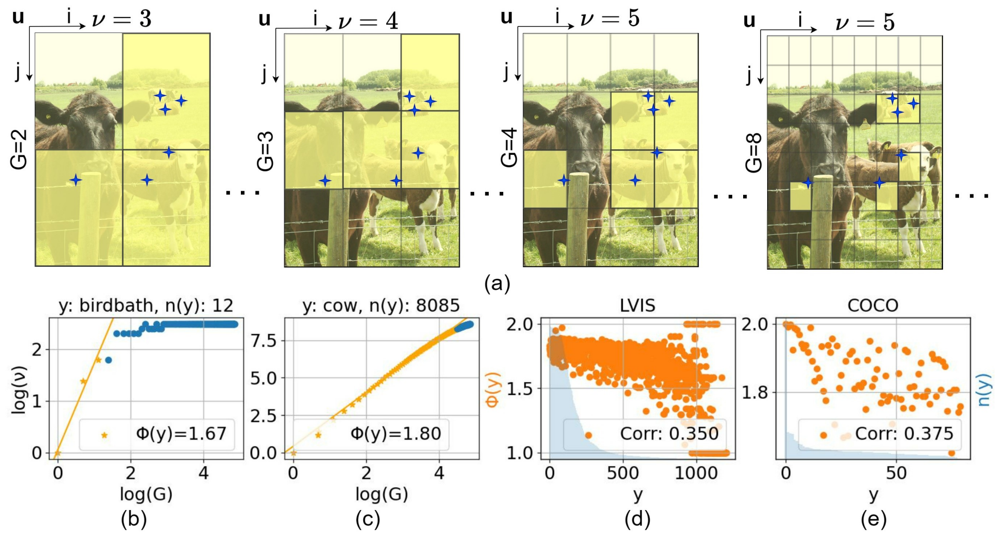

Figure 4. a) An example of the box counting method for the class *cow*. It iteratively counts the boxes ν containing its center, as G grows. b-c) The blue points are all $G - \nu$ pairs, out of them only the orange points are used to calculate the slope Φ based on the quadratic rule $G = \lfloor\sqrt{n_y}\rfloor$. d-e) Fractal dimension and class frequency are weakly correlated, showing that the Φ complements the frequency statistic.

For brevity, we rewrite $\nu_y = \sum_{j=0}^{G-1}\sum_{i=0}^{G-1}\mathbb{1}(n_y(\mathbf{u}))$ and we give an example in Fig. 4-a. In practice, Eq. 8 cannot be computed because by increasing G, the computation becomes intractable. Instead, we approximate Φ, by evaluating nominator-denominator pairs of Eq. 8 for various values of G up to a threshold t and then fit a line to those points. The slope of this line approximates $\Phi(y)$, because it considers all computed $G - \nu_y$ pairs.

Dealing with rare classes. To select the threshold t, we use the quadratic rule $G \leq t = \lfloor\sqrt{n_y}\rfloor$. The motivation for this rule is simple, for example, if an object is rare, e.g., it appears 4 times in the whole training set, then it can, at most, fill a grid of size 2×2. For objects with fewer occurrences *we cannot compute* Φ *and thus we assign* $\Phi = 1$. Using this rule, we define the maximum number of pairs that are required for fitting the "fractality" line highlighted in orange in Fig. 4-b and Fig. 4-c. For example, the rare object *birdbath* appears 12 times in the training set, thus we use the first three orange points in Fig. 4-b that correspond to $G = \{1, 2, 3\}$, to fit the "fractality" line, resulting in a large $\Phi = 1.67$. This rule ensures that the fractal dimension computation does not underestimate the rare classes and it gives robust measurements that increase rare class performance as shown in our experiments. For the *cow* object that has larger frequency we use more $G - \nu$ orange pairs to fit the line as shown in Fig. 4-c, resulting in $\Phi = 1.80$.

Relationship to frequency. As shown in Fig. 4-d and e, the fractal dimension weakly correlates with frequency, i.e., 0.35 for LVIS and 0.375 for COCO using Pearson correlation. Also, there are many rare classes with large $\Phi \approx 2$, showing that our threshold selection technique is robust for small sample sets. The weak correlation between frequency and fractal dimension, highlights that our method complements the frequency statistics and adds new information to the model resulting in superior performance as shown experimentally.

Usage. After calculating Φ for all classes in the training set, we store the fractal dimensions in the memory. During inference, we fuse them with the model's prediction using the space-aware calibration (S):

$$\mathrm{S}(z_y) = \begin{cases} \frac{\sigma(z_y)}{\Phi(y)^\lambda}, & y \in \{1, ..., C\} \\ \sigma(z_y), & y = \text{bg}, \end{cases} \tag{9}$$

where $\sigma(z_y) \in (0, 1)$ is the model's prediction for class y, with $\sigma()$ the Softmax activation, and $\lambda \geq 0$ is a hyperparameter. Eq. 9 downweighs the classes that appear most uniformly and it upweighs the classes that appear less uniformly. In practice, (S) calibration forces the detector to predict both frequent and rare classes uniformly across all spatial locations. For example, in Fig. 2-bottom-right after applying our method, the model detects *hats* and *tiaras* in every image location and not just the in top of the images as the baseline. Intuitively, this removes the spatial bias, producing balanced detectors that have better performance as shown in our ablation and our qualitative results.

3.4. Localised Calibration

By putting Eq. 6 and Eq. 9 together, we get the final FRActal CALibration (FRACAL) as:

$$\mathrm{F}(z_y) = \frac{\mathrm{S}(\mathrm{C}(z_y))}{\sum_{j=1}^{C+1}\mathrm{S}(\mathrm{C}(z_j))}. \tag{10}$$

Our proposed method tackles the classification imbalance using additional space statistics. On the classification axis, we use the class priors $p_s(y)$ and perform logit adjustments. On the space axis, we use the fractal dimension $\Phi(y)$ to perform a space-aware calibration that accounts for the object's location distribution $p_s(y,u)$. In Eq. 10, we renormalise both foreground and background logits to preserve a probabilistic prediction after the space calibration in Eq. 9.

Extending to binary classifiers. In long-tailed object detection there are many works that use only binary classifiers [2, 28, 32, 40, 72, 73, 79]. In this case, the logit z_i performs two tasks simultaneously: It discriminates among the foreground classes and performs background-to-foreground classification. Thus, to correctly apply foreground calibration, we first need to decouple the foreground and background predictions. To do so, we filter out the background proposals using the model's predictions as follows:

$$\mathrm{F}_b(z_i) = \eta(\mathrm{C}(z_i) - \log_\beta(\frac{\Phi(y)^\lambda}{\sum_i^C \Phi(i)^\lambda}) + \log_\beta(\frac{1}{C})) \cdot \eta(z_i), \tag{11}$$

where $\eta(z_i)$ is the sigmoid activation function that acts as a filter for low-scoring proposals. Compared to Eq. 10, Eq. 11 performs class calibration and space calibration in logit space, lowering the false-positive detection rate.

4. Results

4.1. Experimental Setup

We use the Large Vocabulary Instance Segmentation (LVISv1) dataset [20] which consists of $100k$ images in the train set and $20k$ images in the validation set. This dataset has $1,203$ classes grouped according to their image frequency into *frequent* (those that contain >100 images), *common* (those that contain $10 \sim 100$ images) and *rare* classes (those that contain <10 images) in the training set. For evaluation, we use average mask precision AP_m, average box precision AP_b and AP^m_f, AP^m_c and AP^m_r that correspond to AP^m for *frequent*, *common* and *rare* classes. Unless mentioned, we use Mask R-CNN [22] with FPN [49], ResNet50 [21], repeat factor sampler (RFS) [20], Normalised Mask [76], CARAFE [75] and we train the baseline model using the 2x schedule [23], SGD, learning rate 0.02 and weight decay $1e-4$. For Swin models, we train the baseline models with the 1x schedule, RFS, AdamW [36] and 0.001 learning rate. During inference, we set the IoU threshold at 0.3 and the mask threshold at 0.4 and FRACAL is applied before the non-maximum suppression step thus it has little overhead as shown in the Appendix.

4.2. Main Results

Comparison to SOTA. In Table 2, we compare FRACAL to the state-of-the-art using ResNet50 and ResNet101. Using ResNet50, FRACAL significantly surpasses GOL [2] by 0.9 percentage points (pp) in AP^m and by 1.6pp AP^m_r. On ResNet101 FRACAL achieves 29.8% AP^m and 24.5% AP^m_r, outbesting GOL by 0.8pp and 1.7pp respectively.

FRACAL achieves excellent results not only for rare categories but also for frequent ones, due to the use of fractal dimension, which allows the model to upscale the predictions of frequent but non-uniformly located categories. It achieves 31.5% AP^m_f with ResNet50 and 32.7% AP^m_f with ResNet101, surpassing ECM [32] by 0.4pp.

Compared to the previous post-calibration method, Norcal [63], FRACAL increases performance by 3.4pp AP^m, 3.7pp AP^m_r, 3.8pp AP^m_c, 2.5pp AP^m_f and 2.3pp AP^b using ResNet50. This is because FRACAL boosts both rare and frequent categories via classification and space calibration, respectively, while Norcal only boosts the rare categories and lacks space information.

We also compare our method with Transformer backbones. Using Swin-T, FRACAL considerably outperforms Seesaw [76] by 1.2pp AP^m, 1.7pp AP^m_r, 1.2pp AP^m_c, 1.0pp AP^m_f and 0.8pp AP^b as shown in Table 3. Using Swin-S, FRACAL largely surpasses Seesaw in all metrics and particularly in AP^m_r by 2.2pp which is a significant 8.6% relative improvement for the rare classes. Finally, we scale our method to Swin-B pretrained on ImageNet22K, and we show that it substantially enhances the AP^m by 1.9pp, the AP^m_r by 6.6pp and the AP^b by 2.3pp.

Results on object detectors. We evaluate FRACAL with common object detectors in Table 4 using ResNet50. FRACAL boosts the overall and rare category performance of both one-stage detectors such as ATSS [93] or GFLv2 [45] and two-stage detectors such as Cascade RCNN [6] and APA-MaskRCNN [4]. Note that on sigmoid-detectors such as ATSS or GFLv2, FRACAL largely boosts the performance of rare and common categories but it slightly reduces the performance of frequent categories. Since the sigmoid activation performs independent classification, the binary version of FRACAL struggles to properly calibrate the predicted unnormalised vector. This limitation was also found in previous works [63] which also reported that binary logit adjustment produces performance trade-offs between frequent and rare categories. For softmax-based detectors, such as Cascade RCNN and APA, FRACAL boosts all categories. In the Appendix, we discuss FRACAL's expected calibration error and detection error.

4.3. Ablation Study and Analysis

The effect of each module. FRACAL consists of simple modules that we ablate in Table 5-a. First, MaskRCNN with CARAFE [75], normalised mask predictor [76], cosine classifier [76] and random sampler achieves 22.8% AP^m and 8.2% rare category AP^m_r. On top of this, the fractal dimension calibration (S) improves AP^m and AP^m_r by 2.8pp and 5.5pp respectively.

Using only the classification calibration, (C), AP^m and AP^m_r are enhanced by 3.5pp and 8.3pp respectively, be-

Table 2. Comparison against SOTA on LVISv1 dataset. Our method reaches the best results in all metrics.

Method	Reference	Arch.	AP^m	AP_r^m	AP_c^m	AP_f^m	AP^b
Baseline	N/A	Mask RCNN ResNet50	25.7	15.8	25.1	30.6	25.9
NorCal [63]	NeurIPS 21		25.2	19.3	24.2	29.0	26.1
GOL [2]	ECCV 22		27.7	21.4	27.7	30.4	27.5
ECM [32]	ECCV 22		27.4	19.7	27.0	31.1	27.9
CRAT w/ LOCE [81]	IJCV 24		27.5	21.2	26.8	31.0	28.2
LogN [97]	IJCV 24		27.5	21.8	27.1	30.4	28.1
FRACAL (ours)	-		**28.6**$^{+0.9}$	**23.0**$^{+1.2}$	**28.0**$^{+0.3}$	**31.5**$^{+0.4}$	**28.4**$^{+0.2}$
Baseline	N/A	Mask RCNN ResNet101	27.0	16.8	26.5	32.0	27.3
NorCal [63]	NeurIPS 21		27.3	20.8	26.5	31.0	28.1
GOL [2]	ECCV 22		29.0	22.8	29.0	31.7	29.2
ECM [32]	ECCV 22		28.7	21.9	27.9	32.3	29.4
ROG [95]	ICCV 23		28.8	21.1	29.1	31.8	28.8
CRAT w/ LOCE [81]	IJCV 24		28.8	22.0	28.6	32.0	29.7
LogN [97]	IJCV 24		29.0	22.9	28.8	31.8	**29.8**
FRACAL (ours)	-		**29.8**$^{+0.8}$	**24.5**$^{+1.5}$	**29.3**$^{+0.2}$	**32.7**$^{+0.4}$	**29.8**

Table 3. Results with MaskRCNN, Swin-T/S/B and 1x schedule.

Method	AP^m	AP_r^m	AP_c^m	AP_f^m	AP^b
Baseline-(T)	27.7	17.9	27.9	31.8	27.1
Seesaw-(T)	29.5	24.0	29.3	32.2	29.5
GOL-(T)	28.5	21.1	29.5	30.6	28.3
FRACAL-(T)	**30.7**	**25.7**	**30.5**	**33.2**	**30.3**
Baseline-(S)	30.9	21.7	31.0	34.7	31.0
Seesaw-(S)	32.4	25.6	32.8	34.9	32.9
GOL-(S)	31.5	24.1	32.3	33.8	32.0
FRACAL-(S)	**33.6**	**27.8**	**33.9**	**35.9**	**33.4**
Baseline-(B)	36.6	28.9	37.8	**38.7**	37.1
FRACAL-(B)	**38.5**	**35.5**	**39.4**	**38.7**	**39.4**

Table 4. FRACAL can be used with both Sigmoid and Softmax based detectors and improve their precision.

Method	AP^b	AP_r^b	AP_c^b	AP_f^b
ATSS [93]	25.3	15.8	23.4	**31.6**
with FRACAL (ours)	**26.7**	**20.8**	**25.9**	30.9
GFLv2 [45]	26.6	14.7	25.1	**33.5**
with FRACAL (ours)	**28.2**	**19.4**	**27.2**	33.2
GFLv2 (DCN) [45]	27.4	13.7	26.1	**34.8**
with FRACAL (ours)	**28.9**	**18.7**	**27.9**	34.5
APA [4]	26.9	14.3	26.2	33.2
with FRACAL (ours)	**29.2**	**22.1**	**28.0**	**33.7**
Cascade RCNN [6]	28.6	16.5	27.8	34.9
with FRACAL (ours)	**31.5**	**24.3**	**31.0**	**35.3**

cause this technique majorly upweights the rare classes. When (S) is added, then it further increases AP^m by 1.0pp and AP_r^m by 2.5pp compared to only (C), reaching 27.3% AP^m and 19.0% AP_r^m. This suggests that (S) is useful and the detector can benefit from space information. The same trend is observed with RFS in Table 5-d, however, both calibration methods have lower gains because RFS partly balances the classes via oversampling.

Fractal dimension coefficient. We ablate the choice of the λ coefficient in Eq. 9. As shown in Table 5-b, the optimal performance is achieved with $\lambda = 2$ which increases the rare categories by 0.6pp, the common categories by 0.7pp, the frequent categories by 0.3pp, the overall mask performance by 0.6pp and the box performance by 1.0pp.

Class calibration parameter search. We further ablate the choice of the log base β in Eq. 6, using the most common cases: 2 (bit), e (nat), and 10 (hartley). As shown in Table 5-c, the base-10 is the best as it achieves 26.3% AP^m and 16.5% AP_r^m with the random sampler and 28.0% AP^m and 22.4% AP_r^m with RFS, thus we use it for all experiments on LVIS. We also observe that further increasing β does not come with a performance improvement.

Comparison to grid-dependent calibration. We compare FRACAL against the grid-based method, Eq. 7, in Table 5-e. When $G = 1$ the method does not consider any location information because all predictions fall inside the same grid cell. This achieves the second best performance and it is the same result with the $\lambda = 0$ of Table 5-b. When the grid size G is enlarged, the performance of the rare classes drops significantly because the estimated prior distribution $p_s(y, \mathbf{u})$ becomes sparse (see Fig. 3). FRACAL does not suffer from this problem, because it re-weights all classes based on fractal dimension.

FRACAL Opposite. We further test, FRACAL-Opposite which is a variant that applies an invert weighting logic, (i.e. it upweights the uniformly located classes and downweights the non-uniform ones, which rectifies the space bias). As Table 5-f shows, our standard FRACAL achieves better AP^m and AP^b than the Opposite. This shows that it is preferable to remove spatial bias from the object detectors rather than rectify it, because it leads to balanced detectors.

Generalisation to other datasets. We test FRACAL on MS-COCO [48], V3DET [77] and OpenImages [37] to understand its generalisation ability and report the results in

Table 5. Ablations using MaskRCNN-ResNet50. C and S denote the class and location calibration.

C	S	AP^m	AP^m_r
		22.8	8.2
	✓	25.6	13.7
✓		26.3	16.5
✓	✓	**27.3**	**19.0**

(a) With random sampler.

λ	AP^m	AP^m_r	AP^m_c	AP^m_f	AP^b
0.0	28.0	22.4	27.3	31.2	27.4
1.0	28.5	23.0	**28.0**	**31.6**	28.3
2.0	**28.6**	23.0	**28.0**	31.5	**28.4**
3.0	28.5	23.2	**28.0**	31.5	**28.4**
4.0	28.5	**23.4**	27.9	31.3	**28.4**

(b) Ablation study of λ, using RFS.

β	random		RFS	
	AP^m	AP^m_r	AP^m	AP^m_r
2	19.9	14.7	19.9	18.8
e	25.1	16.6	25.8	21.1
10	26.3	16.5	**28.0**	**22.4**

(c) Ablation of β, under various samplers.

C	S	AP^m	AP^m_r
		25.7	15.8
	✓	27.7	20.7
✓		28.0	22.4
✓	✓	**28.6**	**23.0**

(d) Results using RFS [20].

Method	AP^m	AP^m_r	AP^m_c	AP^m_f	AP^b
G=1	28.0	22.4	27.3	31.2	27.4
G=2	27.1	17.5	27.2	31.1	26.6
G=4	25.0	10.5	25.4	31.1	24.9
ours	**28.6**	**23.0**	**28.0**	**31.5**	**28.4**

(e) Comparison against Grid-based methods.

Method	AP^m	AP^m_r	AP^b
FRACAL-Opposite	27.4	20.5	26.9
FRACAL	**28.6**	**23.0**	**28.4**

(f) FRACAL fusion ablation study.

Tables 6,7,8 respectively. The first two datasets are fairly balanced therefore, we do not expect our long-tailed designed detector to massively outperform the others. In Table 6, FRACAL increases the performance of all models, by an average of 0.5pp AP^b and AP^m. In Table 7, FRACAL increases the performance of APA [4] by 0.4pp AP^b. In Table 8, FRACAL outperforms ECM using CascadeRCNN by 1.7pp and it increases the performance of CAS by 2.0pp and 1.2pp using FasterRCNN and CascadeRCNN respectively.

Qualitative Analysis. In Fig. 5, we show: (a) the ground truth distribution, (b) the baseline and (c) FRACAL predicted distributions concerning all objects (1), the rare class *ferret* (2) and the frequent class *zebra* (3). FRACAL achieves better precision than the baseline because it detects more rare objects in (2-c) increasing recall and fewer frequent objects in (3-c) decreasing false positives. Regarding the spatial distributions, FRACAL increases the spatial uniformity of all predictions because it has less centered detections for all objects in (1-c), more evenly spread detections for the *ferret* in (2-c) and less centrally biased detections for the *zebra* in (3-c). This shows that FRACAL makes balanced predictions in both frequency and space perspectives, enabling higher detection performance.

Table 6. Results on COCO with MaskRCNN.

Method	AP^m	AP^m_{50}	AP^b	AP^b_{50}
ResNet-50 [21]	35.4	56.7	39.4	59.9
with FRACAL	**35.8**	**57.5**	**39.9**	**60.6**
SE-ResNet-50 [30]	36.9	58.8	40.5	61.7
with FRACAL	**37.4**	**59.5**	**41.1**	**62.4**
CB-ResNet-50 [84]	37.3	59.2	40.9	62.1
with FRACAL	**37.8**	**60.2**	**41.5**	**62.9**
Swin-T [56]	41.6	65.3	46.0	68.2
with FRACAL	**41.9**	**66.0**	**46.4**	**68.7**

5. Conclusion

We propose FRACAL, a novel post-calibration method for long-tailed object detection. Our method performs a space-aware logit adjustment, utilising the fractal dimension and incorporating space information during calibration. FRACAL majorly boosts the performance of the detectors by detecting rare classes that are evenly spread inside the image. We show that FRACAL can be easily combined with both one-stage Sigmoid detectors and two-stage Softmax segmentation models. Our method boosts the performance of detectors by up to 8.6% without training or additional inference cost, surpassing the SOTA in the LVIS benchmark and generalising well to COCO, V3Det and OpenImages.

Table 7. Results on V3Det [77] using FasterRCNN ResNet50.

Method	AP^b	AP^b_{50}	AP^b_{75}
Normalised Layer [76]	25.3	32.8	28.1
APA [4]	29.9	37.6	32.9
APA + FRACAL (ours)	**30.3**	**37.7**	**33.2**

Table 8. Results on OpenImages [37] using ResNet50.

Method	Detector	AP^b_{50}
CAS [54]	Faster RCNN	65.0
CAS + FRACAL (ours)		**67.0**
ECM [32]	Cascade RCNN	65.8
CAS [54]		66.3
CAS + FRACAL (ours)		**67.5**

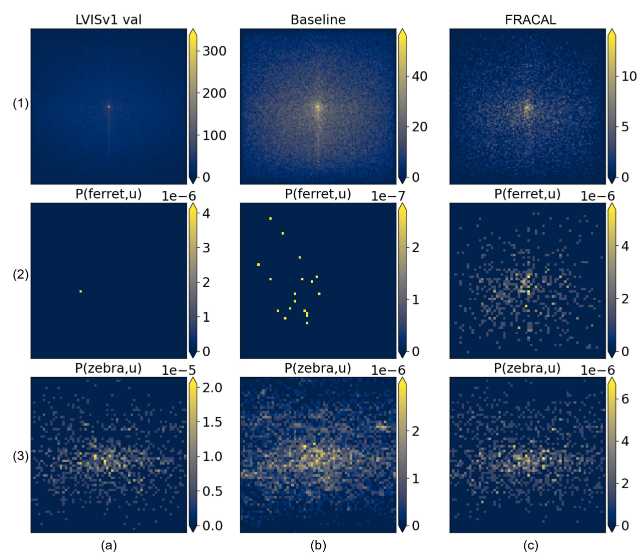

Figure 5. Detection results in LVIS, FRACAL detects more uniformly in both frequency and space axis compared to the baseline.

Acknowledgments. Anh Nguyen was supported by Royal Society ISPF International Collaboration Awards 2023 (Japan) ICA\R1\231067. The project was co-funded by ViTac: Visual-Tactile Synergy for Handling Flexible Materials" (EP/T033517/2).

References

[1] Emanuel Sanchez Aimar, Arvi Jonnarth, Michael Felsberg, and Marco Kuhlmann. Balanced product of calibrated experts for long-tailed recognition. In *CVPR*, 2023. 2

[2] Konstantinos Panagiotis Alexandridis, Jiankang Deng, Anh Nguyen, and Shan Luo. Long-tailed instance segmentation using gumbel optimized loss. In *ECCV*, 2022. 3, 4, 6, 7

[3] Konstantinos Panagiotis Alexandridis, Shan Luo, Anh Nguyen, Jiankang Deng, and Stefanos Zafeiriou. Inverse image frequency for long-tailed image recognition. *IEEE Transactions on Image Processing*, 2023. 1, 2, 3

[4] Konstantinos Panagiotis Alexandridis, Jiankang Deng, Anh Nguyen, and Shan Luo. Adaptive parametric activation. In *European Conference on Computer Vision*, pages 455–476. Springer, 2024. 3, 6, 7, 8

[5] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. Yolact: Real-time instance segmentation. In *CVPR*, 2019. 2

[6] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: High quality object detection and instance segmentation. *tPAMI*, 2019. 2, 6, 7

[7] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. In *NeurIPS*, 2019. 2

[8] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 1, 2

[9] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, et al. Hybrid task cascade for instance segmentation. In *CVPR*, 2019. 2

[10] Shoufa Chen, Peize Sun, Yibing Song, and Ping Luo. Diffusiondet: Diffusion model for object detection. *arXiv preprint arXiv:2211.09788*, 2022. 1, 2

[11] Jiequan Cui, Zhisheng Zhong, Shu Liu, Bei Yu, and Jiaya Jia. Parametric contrastive learning. In *ICCV*, 2021. 1

[12] Jiequan Cui, Shu Liu, Zhuotao Tian, Zhisheng Zhong, and Jiaya Jia. Reslt: Residual learning for long-tailed recognition. *tPAMI*, 2022. 2

[13] Jiequan Cui, Zhisheng Zhong, Zhuotao Tian, Shu Liu, Bei Yu, and Jiaya Jia. Generalized parametric contrastive learning. *tPAMI*, 2023. 2

[14] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *CVPR*, 2019. 2

[15] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 1

[16] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88:303–338, 2010. 4

[17] Chengjian Feng, Yujie Zhong, and Weilin Huang. Exploring classification equilibrium in long-tailed object detection. In *ICCV*, 2021. 3

[18] Xiao Gu, Yao Guo, Zeju Li, Jianing Qiu, Qi Dou, Yuxuan Liu, Benny Lo, and Guang-Zhong Yang. Tackling long-tailed category distribution under domain shifts. In *ECCV*, 2022. 2

[19] Zhangxuan Gu, Haoxing Chen, Zhuoer Xu, Jun Lan, Changhua Meng, and Weiqiang Wang. Diffusioninst: Diffusion model for instance segmentation. *arXiv preprint arXiv:2212.02773*, 2022. 2

[20] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *CVPR*, 2019. 2, 3, 6, 8

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 6, 8

[22] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017. 2, 3, 6

[23] Kaiming He, Ross Girshick, and Piotr Dollár. Rethinking imagenet pre-training. In *ICCV*, 2019. 6

[24] Yin-Yin He, Jianxin Wu, and Xiu-Shen Wei. Distilling virtual examples for long-tailed recognition. In *ICCV*, 2021. 2

[25] Yin-Yin He, Peizhen Zhang, Xiu-Shen Wei, Xiangyu Zhang, and Jian Sun. Relieving long-tailed instance segmentation via pairwise class balance. In *CVPR*, 2022. 3

[26] Youngkyu Hong, Seungju Han, Kwanghee Choi, Seokjun Seo, Beomsu Kim, and Buru Chang. Disentangling label distribution for long-tailed visual recognition. In *CVPR*, 2021. 1, 2, 3

[27] Yan Hong, Jianfu Zhang, Zhongyi Sun, and Ke Yan. Safa: Sample-adaptive feature augmentation for long-tailed image classification. In *ECCV*, 2022. 2

[28] Ting-I Hsieh, Esther Robb, Hwann-Tzong Chen, and Jia-Bin Huang. Droploss for long-tail instance segmentation. In *AAAI*, 2021. 3, 6

[29] Yen-Chi Hsu, Cheng-Yao Hong, Ming-Sui Lee, Davi Geiger, and Tyng-Luh Liu. Abc-norm regularization for fine-grained and long-tailed image classification. *IEEE Transactions on Image Processing*, 2023. 2

[30] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, 2018. 8

[31] Zhaojin Huang, Lichao Huang, Yongchao Gong, Chang Huang, and Xinggang Wang. Mask scoring r-cnn. In *CVPR*, 2019. 2

[32] Jang Hyun Cho and Philipp Krähenbühl. Long-tail detection with effective class-margins. In *ECCV*, 2022. 2, 6, 7, 8

[33] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. In *ICLR*, 2020. 2, 3

[34] Osman Semih Kayhan and Jan C van Gemert. On translation invariance in cnns: Convolutional layers can exploit absolute spatial location. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14274–14285, 2020. 2, 3, 4

[35] Salman H Khan, Munawar Hayat, Mohammed Bennamoun, Ferdous A Sohel, and Roberto Togneri. Cost-sensitive learning of deep feature representations from imbalanced data. *IEEE transactions on neural networks and learning systems*, 2017. 2

[36] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[37] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision*, 128(7):1956–1981, 2020. 7, 8

[38] Banghuai Li. Adaptive hierarchical representation learning for long-tailed object detection. In *CVPR*, 2022. 3

[39] Bolian Li, Zongbo Han, Haining Li, Huazhu Fu, and Changqing Zhang. Trustworthy long-tailed classification. In *CVPR*, 2022. 2

[40] Bo Li, Yongqiang Yao, Jingru Tan, Gang Zhang, Fengwei Yu, Jianwei Lu, and Ye Luo. Equalized focal loss for dense long-tailed object detection. In *CVPR*, 2022. 3, 6

[41] Jun Li, Zichang Tan, Jun Wan, Zhen Lei, and Guodong Guo. Nested collaborative learning for long-tailed visual recognition. In *CVPR*, 2022. 2

[42] Jian Li, Ziyao Meng, Daqian Shi, Rui Song, Xiaolei Diao, Jingwen Wang, and Hao Xu. Fcc: Feature clusters compression for long-tailed visual recognition. In *CVPR*, 2023. 2

[43] Tianhao Li, Limin Wang, and Gangshan Wu. Self supervision to distillation for long-tailed visual recognition. In *ICCV*, 2021. 2

[44] Tianhong Li, Peng Cao, Yuan Yuan, Lijie Fan, Yuzhe Yang, Rogerio S Feris, Piotr Indyk, and Dina Katabi. Targeted supervised contrastive learning for long-tailed recognition. In *CVPR*, 2022. 2

[45] Xiang Li, Wenhai Wang, Xiaolin Hu, Jun Li, Jinhui Tang, and Jian Yang. Generalized focal loss v2: Learning reliable localization quality estimation for dense object detection. In *CVPR*, 2021. 3, 6, 7

[46] 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 *CVPR*, 2020. 1, 3

[47] Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. In *ECCV*, 2022. 2

[48] 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 *ECCV*, 2014. 1, 7

[49] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 6

[50] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017. 2

[51] Zachary Lipton, Yu-Xiang Wang, and Alexander Smola. Detecting and correcting for label shift with black box predictors. In *ICML*, 2018. 3

[52] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *ECCV*, 2016. 2

[53] Xuantong Liu, Jianfeng Zhang, Tianyang Hu, He Cao, Yuan Yao, and Lujia Pan. Inducing neural collapse in deep long-tailed learning. In *AISTAT*, 2023. 2

[54] Yu Liu, Guanglu Song, Yuhang Zang, Yan Gao, Enze Xie, Junjie Yan, Chen Change Loy, and Xiaogang Wang. 1st place solutions for openimage2019–object detection and instance segmentation. *arXiv preprint arXiv:2003.07557*, 2020. 8

[55] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *CVPR*, 2019. 1, 2

[56] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021. 8

[57] Chengqi Lyu, Wenwei Zhang, Haian Huang, Yue Zhou, Yudong Wang, Yanyi Liu, Shilong Zhang, and Kai Chen. Rtmdet: An empirical study of designing real-time object detectors. *arXiv preprint arXiv:2212.07784*, 2022. 1

[58] Chengcheng Ma, Ismail Elezi, Jiankang Deng, Weiming Dong, and Changsheng Xu. Three heads are better than one: Complementary experts for long-tailed semi-supervised learning. In *AAAI*, 2024. 2

[59] Yanbiao Ma, Licheng Jiao, Fang Liu, Shuyuan Yang, Xu Liu, and Lingling Li. Curvature-balanced feature manifold learning for long-tailed classification. In *CVPR*, 2023. 2

[60] Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. In *ICLR*, 2021. 2, 3

[61] Ravi Teja Mullapudi, Fait Poms, William R Mark, Deva Ramanan, and Kayvon Fatahalian. Background splitting: Finding rare classes in a sea of background. In *CVPR*, 2021. 2

[62] Kemal Oksuz, Baris Can Cam, Sinan Kalkan, and Emre Akbas. Imbalance problems in object detection: A review. *tPAMI*, 2020. 2, 4

[63] Tai-Yu Pan, Cheng Zhang, Yandong Li, Hexiang Hu, Dong Xuan, Soravit Changpinyo, Boqing Gong, and Wei-Lun Chao. On model calibration for long-tailed object detection and instance segmentation. In *NeurIPS*, 2021. 2, 3, 4, 6, 7

[64] Chinmaya Panigrahy, Ayan Seal, Nihar Kumar Mahato, and Debotosh Bhattacharjee. Differential box counting methods

for estimating fractal dimension of gray-scale images: A survey. *Chaos, Solitons & Fractals*, 126:178–202, 2019. 4
[65] Sarah Parisot, Pedro M Esperança, Steven McDonagh, Tamas J Madarasz, Yongxin Yang, and Zhenguo Li. Long-tail recognition via compositional knowledge transfer. In *CVPR*, 2022. 2
[66] Seulki Park, Youngkyu Hong, Byeongho Heo, Sangdoo Yun, and Jin Young Choi. The majority can help the minority: Context-rich minority oversampling for long-tailed classification. In *CVPR*, 2022. 2
[67] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *CVPR*, 2017. 2
[68] Jiawei Ren, Cunjun Yu, Shunan Sheng, Xiao Ma, Haiyu Zhao, Shuai Yi, and Hongsheng Li. Balanced meta-softmax for long-tailed visual recognition. In *NeurIPS*, 2020. 1, 2, 3
[69] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015. 2
[70] Manfred Schroeder. *Fractals, chaos, power laws: Minutes from an infinite paradise*. Courier Corporation, 2009. 2, 4
[71] Peize Sun, Rufeng Zhang, Yi Jiang, Tao Kong, Chenfeng Xu, Wei Zhan, Masayoshi Tomizuka, Lei Li, Zehuan Yuan, Changhu Wang, et al. Sparse r-cnn: End-to-end object detection with learnable proposals. In *CVPR*, 2021. 2
[72] Jingru Tan, Changbao Wang, Buyu Li, Quanquan Li, Wanli Ouyang, Changqing Yin, and Junjie Yan. Equalization loss for long-tailed object recognition. In *CVPR*, 2020. 3, 6
[73] 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 *CVPR*, 2021. 3, 6
[74] Kaihua Tang, Mingyuan Tao, Jiaxin Qi, Zhenguang Liu, and Hanwang Zhang. Invariant feature learning for generalized long-tailed classification. In *ECCV*, 2022. 2
[75] Jiaqi Wang, Kai Chen, Rui Xu, Ziwei Liu, Chen Change Loy, and Dahua Lin. Carafe: Content-aware reassembly of features. In *ICCV*, 2019. 2, 6
[76] 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 *CVPR*, 2021. 1, 3, 6, 8
[77] Jiaqi Wang, Pan Zhang, Tao Chu, Yuhang Cao, Yujie Zhou, Tong Wu, Bin Wang, Conghui He, and Dahua Lin. V3det: Vast vocabulary visual detection dataset. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 19844–19854, 2023. 7, 8
[78] 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 *ECCV*, 2020. 1
[79] Tong Wang, Yousong Zhu, Chaoyang Zhao, Wei Zeng, Jinqiao Wang, and Ming Tang. Adaptive class suppression loss for long-tail object detection. In *CVPR*, 2021. 1, 3, 6
[80] Tong Wang, Yousong Zhu, Yingying Chen, Chaoyang Zhao, Bin Yu, Jinqiao Wang, and Ming Tang. C2am loss: Chasing a better decision boundary for long-tail object detection. In *CVPR*, 2022. 3
[81] Tao Wang, Li Yuan, Xinchao Wang, and Jiashi Feng. Learning box regression and mask segmentation under long-tailed distribution with gradient transfusing. *International Journal of Computer Vision*, pages 1–17, 2024. 7
[82] Xudong Wang, Long Lian, Zhongqi Miao, Ziwei Liu, and Stella Yu. Long-tailed recognition by routing diverse distribution-aware experts. In *ICLR*, 2021. 2
[83] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Learning to model the tail. In *NeurIPS*, 2017. 2
[84] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *ECCV*, 2018. 8
[85] Jialian Wu, Liangchen Song, Tiancai Wang, Qian Zhang, and Junsong Yuan. Forest r-cnn: Large-vocabulary long-tailed object detection and instance segmentation. In *PACM International Conference on Multimedia*, 2020. 3
[86] Zhengzhuo Xu, Ruikang Liu, Shuo Yang, Zenghao Chai, and Chun Yuan. Learning imbalanced data with vision transformers. In *CVPR*, 2023. 1
[87] Lu Yang, He Jiang, Qing Song, and Jun Guo. A survey on long-tailed visual recognition. *IJCV*, 2022. 2
[88] Yuzhe Yang, Hao Wang, and Dina Katabi. On multi-domain long-tailed recognition, imbalanced domain generalization and beyond. In *ECCV*, 2022. 2
[89] Han-Jia Ye, Hong-You Chen, De-Chuan Zhan, and Wei-Lun Chao. Identifying and compensating for feature deviation in imbalanced deep learning. *arXiv preprint arXiv:2001.01385*, 2020. 2
[90] Yuhang Zang, Chen Huang, and Chen Change Loy. Fasa: Feature augmentation and sampling adaptation for long-tailed instance segmentation. In *ICCV*, 2021. 1, 2, 3
[91] Cheng Zhang, Tai-Yu Pan, Yandong Li, Hexiang Hu, Dong Xuan, Soravit Changpinyo, Boqing Gong, and Wei-Lun Chao. Mosaicos: a simple and effective use of object-centric images for long-tailed object detection. In *ICCV*, 2021. 3
[92] Cheng Zhang, Tai-Yu Pan, Tianle Chen, Jike Zhong, Wenjin Fu, and Wei-Lun Chao. Learning with free object segments for long-tailed instance segmentation. In *ECCV*, 2022. 3
[93] 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 *CVPR*, 2020. 6, 7
[94] Songyang Zhang, Zeming Li, Shipeng Yan, Xuming He, and Jian Sun. Distribution alignment: A unified framework for long-tail visual recognition. In *CVPR*, 2021. 2
[95] Shaoyu Zhang, Chen Chen, and Silong Peng. Reconciling object-level and global-level objectives for long-tail detection. In *ICCV*, 2023. 3, 7
[96] Yifan Zhang, Bingyi Kang, Bryan Hooi, Shuicheng Yan, and Jiashi Feng. Deep long-tailed learning: A survey. *tPAMI*, 2023. 2
[97] Liang Zhao, Yao Teng, and Limin Wang. Logit normalization for long-tail object detection. *arXiv preprint arXiv:2203.17020*, 2022. 2, 3, 4, 7
[98] Yan Zhao, Weicong Chen, Xu Tan, Kai Huang, and Jihong Zhu. Adaptive logit adjustment loss for long-tailed visual recognition. In *AAAI*, 2022. 2

[99] Zhisheng Zhong, Jiequan Cui, Yibo Yang, Xiaoyang Wu, Xiaojuan Qi, Xiangyu Zhang, and Jiaya Jia. Understanding imbalanced semantic segmentation through neural collapse. In *CVPR*, 2023. 2

[100] Yixuan Zhou, Yi Qu, Xing Xu, and Hengtao Shen. Imbsam: A closer look at sharpness-aware minimization in class-imbalanced recognition. *arXiv preprint arXiv:2308.07815*, 2023. 2

[101] Zhipeng Zhou, Lanqing Li, Peilin Zhao, Pheng-Ann Heng, and Wei Gong. Class-conditional sharpness-aware minimization for deep long-tailed recognition. In *CVPR*, 2023. 2

[102] Jianggang Zhu, Zheng Wang, Jingjing Chen, Yi-Ping Phoebe Chen, and Yu-Gang Jiang. Balanced contrastive learning for long-tailed visual recognition. In *CVPR*, 2022. 1, 2

[103] Linchao Zhu and Yi Yang. Inflated episodic memory with region self-attention for long-tailed visual recognition. In *CVPR*, 2020. 2

[104] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable {detr}: Deformable transformers for end-to-end object detection. In *ICLR*, 2021. 2