Equalization Loss v2: A New Gradient Balance Approach for Long-tailed Object Detection

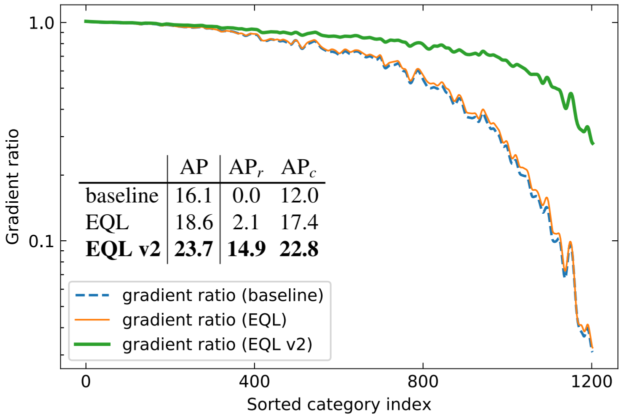
Jingru Tan[1] Xin Lu[2] Gang Zhang[3]
Changqing Yin[1] Quanquan Li[2]
[1]Tongji University [2]SenseTime Research [3]Tsinghua University
{tjr120,yinchangqing}@tongji.edu.cn, {luxin,liquanquan}@sensetime.com
zhang-g19@mails.tsinghua.edu.cn

Abstract

Recently proposed decoupled training methods emerge as a dominant paradigm for long-tailed object detection. But they require an extra fine-tuning stage, and the disjointed optimization of representation and classifier might lead to suboptimal results. However, end-to-end training methods, like equalization loss (EQL), still perform worse than decoupled training methods. In this paper, we reveal the main issue in long-tailed object detection is the imbalanced gradients between positives and negatives, and find that EQL does not solve it well. To address the problem of imbalanced gradients, we introduce a new version of equalization loss, called equalization loss v2 (EQL v2), a novel gradient guided reweighing mechanism that rebalances the training process for each category independently and equally. Extensive experiments are performed on the challenging LVIS benchmark. EQL v2 outperforms origin EQL by about 4 points overall AP with 14 $\sim$ 18 points improvements on the rare categories. More importantly, it also surpasses decoupled training methods. Without further tuning for the Open Images dataset, EQL v2 improves EQL by 7.3 points AP, showing strong generalization ability. Codes have been released at https://github.com/tztztztztz/eqlv2

1. Introduction

Object detection is a fundamental computer vision task that aims to recognize and locate objects of a set of predefined categories. Modern object detectors [31, 30, 27, 24, 25, 1] have shown promising results on some conventional benchmarks such as COCO [26] and PASCAL VOC [9]. Collected images in these datasets have been carefully selected and the quantities of each category are relatively balanced. However, in natural images, quantities of categories subject to a long-tailed Zipfian distribution. It means that, in a realistic scenario, we are confronted with a more complex situation that the obtained objects show an extreme imbalance in different categories.



	AP	AP_r	AP_c
baseline	16.1	0.0	12.0
EQL	18.6	2.1	17.4
EQL v2	**23.7**	**14.9**	**22.8**

Figure 1: Visualization of accumulative gradients ratio of different trained models. Best view in color. The x-axis is the sorted category index of 1203 categories of LVIS dataset. The y-axis is the accumulative gradient ratio of positives to negatives. Here gradient is the gradient of the output logits with respect to classification loss. AP_r and AP_c are the AP for rare and common categories.

The difficulty of training detectors on a long-tailed dataset mainly comes from two aspects. First, deep learning methods are hungry for data, but annotations of tail classes (classes with few samples) might be insufficient for training. Second, the model tends to bias towards head classes (classes with many samples) since the head class objects are the overwhelming majority in the entire datasets.

Current state-of-the-art approaches are based on decoupled training schema [18, 23, 39]. In general, decoupled training involves a two-stage pipeline that learns representations under the imbalance dataset at the first stage, then re-balances the classifier with frozen representation at the second stage. Despite the success of the decoupled training, it needs an extra fine-tuning stage in training phase. In addition, the representation could be suboptimal since it is not jointly learned with the classifier. So a natural question to ask is: could end-to-end training methods match or surpass the accuracy of decoupled training methods?

Recently, Tan *et al.* [36] propose the Equalization Loss (EQL) [36], an end-to-end re-weighing loss function, to protect the learning of tail categories by blocking some negative gradients. Although EQL makes improvements to long-tailed training, the accuracy gap between the end-to-end and decoupled training approaches still exists. To take a step forward, we analyze the gradient statistics of EQL. Here we plot the positive gradient to negative gradient ratio accumulated in the entire training process for each category classifier, as present in Figure 1. The key observation is: for head categories, the ratio is close to 1, which means the positive gradients and the negative gradients have similar magnitude; for tail categories, the gradients are near 0, which means the positive gradients are overwhelmed by the negative gradients. Therefore the gradient ratio could indicate whether a classifier is trained in balance. Compared with the baseline (blue line), the gradient ratio of EQL (orange line) just increases slightly.

In this paper, we propose a new version of equalization loss, called equalization loss v2 (EQL v2) which improves the long-tailed object detection by balancing the positive to negative gradient ratio. In EQL v2, we first model the detection problem as a set of independent sub-tasks, each task for one category. Next, we propose a gradient guided re-weighing mechanism to balance the training process of each task independently and equally. Specifically, the accumulated gradient ratio is used as an indicator to up-weight the positive gradients and down-weight the negative gradients. It dynamically controls the training of all sub-tasks and each sub-task is treated equally with the same simple re-weighing rule. The positive to negative gradient ratio of EQL v2 are shown in Figure 1 (green line). Compared to the baseline and EQL, EQL v2 achieves a more balanced training for most categories.

We conduct experiments on two long-tailed object detection dataset, LVIS [12] and OpenImages [20]. On LVIS, compared to the baseline models, including Mask R-CNN [14] and Cascade Mask R-CNN [1], it increases overall AP by about 6 points and gains 17 $\sim$ 20 points AP for tail categories. It outperforms EQL by about 4 points AP. In addition, EQL v2 surpasses all of the existing long-tailed object detection methods, including end-to-end training and decoupled training methods. On OpenImages, EQL v2 achieves a 9 points AP gain over the baseline model with the same hyper-parameters as on LVIS, which shows the good generalization ability.

2. Related Work

General object detection. Modern object detection frameworks rely on the outstanding ability of classification powered by convolutional neural networks (CNN) [34, 35, 15]. They can be divided into region-based detectors [11, 10, 31, 24, 25, 14, 1, 4] and anchor-free detectors [38, 46, 19, 21, 8, 45] depending on the concept they want to classify. However, all those frameworks are developed under the condition of balanced data. When it comes to the long-tailed distribution of data, the performance deteriorates severely due to the imbalanced among categories.

Long-tailed image classification. Common solutions for long-tailed image classification are data re-sampling and loss reweighing. However, data re-sampling [32, 29, 13, 3] methods have to access the pre-computed statistics of data distribution and might make models under the risks of over-fitting for tail classes and under-fitting for head classes. For the loss re-weighing methods, including instance-level [25, 22, 33] ones and class-level ones [7, 40, 17, 44, 2], they suffer from the sensitive hyper-parameters, the optimal setting for different dataset might vary largely and finding them takes too many efforts. There are also some works trying to transfer the knowledge from head classes to tail classes. OLTR [28] designs a memory module to augment the feature for tail classes. [6, 43] augment the under-represented classes in the feature space by using the knowledge learned from head classes. Recently, the decoupled training [18] schema attracts much attention. They argue that universal representations can be learned without re-sampling, and the classifier should be re-balanced in the second fine-tuning stage with representations frozen. In spite of excellent results, the extra fine-tuning stage seems unnatural and we can not explain why the representation and the classifier have to be learned separately.

Long-tailed object detection. Long-tailed object detection is a more difficult problem than long-tailed classification. It has to find all objects with various scale in every possible location. Li *et al.* [23] empirically find methods that are designed for long-tailed image classification can not achieve good results in object detection. Tan *et al.* [36] first shows the tail classes are heavily suppressed by the head classes and they propose an equalization loss to tackle this problem by ignoring the suppressing part for tail categories. However, they think the negative suppressing comes from competition of foreground categories and ignore the impact of background proposal. Instead, we treat background and foreground uniformly. EQL also has to access the frequency of categories and uses a threshold function to explicitly split head and tail categories. LST [16] models the learning for long-tailed distribution as a kind of incremental learning, the learning switches from head classes to tail classes in several cascaded stages. SimCal [39] and Balanced Group-Softmax (BAGS) [23] follow the spirit of decoupled training. For SimCal, they train an extra classification branch with class-balanced proposals in the fine-tuning stage and combine its score with a normal trained softmax classification branch via dual inference. BAGS divides all categories into several groups based on the instance count and does softmax separately in each group to avoid the domination of head classes. In contrast, our method does not have to split categories into different groups and treat all categories equally. Moreover, we do not need the fine-tuning stage and can be trained end-to-end. Tang *et al.* [37] shows that SGD momentum makes the classifier biased towards head classes. They introduce causal intervention in training and remove the biased part for tail classes in inference. On the contrary, our method is simpler and more efficient, and keeps consistent between training and inference.

3. Equalization Loss v2

In this section, we introduce the Equalization Loss v2. We begin by revisiting the entanglement of instances and tasks in Section 3.1, then present our novel gradient guided reweighing strategy in Section 3.2.

3.1. Entanglement of Instances and Tasks

Suppose we have a batch of instances $\mathcal{I}$ and their representations. To output logits $\mathcal{Z}$ for C categories, a weight matrix $\mathcal{W}$ is used as a linear transformation of representations. Each weight vector in $\mathcal{W}$, which we refers as a category classifier, is responsible for a specific category, *i.e.* a task. Then the output logits are transformed to an estimated probability distribution $\mathcal{P}$ by the sigmoid function. We expect that for each instance, only the corresponding classifier gives the high score while others give a low score. That is saying, *one task with positive label and $C-1$ tasks with negative labels are introduced by a single instance*. Hence, we can calculate the actual number of positive samples m_j^{pos} and negative samples m_j^{neg} for classifier j:

$$m_j^{pos} = \sum_{i \in \mathcal{I}} y_j^i, \quad m_j^{neg} = \sum_{i \in \mathcal{I}} (1 - y_j^i) \tag{1}$$

Where the y^i is the one-hot groud truth label for the i-th instance, and usually we have $\sum_j y_j^i = 1$. The ratio of expectation of positive samples to the negative samples over the dataset is then:

$$\frac{\mathbb{E}|m_j^{pos}|}{\mathbb{E}|m_j^{neg}|} = \frac{1}{\frac{N}{n_j} - 1} \tag{2}$$

Where n_j is the instance number of category j and N is the total instance number over the dataset. Equation 2 shows that if we consider each classifier separately, the ratio of the positive samples to the negative samples could have a big difference for different classifiers.

3.2. Gradient Guided Reweighing

Obviously, we have $\mathbb{E}|m_j^{pos}| \ll \mathbb{E}|m_j^{neg}|$ especially when category j is a rare category. But the ratio in Equation 2 might not be a good indicator of how balanced the training is. The reason behind it is that the influence of each sample is different. For example, the negative gradients accumulated by large quantities of easy negatives might be smaller than the positive gradients generated by a few hard positives. Therefore, we directly choose gradient statistics as a metric to indicate whether a task is in balanced training. The positive and negative gradients for each classifier's output z_j with respect to the loss $\mathcal{L}$ are formulated as:

$$\nabla_{z_j}^{pos}(\mathcal{L}) = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} y_j^i (p_j^i - 1) \tag{3}$$

$$\nabla_{z_j}^{neg}(\mathcal{L}) = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} (1 - y_j^i) p_j^i \tag{4}$$

p_j^i is the estimated probability of category j for the i-th instance. The basic idea of gradient guided balanced reweighing is that we up-weight the positive gradients and down-weight negative gradients for each classifier independently according to their *accumulated* gradient ratio of positives to negatives.

To achieve this, we first define $g_j^{(t)}$ as the ratio of *accumulated* positive gradients to negative gradients of task j until the iteration t. Then the weight for positive gradients q_j^t and negative gradients r_j^t at this iteration can be computed by:

$$q_j^{(t)} = 1 + \alpha(1 - f(g_j^{(t)})), \quad r_j^{(t)} = f(g_j^t) \tag{5}$$

Where $f(\cdot)$ is a mapping function:

$$f(x) = \frac{1}{1 + e^{-\gamma(x - \mu)}} \tag{6}$$

After obtaining q_j^t and r_j^t, we apply them to the positive gradient and negative gradient for the current batch, so the re-weighted gradients become:

$$\nabla_{z_j}^{pos'}(\mathcal{L}^{(t)}) = q_j^{(t)} \nabla_{z_j}^{pos}(\mathcal{L}^{(t)}) \tag{7}$$

$$\nabla_{z_j}^{neg'}(\mathcal{L}^{(t)}) = r_j^{(t)} \nabla_{z_j}^{neg}(\mathcal{L}^{(t)}) \tag{8}$$

Finally we update the ratio of *accumulated* positive gradients to negative gradients for the next iteration $t+1$:

$$g_j^{t+1} = \frac{\sum_{t^*=0}^{t} |\nabla_{z_j}^{pos'}(\mathcal{L}^{(t^*)})|}{\sum_{t^*=0}^{t} |\nabla_{z_j}^{neg'}(\mathcal{L}^{(t^*)})|} \tag{9}$$

method	#sampler	#epoch	AP	AP$_r$	AP$_c$	AP$_f$	AP$_b$
End-to-end Training							
(a) Softmax	random	12	16.1	0.0	12.0	27.4	16.7
(b) Sigmoid	random	12	16.5	0.0	13.1	27.3	17.2
(c) EQL [36]	random	12	18.6	2.1	17.4	27.2	19.3
(d) RFS [12]	repeat factor	12	22.2	11.5	21.2	28.0	22.9
Decoupled Training							
(e) LWS [18]	random/balance	12+12	17.0	2.0	13.5	27.4	17.5
(f) cRT [18]	random/balance	12+12	22.1	11.9	20.2	29.0	22.2
(g) BAGS [23]	random/random	12+12	23.1	13.1	22.5	28.2	23.7
(h) EQL v2 (Ours)	random	12	23.7	14.9	22.8	28.6	24.2

Table 1: Comparison with end-to-end and decoupled training methods on LVIS v1.0 `val` set with ResNet-50-FPN Mask R-CNN by 1x schedule. For cRT and LWS, they use class-balance sampler to fine-tune their model in the second stage, and BAGS uses a random sampler following the origin paper. Instead, our method train models in an end-to-end fashion without any fine-tuning stage.

obj?	neg?	pos?	AP	AP$_r$	AP$_c$	AP$_f$	AP$_b$
			16.1	0	12.0	27.4	17.2
✓			18.1	1.9	16.4	28.3	19.0
✓	✓		19.7	7.3	17.6	27.6	20.5
✓	✓	✓	**23.7**	**14.9**	**22.8**	28.6	**24.2**

Table 2: Effect of different components. *obj* for adding the category-agnostic task, *neg* for reweighing negative gradients, *pos* for reweighing positive gradients. Models are trained with random samplers by standard 1x schedule. AP and AP$_b$ denotes mask AP and box AP respectively.

4. Experiments

4.1. Dataset and Evaluation Metric

LVIS [12] is a new benchmark for long-tailed object recognition. It provides precise bounding box and mask annotations for various categories with long-tailed distribution. We mainly perform experiments on the recently released challenging LVIS v1.0 dataset. It consists of 1203 categories. We train our models on the `train` set, which contains about 100k images with 1.3M instances. In addition to widely-used metric AP across IoU threshold from 0.5 to 0.95, LVIS also reports AP$_\mathrm{r}$ (rare categories with 1-10 images), AP$_\mathrm{c}$ (common categories with 11-100 images), AP$_\mathrm{f}$ (frequent categories with $>$ 100 images). Since LVIS is federated dataset, categories are not annotated exhaustively. Each image have two more types of labels: `pos_category_ids` and `neg_category_ids`, indicating which categories are or are not present in that image. Detection results that do not belong to those categories will be ignored for that image. We report results on the `val` set of 20k images.

4.2. Implementation Details

We implement our method using MMDetection [5]. Models are trained using SGD with a momentum of 0.9 and a weight decay of 0.0001. The ResNet [15] backbones are initialized by ImageNet pre-trained models. Following the convention, scale jitter and horizontal flipping are used in training and no test time augmentation is used. We use a total batch size of 16 on 16 GPUs (1 image per GPU), and set the initial learning rate to 0.02. Following [12], the maximum number of detection per image are up to 300, and the minimum score threshold are reduced to 0.0001 from 0.01. For our method, we set $\gamma = 12$, $\mu = 0.8$ and $\alpha = 4$. More details about hyper-parameters are showcased in section 4.7. Since EQL v2 uses sigmoid loss function, we initialize the bias of the last fully-connected classification layer(fc-cls) with values of 0.001 to stabilize the training at the beginning.

Following [23], we also add a branch for detecting objectiveness instead of concrete category to reduce false-positives, which we refer as **category-agnostic task**. In the training phase, this task treats all other tasks' positive samples as its positive samples. In the inference phase, the estimated probability of other sub-tasks becomes:

$$p'_j = p_j * p_{obj} \tag{10}$$

Where p_{obj} is the probability for a proposal being a object. The proposed gradient guided reweighing are not applied on the category-agnostic task.

4.3. Ablation Studies

We take Mask R-CNN [14] equipped with ResNet-50 and FPN [24] as our baseline model. The effect of each component is shown in Table 2. The baseline model performs poorly on the tail classes, resulting in 0% and 12.0% AP for rare and common categories. And the performance

method	AP	AP$_r$	AP$_c$	AP$_f$
(a) Mask-R50	20.5	2.0	19.0	30.3
(b) +EQL v2	26.2	19.1	25.0	30.7
(c) Mask-R101	21.7	1.6	20.7	31.7
(d) +EQL v2	27.5	20.5	26.2	32.0

Table 3: Results of larger backbones with a longer 3x schedule. A Random sampler is used. The models are trained with totally 36 epochs, and the learning rate is divided by 10 at the 28th epoch and 34th epoch respectively.

gap between head and tail classes are very large. Adding a category-agnostic task helps all categories to some extent, improving the overall AP by 2.0% but not very much for the rare categories since the main problem for them is the unbalanced positive and negative gradients, *i.e*. their positive gradients are overwhelmed by negative gradients cause by a vast number of negative samples. By down-weighting the influence of negative gradients, their accuracy is boosted significantly (5.4% for rare categories). Up-weighting the positive gradient helps to achieve a more balanced ratio of positive to negative gradients. It brings a 7.6% performance boosting for rare categories. With these three components, we achieve a 23.7% AP, outperforming the baseline model 16.1% AP by a large margin without any re-sampling techniques. These ablation experiments verified the effectiveness of our proposed loss function.

4.4. Main Results

Comparison with Decoupled Training methods. We mainly compare our method with three decoupled training methods (cRT [18], LWS [18], and BAGS [23]). The results are present in Table 1. The decoupled training models (Table 1 (e) (f) (g)) are first initialized from naive softmax baseline (Table 1 (a)), then re-train their classifier layer (fc-cls) for another 12 epoch with other layers frozen, resulting in a total 24 epoch training. Those decoupled training methods all improve the AP, mainly for tail classes. The improvements brought by LWS is limited. We conjecture it is because that LWS only learns a scaling factor to adjust the decision boundary of the classifier but the classifier itself is not good and imbalanced. Our method achieves a overall 23.7% AP, increasing AP$_r$ by 14.9%, AP$_c$ by 10.8%. It is worth noting that EQL v2 does not require the extra fine-tuning stage, and the representation and classifier are learned jointly. More importantly, it has already surpassed the decoupled training methods.

Comparison with End-to-End Training methods. Table 1 compares EQL v2 with two popular end-to-end training methods, Repeat Factor Sampling [12] (re-sampling) and Equalization Loss [36] (re-weighting). With a random sampler, our method outperforms naive softmax and EQL by a large margin, 7.6% and 5.1% respectively. Note that RFS repeats images that contains tail categories in each epoch, so it increases the total training time. Instead, our method only uses a random sampler and does not increase the training time, and achieves better results, 23.7% *vs*. 22.2%.

Larger Model & Longer Training. To verify the generalization ability across different backbones and training schedule. We conduct experiments with larger models by a 3x schedule. The results are present in Table 3. Note that training Mask R-CNN with longer schedule does not help rare categories a lot (Table 1 (a) *vs*. Table 3 (a)), the AP of rare categories is still bad because rare categories are heavily suppressed by the negative gradients caused by the entanglement of instances and tasks. In contrast, with the proposed EQL v2, the performance of rare categories can be further improved from 14.9% to 19.1% (Table 1 (h) and Table 3 (b)). When using large ResNet-101 backbone, we do not observe the over-fitting of tail classes in such a long schedule, and the gap between Mask R-CNN and EQL v2 holds.

4.5. Comparison with State-of-the-Art Methods

In this section, we compare our method with other work that report state-of-the-art results on LVIS v0.5 and LVIS v1.0. The results on LVIS v0.5 is present in Table 4, including RFS [12] for re-sampling, EQL [36] for re-weighing, LST [16] for incremental learning, SimCal [39] and BAGS [23] for decoupled training, Forest R-CNN [41] for hierachy classification, De-cofound-TDE [37] for causal inference. EQL v2 achieves better results than all those methods. With ResNet-50-FPN backbone, it outperforms the winner of last year's challenge EQL by 4.3%. We also compare the results under large models. With the same Cascade Mask R-CNN [1] framework equipped with ResNet-101-FPN, EQL v2 still has a 1.8% higher AP than De-confound-TDE. With the same Hybrid Cascade R-CNN [4] framework with ResNeXt-64x4d-FPN [42], EQL v2 outperform BAGS by 0.8% AP. Since LVIS v1.0 is recently proposed, not much work has reported their results on it. We mainly compare EQL v2 with De-confound-TDE. In addition, we also re-implement the equalization loss. The original EQL chooses a hyper-parameter λ of 1.76×10^{-3} for LVIS v0.5, we found this is not optimal for LVIS v1.0, so we tune this hyper-parameter and report the results with best value of $\lambda = 1.1 \times 10^{-3}$. The results are shown in Table 5. Our method achieves higher overall AP across different backbones and frameworks. EQL v2 outperforms the De-confound-TDE by 6.3% for rare categories.

4.6. Model Analysis

Do we have a more balanced gradient ratio? We visualize the gradient ratio of our method (Table 1 (h)) and base-

method	framework	#sampler	AP	AP_r	AP_c	AP_f	AP_b
RFS† [12]	Mask-R50	repeat	24.4	14.5	24.3	28.4	-
EQL† [36]	Mask-R50	random	22.8	11.3	24.7	25.1	23.3
LST† [16]	Mask-R50	-	23.0	-	-	-	-
SimCal† [39]	Mask-R50	random/balance	23.4	16.4	22.5	27.2	-
Forest R-CNN† [41]	Mask-R50	nms-resample	25.6	18.3	26.4	27.6	25.9
BAGS† [23]	Mask-R50	random/random	26.3	18.0	26.9	28.7	25.8
De-confound-TDE† [37]	Cascade-R101	random	28.4	22.1	29.0	30.3	31.0
BAGS† [23]	HTC-X101	random/random	31.2	23.4	32.3	32.9	33.7
EQL v2 (Ours)	Mask-R50	random	27.1	18.6	27.6	29.9	27.0
EQL v2 (Ours)	Mask-R101	random	28.1	20.7	28.3	30.9	28.1
EQL v2 (Ours)	Cascade-R101	random	30.2	23.0	30.9	32.1	33.0
EQL v2 (Ours)	HTC-X101	random	**32.0**	**24.2**	**32.8**	**34.1**	**34.0**

Table 4: Comparison with state-of-the-art methods on **LVIS v0.5** `val` set. † indicates that the reported result is directly copied from referenced paper. 'Mask-R50' indicates Mask R-CNN [14] with ResNet50-FPN [15, 24], 'Cascade' is for Cascade Mask R-CNN [1], 'HTC' is for Hybrid Task Cascade [4]. Models are trained with the corresponding **LVIS v0.5** `train` set.

method	framework	AP	AP_r	AP_c	AP_f	AP_b
De-confound† [37]	Cascade-R101	23.5	5.2	22.7	32.3	25.8
De-confound-TDE† [37]	Cascade-R101	27.1	16.0	26.9	32.1	30.0
Mask R-CNN	Mask-R50	19.2	0	17.2	29.5	20.0
EQL*	Mask-R50	21.6	3.8	21.7	29.2	22.5
EQL v2 (Ours)	Mask-R50	25.5	17.7	24.3	30.2	26.1
Mask R-CNN	Mask-R101	20.8	1.4	19.4	30.9	21.7
EQL*	Mask-R101	22.9	3.7	23.6	30.7	24.2
EQL v2 (Ours)	Mask-R101	27.2	20.6	25.9	31.4	27.9
Cascade Mask R-CNN	Cascade-R101	22.6	2.0	22.0	32.5	25.2
EQL*	Cascade-R101	24.5	4.1	25.8	32.0	27.2
EQL v2 (Ours)	Cascade-R101	**28.8**	**22.3**	**27.8**	**32.8**	**32.3**

Table 5: Comparison with state-of-the-art methods on **LVIS v1.0** `val` set. † indicates that the reported result is directly copied from the referenced papers. * indicates our re-implementation. Models are trained using **LVIS v1.0** `train` set. We train our models with a standard 2x schedule.

line model (Table 1 (b)) during the training process, see Figure 2. The baseline model does not have a balanced ratio for all categories. The positive gradients are overwhelmed by the negative gradients, especially for tail classes, which makes it hard to detect them. And training longer does not help a lot. In contrast, EQL v2 preserves a more balanced gradient ratio in the entire training phase.

Whether the classifiers are balanced? Decoupled training methods [18, 23] have shown that if models are trained with long-tailed distributed data, the weight norm in the last classifier layer (fc-cls) is heavily biased. Those methods re-balance the classifier at the second fine-tuning stage, resulting in balanced weight norms. We also visualize the weight norm of the fc-cls of three models: baseline, RFS and EQL v2(Table 1 (a), (c) and (h)), in Figure 3. The model trained with repeat factor sampling still suffers from biased weight norm. On the contrary, the model trained with EQL v2 has a more balanced weight norm.

Do we have a better representation? To evaluate the quality of representations trained with our method. We adopt models trained with our method and standard training as pre-trained model. Then we follow the classic decoupled training recipe to re-train the classifier with frozen representations. The results are shown in Table 6. There are two main observations: Firstly, models initialized with EQL v2 always achieve a higher AP, resulting in 22.4 *vs.* 22.0 for cRT, 23.1 *vs.* 17.0 for LWS, 24.0 *vs.* 23.1 for BAGS. It verifies that we obtain a better representation by adopting EQL v2 compared to standard training. This result doubts the claim [18] that re-weighing will hurt the representation. Secondly, the models get marginal improvements or even worse performance after decoupled training. The AP only

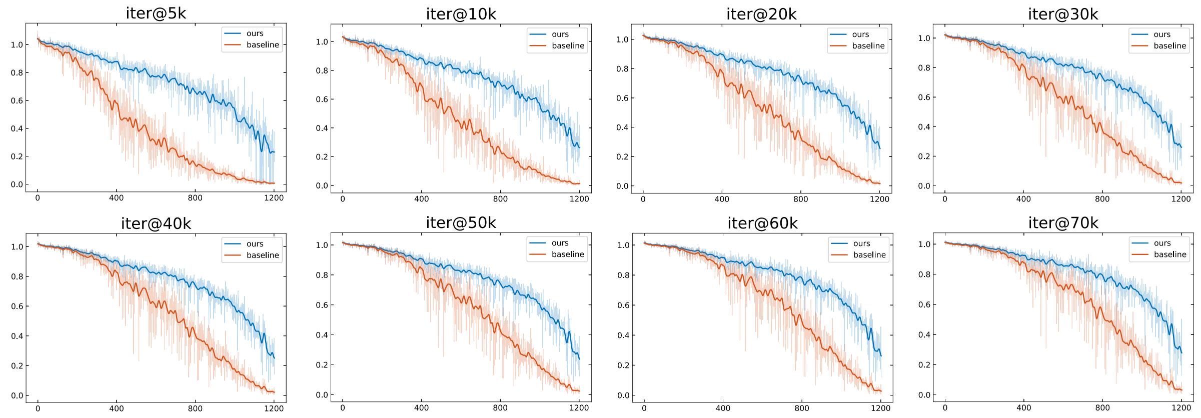


Figure 2: The accumulated gradients ratio of positives to negatives. Models are trained with total 75k iterations. We show the values at different training iteration. We compare the accumulated gradients of two models, Mask R-CNN with sigmoid loss and EQL v2.

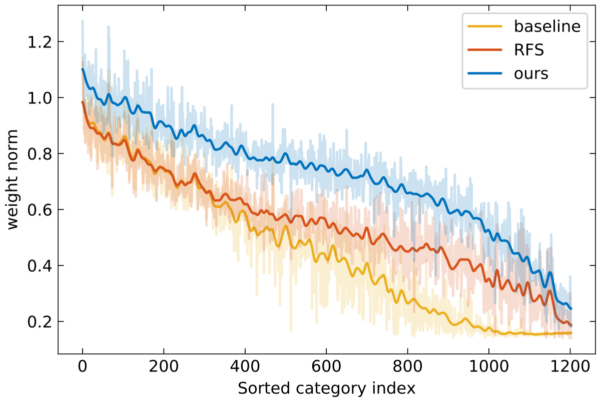


Figure 3: The L2 weight norm of the fc-cls layer of models.

increase 0.3 % after using BAGS re-training, compared to the 23.7% AP of EQL v2 (Table 1(h)), and AP drops 1.3 % and 0.6 % after using cRT and LWS respectively. It shows that decoupled training is not always necessary, we can train models with both a balanced classifier and better representations in an end-to-end manner.

4.7. Influence of hyper-parameters

In Table 7 we investigate how the shape of mapping function $f(\cdot)$ in equation 6 affects the training. There are two hyper-parameters in the function, γ and μ. We can see that the detection result is not sensitive to the shape of the mapping function, and the AP increases stably when those two hyper-parameters move in a wide range. The visualization of mapping functions is shown in Figure 4. The effect of α is present in Table 8.

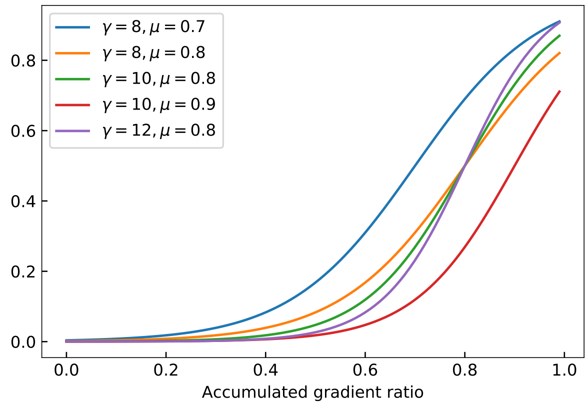


Figure 4: Mapping functions with different μ and γ

method	EQL v2	AP	AP$_r$	AP$_c$	AP$_f$	AP$_b$
cRT		22.0	13.5	20.8	27.1	22.1
cRT	✓	22.4	13.3	21.7	27.3	22.4
LWS		17.0	2.0	13.5	27.4	17.5
LWS	✓	23.1	13.8	22.2	28.1	23.2
BAGS		23.1	13.1	22.5	28.2	23.7
BAGS	✓	24.0	14.6	23.8	28.5	24.5

Table 6: Results of various decoupled training methods with different pre-trained models. EQL v2 indicates the pre-trained models are trained with EQL v2, otherwise with standard training. Only random samplers are used.

4.8. C Sigmoid *vs*. 2C Softmax

The idea of balancing gradient in EQL v2 is not restricted to sigmoid classifier. Recall that sigmoid uses a single output logit to represent each task and use the function σ to

γ	μ	AP	AP$_r$	AP$_c$	AP$_f$	AP$_b$
8	0.7	23.1	12.2	22.3	28.6	23.7
8	0.8	23.6	13.9	22.7	**28.8**	24.1
10	0.8	23.6	13.8	**22.9**	28.7	**24.3**
10	0.9	23.5	14.0	22.8	28.5	24.0
12	0.8	**23.7**	14.9	22.8	28.6	24.2
12	0.9	23.6	**15.5**	22.6	28.2	24.0

Table 7: Varying γ and μ for mapping function. The positive up-weighting parameter α is set to 4.

α	AP	AP$_r$	AP$_c$	AP$_f$	AP$_b$
0	19.7	7.3	17.6	27.6	20.5
1	21.9	11.3	20.7	28.0	22.8
2	23.0	13.5	22.0	28.4	23.7
4	23.7	14.9	**22.8**	**28.6**	24.2
8	**24.0**	**16.5**	22.7	28.6	**24.4**

Table 8: Varying α. γ and μ is set to 12 and 0.8 respectively.

estimate the probability. We show another choice of the classifier: 2C Softmax which uses 2 output logits for each task and adopts softmax function to estimate the probability. The extra output logit for each task can be regarded as a concept of *others category* and introduces competition when doing inference so it helps reducing false-positives. In Table 9, we compare the results of C-sigmoid and 2C-softmax under different settings. When only adding the objectiveness task and down-weighting the negative gradients, 2C-softmax achieves higher accuracy than C-sigmoid. These two designs reach comparable results after up-weighing the positive gradients.

4.9. Experiments on Open Images Detection

To verify the generalization ability to other datasets, we conduct experiments on the OpenImages [20]. OpenImages is another large-scale object detection dataset with long-tailed distributed categories. We use the data split of challenge 2019, which is a subset of OpenImages V5. The `train` set consists of 1.7M images of 500 categories. We evaluate our models on the 41k `val` set. In addition to the standard mAP@IOU=0.5 metric, we also group categories into five groups (100 categories per group) according to their instance numbers and report the mAP within each group respectively. The results are shown in Table 10. EQL v2 reaches an AP of 52.6, outperforming the baseline model and EQL by 9.5 AP and 7.3 AP respectively. For the tail group (AP1), the EQL v2 increases the AP by 22.3 point, which is much more than the improvement of EQL (6.4 AP). EQL v2 also outperform EQL considerably on the larger ResNet-101 backbone. For both baseline and EQL models, there is still a large performance gap between head and tail classes. EQL v2 brings all categories into a more equal status. It achieves similar accuracy for all categories groups. It is worth noting that we tune the hyper-parameter λ in EQL which puts 250 categories into tail group for OpenImage. In contrast, the hyper-parameters of EQL v2 are kept the same as that on LVIS. Those experiments not only show the effectiveness but also good generalization ability of EQL v2. We also report the further tuned results of EQL v2 on Open Images. The values of μ, γ and α are 0.9, 12, 8 respectively.

	obj?	neg?	pos?	AP	AP$_r$	AP$_c$	AP$_f$
C-sigmoid	✓			18.1	1.9	16.4	28.3
2C-softmax				**19.0**	**2.0**	**17.3**	**28.4**
C-sigmoid	✓	✓		19.7	7.3	17.6	27.6
2C-softmax				**20.7**	**9.5**	**18.9**	**27.7**
C-sigmoid	✓	✓	✓	23.7	**14.9**	**22.8**	28.6
2C-softmax				**23.7**	14.9	22.7	**28.7**

Table 9: Comparison between C-sigmoid and 2C-softmax under different components.

	AP	AP1	AP2	AP3	AP4	AP5
Faster-R50	43.1	26.3	42.5	45.2	48.2	52.6
EQL	45.3	32.7	44.6	47.3	48.3	53.1
EQL v2†	52.6	48.6	52.0	53.0	53.4	55.8
EQL v2‡	**53.8**	**49.6**	**53.3**	**54.5**	**54.9**	**56.6**
Faster-R101	46.0	29.2	45.5	49.3	50.9	54.7
EQL	48.0	36.1	47.2	50.5	51.0	55.0
EQL v2†	55.1	51.0	55.2	56.6	55.6	57.5
EQL v2‡	**55.6**	**51.5**	**55.5**	**57.5**	**55.8**	**57.6**

Table 10: Results on **Open Images Challenge 2019** `val` set. The model Faster R-CNN [31] with ResNet-FPN is trained with a schedule of 120k/160k/180k. Categories are grouped into five groups according to the instance number. AP1 is the AP of the first group, where categories have least annotations, AP5 is the AP of the last group, where categories have most annotations. †means that we directly use the hyper-parameters searched in LVIS, ‡means that we tune them in OpenImages.

5. Conclusion

In this work, we propose the key of improving performance for long-tailed object detection is to maintain balanced gradients between positives and negatives. An improved version of EQL, EQL v2 is then proposed to dynamically balance the gradient ratio between positives to negatives in the training phase. It brings large improvements with notably boosting on tail categories across various frameworks. As an end-to-end training method, it beats all existing methods on the challenging LVIS benchmark, including the dominant decoupled training schema.

References

[1] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162, 2018. 1, 2, 5, 6

[2] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. In *Advances in Neural Information Processing Systems*, pages 1567–1578, 2019. 2

[3] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002. 2

[4] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4974–4983, 2019. 2, 5, 6

[5] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, et al. Mmdetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019. 4

[6] Peng Chu, Xiao Bian, Shaopeng Liu, and Haibin Ling. Feature space augmentation for long-tailed data. *arXiv preprint arXiv:2008.03673*, 2020. 2

[7] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9268–9277, 2019. 2

[8] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. Centernet: Keypoint triplets for object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6569–6578, 2019. 2

[9] 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(2):303–338, 2010. 1

[10] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. 2

[11] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014. 2

[12] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5356–5364, 2019. 2, 4, 5, 6

[13] Hui Han, Wen-Yuan Wang, and Bing-Huan Mao. Borderline-smote: a new over-sampling method in imbalanced data sets learning. In *International conference on intelligent computing*, pages 878–887. Springer, 2005. 2

[14] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 2, 4, 6

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2, 4, 6

[16] Xinting Hu, Yi Jiang, Kaihua Tang, Jingyuan Chen, Chunyan Miao, and Hanwang Zhang. Learning to segment the tail. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14045–14054, 2020. 2, 5, 6

[17] Chen Huang, Yining Li, Chen Change Loy, and Xiaoou Tang. Learning deep representation for imbalanced classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5375–5384, 2016. 2

[18] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. *arXiv preprint arXiv:1910.09217*, 2019. 1, 2, 4, 5, 6

[19] Tao Kong, Fuchun Sun, Huaping Liu, Yuning Jiang, Lei Li, and Jianbo Shi. Foveabox: Beyound anchor-based object detection. *IEEE Transactions on Image Processing*, 29:7389–7398, 2020. 2

[20] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Tom Duerig, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *arXiv preprint arXiv:1811.00982*, 2018. 2, 8

[21] Hei Law and Jia Deng. Cornernet: Detecting objects as paired keypoints. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 734–750, 2018. 2

[22] Buyu Li, Yu Liu, and Xiaogang Wang. Gradient harmonized single-stage detector. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8577–8584, 2019. 2

[23] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10991–11000, 2020. 1, 2, 3, 4, 5, 6

[24] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 1, 2, 4, 6

[25] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 1, 2

[26] 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

European conference on computer vision, pages 740–755. Springer, 2014. 1

[27] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016. 1

[28] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2537–2546, 2019. 2

[29] Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten. Exploring the limits of weakly supervised pretraining. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 181–196, 2018. 2

[30] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 1

[31] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 1, 2, 8

[32] Li Shen, Zhouchen Lin, and Qingming Huang. Relay backpropagation for effective learning of deep convolutional neural networks. In *European conference on computer vision*, pages 467–482. Springer, 2016. 2

[33] Jun Shu, Qi Xie, Lixuan Yi, Qian Zhao, Sanping Zhou, Zongben Xu, and Deyu Meng. Meta-weight-net: Learning an explicit mapping for sample weighting. *arXiv preprint arXiv:1902.07379*, 2019. 2

[34] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2

[35] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 2

[36] Jingru Tan, Changbao Wang, Buyu Li, Quanquan Li, Wanli Ouyang, Changqing Yin, and Junjie Yan. Equalization loss for long-tailed object recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11662–11671, 2020. 2, 4, 5, 6

[37] Kaihua Tang, Jianqiang Huang, and Hanwang Zhang. Long-tailed classification by keeping the good and removing the bad momentum causal effect. *arXiv preprint arXiv:2009.12991*, 2020. 3, 5, 6

[38] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 9627–9636, 2019. 2

[39] 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. *arXiv preprint arXiv:2007.11978*, 2020. 1, 3, 5, 6

[40] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Learning to model the tail. In *Advances in Neural Information Processing Systems*, pages 7029–7039, 2017. 2

[41] Jialian Wu, Liangchen Song, Tiancai Wang, Qian Zhang, and Junsong Yuan. Forest r-cnn: Large-vocabulary long-tailed object detection and instance segmentation. *arXiv preprint arXiv:2008.05676*, 2020. 5, 6

[42] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017. 5

[43] Xi Yin, Xiang Yu, Kihyuk Sohn, Xiaoming Liu, and Manmohan Chandraker. Feature transfer learning for face recognition with under-represented data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5704–5713, 2019. 2

[44] Xiao Zhang, Zhiyuan Fang, Yandong Wen, Zhifeng Li, and Yu Qiao. Range loss for deep face recognition with long-tailed training data. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5409–5418, 2017. 2

[45] Xingyi Zhou, Jiacheng Zhuo, and Philipp Krahenbuhl. Bottom-up object detection by grouping extreme and center points. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 850–859, 2019. 2

[46] Chenchen Zhu, Yihui He, and Marios Savvides. Feature selective anchor-free module for single-shot object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 840–849, 2019. 2