# Towards Visual Recognition in the Wild

Jiarui Cai

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2022

Reading Committee:

Jenq-Neng Hwang, Chair

Linda Shapiro

Radha Poovendran

Program Authorized to Offer Degree:
UW Electrical and Computer Engineering

University of Washington

**Abstract**

Towards Visual Recognition in the Wild

Jiarui Cai

Chair of the Supervisory Committee:
Dr. Jenq-Neng Hwang
Electrical and Computer Engineering

The predefined artificially-balanced training classes in object recognition have limited capability in modeling real-world scenarios where objects are imbalanced-distributed with unknown classes. In this thesis, we present three research works on Long-Tailed Recognition task for both closed-set and open-set scenarios.

For closed-set long-tailed recognition, existing one-stage methods improve the overall performance in a "seesaw" manner, i.e., either sacrifice the head's accuracy for better tail classification or elevate the head's accuracy even higher but ignore the tail. Other algorithms bypass such trade-off by a multi-stage training process: pre-training on imbalanced set and fine-tuning on balanced set. Though achieving promising performance, not only are they sensitive to the generalizability of the pre-trained model, but also not easily integrated into other computer vision tasks like detection and segmentation, where pre-training of classifier solely is not applicable. In this thesis, we introduce a one-stage long-tailed recognition scheme, ally complementary experts (ACE), where the expert is the most knowledgeable specialist in a sub-set that dominates its training, and is complementary to other experts in the less-seen categories without disturbed by what it has never seen. We design a distribution-adaptive optimizer to adjust the learning pace of each expert to avoid over-fitting. Without special bells and whistles, the vanilla ACE outperforms the current one-stage SOTA method by 3$\sim$ 10% on CIFAR10-LT, CIFAR100-LT, ImageNet-LT and iNaturalist datasets. It is also shown to be the first one to break the "seesaw" trade-off by improving the accuracy of the majority and minority categories simultaneously in only one stage.

For open-set long-tailed recognition, firstly, we propose a distribution-sensitive loss, which weighs more on the tail classes to decrease the intra-class distance in the feature space. Building upon these concentrated feature clusters, a local-density-based metric is introduced, called Localizing Unfamiliarity Near Acquaintance (LUNA), to measure the novelty of a testing sample. LUNA is flexible with different cluster sizes and is reliable on the cluster boundary by considering neighbors of different properties. Moreover, contrary to most of the existing works that alleviate the open-set detection as a simple binary decision, LUNA is a quantitative measurement with interpretable meanings. Our proposed method exceeds the state-of-the-art algorithm by 4-6% in the closed-set recognition accuracy and 4% in F-measure under the open-set on the public benchmark datasets, including our own newly introduced fine-grained OLTR dataset about marine species (MS-LT), which is the first naturally-distributed OLTR dataset revealing the genuine genetic relationships of the classes.

LUNA is a step closer to the real-world open-set long-tailed recognition problem, however, we see two deficiencies: technically, it suffers from a trade-off between representation learning and classifier training; in terms of result interpretation, the semantic meaning of the learned features are unexplored. Therefore, we present an improved LUNA framework, called LUNA+. In LUNA+, the feature learning and classifier learning are decoupled with an extra feature projection module. In addition, cluster centers are pre-optimized to be uniformly distribution in latent space to eliminate bias. LUNA+ further improves the OLTR performance, and enables more automated, robust and scalable applications.

# TABLE OF CONTENTS

## LIST OF FIGURES

## LIST OF TABLES

## ACKNOWLEDGMENTS

First of all, I would like to express my deepest appreciation to my advisor, Professor Jenq-Neng Hwang, for his support, encouragement, and patience. He always conveys a spirit of modesty and positivity regarding research and life. His supervision helps me grow in many ways. This dissertation could not have been possible without his guidance and persistent help. Additionally, I sincerely thank and acknowledge my Ph.D. committee members, Professor Radha Poovendran, Professor Linda Shapiro, and Professor Simon Shaolei Du, for their invaluable inputs, suggestions, and encouragement. Their insightful questions in the general exam inspired me to improve the proposal and eventually led to the thesis. I am also grateful to have Professor Blake Hannaford, Professor Ming-Ting Sun, and Professor Eli Shlizerman for serving on the qualifying exam committee.

Special thanks are given to my former advisor Professor Chun Yuan and members of Vascular Imaging Lab: Dr. Tom Hatsukami, Dr. Niranjan Balu, Dr. Gador Canton, Dr. Paul Chu, Dr. Jie Sun, Dr. Hiroko Watase, Dr. Duygu Baylam Geleri, Dr. Zechen Zhou, Dr. Wenjin Liu, Dr. Zhensen Chen, Tong Zhu, Daniel Hippe, Zach Miller and Kristi Pimentel. Thank you for the opportunity of studying at VIL and for all the support. Your patience and kind help meant a lot to me.

To my kindest current and former collaborators in NOAA, Craig Rose, Suzanne Romain, Kelsey Magrane, Paul Packer, Graeme LeeSon, Andy Kingham, Keith Fuller, Brandon Moore, Farron Wallace and Farron's team: thank you for always being so supportive and patience during our collaboration. I learned a lot from your expertise and your attitude towards research. I sincerely hope the collaboration between NOAA and UW can continue well in the future and benefit more significant applications.

To my friends in the Information Processing Laboratory, Yizhou Wang, Haotian Zhang, Hung-Min Hsu, Renshu Gu, Gaoang Wang, Thomas Tang, Tsung-Wei Hwang, Ying Jin, Li Chen, Zhongyu Jiang, Andy Cheng, Jie Mei, Haorui Ji, Aotian Zheng, Chris Yang, Yu-Hsuan Li, Fangyi Zhu,

Yanting Zhang, Professor Shan Gao: I would like to thank you all for the great moments we had in and outside the lab. I cannot survive the pressures without your support and I feel so blessed to share my happiness with friends like you.

Finally, to my parents, parents-in-law and my other families: I have been blessed with a loving family. Thank you for being with me through every step of the long journey and for being my most enthusiastic cheerleader. This thesis is dedicated to you.

Lastly, to my dearest husband and best friend Hao: there are countless moments I doubt myself; you are always the one believing in me, patiently dealing with all my emotions and taking care of me. Your unconditional love, trust, understanding, respect and support shape me to a better person. You are an amazing husband, thank you for everything.

# Chapter 1

# INTRODUCTION

## 1.1 Background and Motivations

There is a wide range of real-life object recognition applications that operates under the training-learning paradigm and can be naturally modeled as the image recognition task, which is one of the most fundamental and substantial studies in computer vision. Technical revolution sweeps various fields like species identification [98], medical imaging perception [106], human face recognition [21] and scene classification in autonomous driving [75]. However, in real applications, the performances of the off-the-shelf object recognition methods mostly bias on the sample-rich classes that have been seen in the training set, with a limited ability on classifying the sample-few classes, not to mention the novel classes of objects [47, 127].

The main culprit of this phenomenon is the simulation scenario in the academia cannot fully model reality: the conformity between training and testing sets determines the system's dynamic performance, reliability and scalability. Back in the real world, the factual object samples are unevenly distributed, and the object classes are always open-ended. Specifically, the frequency distribution of visual objects is long-tailed [84], that most samples belong to very few classes (called head or majority classes) while most classes (called tail or minority classes) only count for a small portion of samples. To achieve the goal of visual recognition in the wild, the tasks include classifying the closed-set categories with imbalanced training data and identifying the novel classes during inference.

We compared concepts in Table 1.1 to further clarify the terminologies and demonstrate the challenges. From small-scale [54] to large-scale setting [55], conventional object recognition are widely studied. A dataset is *imbalanced* if the the number of samples in the largest classes is over 10 times to the smallest class. *Long-tailed* is the extreme situation of data imbalance, where that smallest class only has less than 20 training samples. Learned models are easily biased towards the majority classes and thus perform poorly on the minority classes due to the disparity of data

| Task | Balanced Training Set? | Size of Tail (min img/class) | Balanced Testing Set? | Novel Classes in Testing? | Example Benchmarks |
|---|---|---|---|---|---|
| Object Recognition | ✓ | - | ✓ | ✗ | ImageNet [55]<br>CIFAR [54] |
| Imbalanced Object Recognition | ✗ | $> 20$ | ✓ | ✗ | MIT-67 [82]<br>Caltech-101 [24] |
| Long-tailed Object Recognition | ✗ | $< 20$ | ✓ | ✗ | ImageNet-LT [70]<br>CIFAR-LT [11] |
| Open-set Object Recognition | ✓ | - | ✓ | ✓ | Letter [26]<br>MNIST [59] † |
| Open-set Long-tailed Object Recognition | ✗ | $< 20$ | ✓ | ✓ | ImageNet-LT [70]<br>MarineSpecies-LT [9] ‡ |

Table 1.1: Comparison between visual recognition tasks. Balanced set means samples per class are evenly distributed. †: open-set object recognition datasets are usually stimulated from balanced object recognition datasets by by defining closed and open classes. ‡: the closed testing set of open-set long-tailed recognition dataset can be used for closed-set long-tailed recognition task.

distribution. For fair comparison over the entire dataset, balanced testing set are purposely adopted for all recognition benchmark. Statistically, the expectation of accuracy of the unbiased model is the highest for a unknown testing distribution. Besides, the same categories in training/testing splits are called *closed-set* and the novel categories in testing are called *open-set*. *Open-set object recognition* aims to detect samples from the untrained classes and reject their closed-set classification results, meanwhile, the closed-set accuracy should be preserved.

With the emerging need of bridging the gap between the practical applications and the academic models, an appealing trend is to study the networks that are trained with long-tailed-distributed data and tested with both trained and novel classes, called opened-set long-tailed recognition (OLTR) [70]. In summary, the two challenges in *open-set long-tailed recognition* lie in

- The balancing between representation learning and classification. Prior statistic-based [23, 117, 5] or feature distance-based [88, 115, 35] methods suggest the promising potentials of detecting anomalies features from a well-modeled balanced feature space. However, the OLTR suffers from head-tail performance trade-off [10, 107]. That says re-balancing techniques [41, 127] improve the performance of tail classes, but hurt the feature learning over-

all [47]. The under-representation of the dataset causes difficulties in identifying anomalies. On the other hand, a biased model fits the majority data helps novelty detection but results in poor classification performance.

- Lack of quantitative measurement or semantic interpretation of novelty. Recent OLTR approaches [70, 130] use thresholding on feature distance to identify novel data. The level of novelty and the semantic meanings of the distance are not explored. However, they are useful clues for manual annotation and analysis after the outliers are identified, thus worth studying.

Motivated by the above challenges, in this thesis, we present three research works to tackle the model biases caused by long-tailed training data and novelty detection issue due to open-bounded testing data.

## 1.2 Overview of the Proposed Methods

### 1.2.1 Closed-set Long-tailed Recognition

Object recognition researches are mostly driven by artificially-balanced benchmarks, e.g., ImageNet [20] and CIFAR [54], current models tend to be dominated by the heads [47, 127]. Facing up to the reality, scarce as the tail categories are, they are of the same or even higher significance than the heads in various fields, such as biological species identification [98], disease classification [99] and web-spam message detection [124]. This long-lasting bottleneck significantly restricted classification related computer vision tasks into practical use, including detection [94, 111, 118] and instance segmentation [102, 119].

To ensure a recognition capability over all categories, a tail-sensitive classifier becomes necessary. Existing solutions fall in three categories: one-stage [41, 108], two-stage with pre-training[47, 11], and multi-stage multi-expert frameworks [113, 107]. The one-stage algorithms follow a straightforward idea to addressed the imbalance of training set by re-balancing, including re-sampling [47] and re-weighting [11, 17, 121]. Despite the promotion of the tails, balancing techniques show an obvious "seesaw" phenomenon (Figure 1.1), that the accuracy of majority classes is sacrificed, indicating the under-representation of the heads. This raises a new concern that reducing the heads' accuracy might lead to more serious consequences. Taking the animal identification system as an

example, some species are much richer in population than the endangered ones. Increasing the recognition accuracy of the snow leopards has little chance to be verified as they are rarely seen; on the contrary, failing to precisely classify two bird kinds can easily result in a misunderstanding of the local ecology.

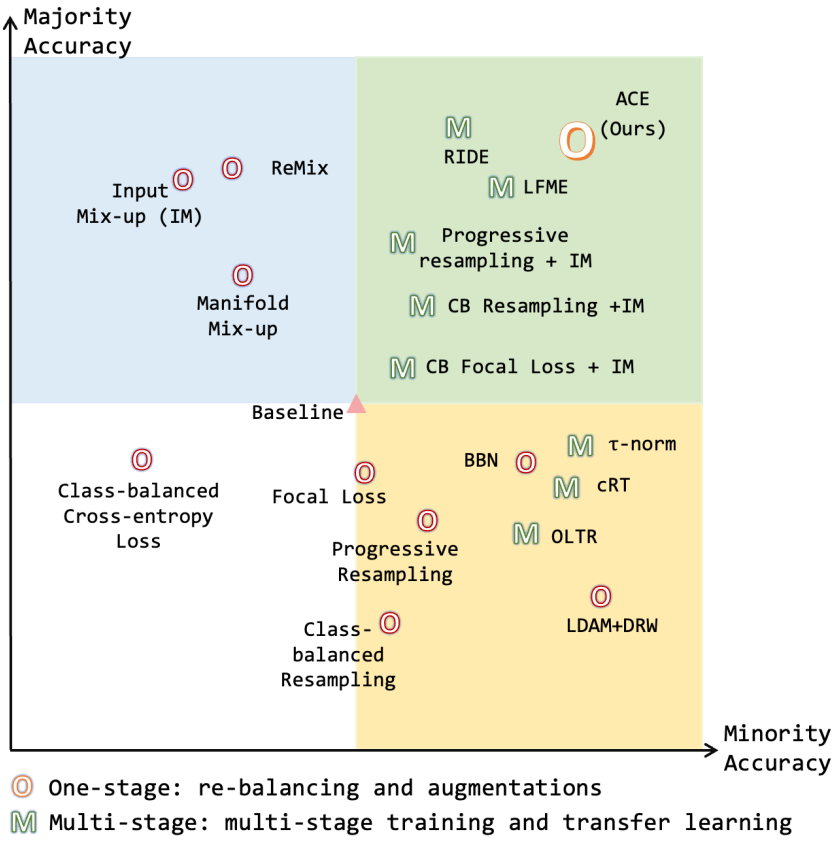



Figure 1.1: Performance of representative long-tailed recognition methods in terms of majority and minority classes compared to the baseline model. The result indicates most re-balancing methods improve the performance of minority categories by sacrificing that of the majority even with two-stage training (quadrant IV). Data augmentations are effective on the heads but slightly hurt the tails (quadrant II).

Literature in the recent two years [47, 107, 113, 123] handles the issue in a roundabout way: firstly train the feature extractor (backbone) with the whole imbalanced set for generalizable representation learning, then re-adjust the classifier by re-sampled data or build diverse experts for

various tasks in cascading stages. Further improving the performance as they are, however, the general idea still holds old wine in a new bottle by making new trade-offs. To re-balance the data distribution, heavily relying on the well-adjusted pre-trained model and re-balancing skills make the frameworks sensitive to hyper-parameters and hard to find a sweet point. More importantly, the accumulated training steps make the multi-stage models redundant and less practical to be integrated with other tasks simultaneously, e.g., detection [102] and segmentation [119]. To guarantee the plug-in and play property, it is thus highly desirable to have a classifier that shot the long-tail challenge with only one stage.

The hankerings of current long-tail challenges make us look more profound to the human intelligence. When human-beings make hard classification choices, saying diagnosis of diseases, it is advantageous to involve specialists' insights who are well-aware of their own fields. Moreover, for the rare diseases, panel discussion and consultation are indispensable to exclude interfering potentials. Similar in the long-tailed issue, we are inspired to design a group of experts with *complementary* skills: (1) they share elementary knowledge from the most diverse data source; (2) they are professional at splits of data respectively, and aware of what they do not specialize in; (3) opinions from the experienced experts (who see more data) complement the judgment from junior experts (who see less) for optimal decision.

Following the idea, we propose the Ally Complementary Experts (ACE) for one-stage long-tailed recognition. ACE is a multi-expert structure where experts are training in parallel with a shared backbone. The experts are assigned with diverse but overlapping imbalanced subsets, to benefit from specialization in the dominating part. We also introduce a distribution-adaptive optimizer that controls the update of each expert according to the volume of its training set. Finally, the output makes the best out of all experts learned by the designed complementary loss. ACE is trained end-to-end without any pre-training.

We evaluate ACE on various widely-used long-tailed datasets, including CIFAR-10-LT, CIFAR-100-LT [17], ImageNet-LT [70] and iNaturalist2018 [98] extensively with various experimental settings. Our method becomes the new SOTA among all one-stage long-tailed recognition methods with by 3-10% accuracy gain and is the first one that improves performance on all the three frequency groups. ACE also surpasses several multi-stage methods [47, 51, 70, 113] by a large margin.

### 1.2.2 Open-set Long-tailed Recognition

The state-of-the-art algorithms are either focused on solving open-set issues [5, 89, 28], or only aimed to classify the objects under a closed long-tail distribution [47, 127]. However, when there is a call for implementing object recognition in daily life, the open-set and long-tail challenges commonly coincide [98, 109, 1]. Separating them is twice the effort for half the result. To step closer to reality, Liu et al. attempt to merge and deal with the open-set long-tail recognition (OLTR) together by one framework in 2019 and proposed the OLTR baseline [70]. However, existing OLTR approaches [70, 130] still see some fundamental and methodological gaps:

- No authentic collected open long-tail dataset to evaluate the OLTR methodology [70]: existing benchmarks are limited to artificially-sampled ones. The generic relationship among objects are disrupted. For example, there are only 9 samples for truck while 516 samples for white shark in ImageNet-LT.

- Decoupling open set challenges with long-tailed distribution [5, 28, 45, 36]: when people study the open set issues, models are designed based on balanced sets [20, 126, 56], which reduce the utility and value of transferring the methodology to OLTR tasks.

- Exhaustively engage into the long-tail recognition and ignore the open set issues [11, 41, 64, 18, 62]: due to the hurdle of recognition of objects from the imbalanced set, literature focuses on improving the accuracy in the long-tailed distribution without considering the open set scenario, which needs accommodations for the actual OLTR tasks.

The above three significant hurdles motivate us to research in the metric domain, which automates the feature selection and learns task-specific distance functions to access similarity [58]. In this proposal, we propose a metric learning framework, called Localizing Unfamiliarity Near Acquaintance (LUNA), to quantitatively measure the level of novelty based on the local density of the deep CNN features for the open-set long-tailed recognition task. With LUNA, two questions can be answered precisely, (1) whether the input is novel or not; (2) if no, which class it is; if yes, what is the unfamiliarity level of the new class concerning the pretrained acquaintance classes. In summary, we claim our contributions and technical innovations as follows,

- We collect a new well-annotated real Marine Species open long-tailed (MS-LT) dataset. As the first natural OLTR dataset in a fine-grained domain, it will be a solid supplement to the existing manually re-sampled OLTR datasets. It poses new challenges on representation learning and novel species detection.

- To make the categories concentrated in feature space individually, the feature extractor is trained by a newly proposed loss, called weighted center loss, to minimize their intra-class distances so as to form dense clusters in the high-dimensional space. It centralizes the deep features of the head classes, while preserving the classification accuracy of the tails, resulting in more distinctive features.

- To measure the unfamiliarity level of the new class and evaluated the closeness with acquaintance classes, we propose a LUNA factor, an outlier metric based on the relative density of the deep features, which is adaptive to the frequency of the targeted category. The LUNA factor is the first indicator, to our best knowledge, that provides quantitative measurements of novelty under the long-tailed distribution.

- We extensively evaluate LUNA on the MS-LT dataset and two commonly-used artificially-sampled datasets, ImageNet-LT and Place-LT, in both long-tailed recognition and open-set detection tacks. The result shows that the LUNA significantly outperforms the state-of-the-art methods by 4-6% on the closed set and in average 4% improvement of the F-measure under the open-set setting.

### 1.2.3 Improving Open-set Long-tailed Recognition

Although achieving the state-of-the-art performance, LUNA framework suffers from parameter selection and imbalanced feature space. We further tackle these problems from two aspects:

- To enforce a balanced feature space, we take advantages of a set of pre-generated center points that are uniformly-distributed. The distance between the training data and their corresponding targeted center points are minimized through optimization. Therefore, the conflict between representation learning and classifier training mitigates. Besides, to ensure

nearby data clusters are also close in the semantic field, we apply a self-assignment algorithm between learned centers and targeted.

- To better model the long-tailed data, we propose a distribution-sensitive loss function and frequency-based parameter selection. Compared to the original LUNA, the new framework LUNA+ is more automatic and robust.

The remaining of the thesis is organized as follows. In Chapter 2, reviews of the related works are presented on both closed-set long-tailed recognition and open-set recognition. The proposed ACE network, LUNA and LUNA+ algorithms are addressed in Chapter 3, Chapter 4 and Chapter 5, respectively, where the methods are illustrated in detail and experimental results are discussed. We then conclude in Chapter 6 and discuss the future works.

# Chapter 2

# RELATED WORKS

## 2.1 Long-tailed Recognition

Methods for long-tailed recognition can be mainly grouped into three types: (1) revision of the data distribution; (2) two-stage training and transfer learning; (3) multi-expert/branch frameworks.

### 2.1.1 On the Data: Re-balancing and Augmentations

Re-balancing consists of under-sampling of the head classes, over-sampling of the tail classes and re-weighting the loss function by the frequency or importance of the samples [44, 65, 17, 11]. Naive re-sampling in a class-balanced manner [41, 108] can easily overfit on the sample-few classes, either constructing a less imbalanced distribution by square-root sampling [73] or adjusting from instance-balanced to class-balanced sampling progressively [11, 17, 127] is a more stable and promising alternative.

Besides, strong data augmentations could increase the diversity of the training set, which compensates for the insufficiency of data and improves the model's generalizability. Mixup [120] along with its long-tailed variant re-balanced Mix-up (ReMix) [14]; and tail classes synthesis [123] are representative methods. However, the above algorithms always sacrifice the tails for the heads, or vice versa. The reason is the contradiction between representation learning and classifier learning, i.e., instanced-based (bias) sampling learns the most generalizable representations while the unbiased classifier is less likely to overfit the re-sampled set.

### 2.1.2 On the Representation: Two-stage Training and Transfer Learning

The second category migrates the learned knowledge from the heads to tails by two-stage training or memory-based transfer learning. Deferred re-balancing by re-sampling (DRS) and by re-weighting (DRW) [11] train the classifier layers with re-balancing after obtaining good representation on the imbalanced set at the first stage. Kang et al. [47] proposed $\tau$-norm and learnable weight scaling

(LWS) to re-balance the decision boundaries of classifiers in the parameter domain. OLTR [70] and inflated episodic memory (IEM) [130] utilize memory banks for prototype learning and knowledge transfer among classes. However, the use of re-balancing can still hurt the accuracy of heads, and the inevitable memory consumption potentially limits the deployment on large-scale datasets.

### 2.1.3 Ensemble Methods: Multi-expert Networks

The recent trend on multi-expert or multi-branch networks shows the strong potential to address the long-tailed issue by treating the relatively balanced sub-groups separately. BBN [127] assigns two branches with normal and reversed sampling, respectively, which cooperates cumulative learning strategy to adjust the bilateral training. BBN merges the two-stage methods into one, but still suffers from the same drawbacks. LFME [113] and RIDE [107] are multi-expert architectures that learn diverse classifiers in parallel, combining with knowledge distillation and distribution-aware expert selection. The main difference between our proposal and these two state-of-the-art methods are summarized in Table 2.1. Though achieving impressive performance, both of them suffer from extensive hyper-parameter tuning to balance the multiple optimization functions. More importantly, the multi-stage training requirement makes them not integrated into other tasks, like detection and segmentation.

| Method | Data for Experts | Relationship of Experts | Number of Training Stages | Majority Gain | Minority Gain |
|---|---|---|---|---|---|
| LFME [113] | non-overlapping splits | independent | 2 | + | + |
| RIDE [107] | same full set | competing and complementary | 3 | ++ | + |
| **ACE (Ours)** | overlapping splits | supportive and complementary | 1 | + | ++ |

Table 2.1: Comparisons between the proposed method with two SOTA multi-expert networks.

## 2.2 Novelty Detection

Novelty detection means the identification of out-of-distribution samples within the testing dataset, including the anomalous ones from trained classes and those from new categories, while maintaining the accuracy on known classes. Since negative samples are not available during training, the purpose

of novelty detection is to reject the initial outputs on the outliers, which are beyond the trained model's capability and requires further processing. Specifically, in the context of deep learning and taxonomy anomalies (i.e., out-of-class in testing), the methods fall in two types:

### 2.2.1 Single-class Novelty Detection

Single-class novelty detection means only one class is considered novel while the reminders are normal. Early works [23, 117] are statistical-based, by modeling the distributions of inliers and identifying the poorly-fit data as outliers. Later, learning-based methods usually learn a latent space for the known classes and evaluate the query image's projections in that space. [88, 115, 35] use the distance between the input and its reconstruction from a auto-encoder structure to detect anomalies. Generative adversarial networks [87, 81, 80] are also widely applied to extensively generate in-class samples to improve the representation of known classes.

### 2.2.2 Multi-class Novelty Detection

Multi-class novelty detection means novel samples are from multiple classes, also known as open-set recognition (OSR). OpenMax [5] is an extension of SoftMax, which uses the scores from the penultimate layer (activation vector, AV) of deep networks to estimate the probabilities of novelty. Extreme Value Theory (EVT) implies that the AVs follow Weibull distribution for closed set samples, therefore, OpenMax takes the all training data's AV per class to do Weibull fitting and results in a thresholding parameter that is used to estimate outsider regarding that class. Generative OpenMax (G-OpenMax) [28] synthesises novel class samples from mixture distributions of known classes in latent space, which enables OpenMax can also learn from unknown classes.

OpenMax and its variants are a huge step moving from closed-set to open-set recognition, however, there are still unexplored problems. First, the training data for these approaches are balanced and sufficient. In practice, the few-shot issue and imbalanced distribution post extra challenges in closed-set accuracy and latent space modeling. In addition, generative models are less stable and generalizable on large-scale high-resolution data. Therefore, open-set long-tailed recognition (OLTR) is has attracted increasing attention recently.

### 2.3 Open-set Long-tailed Recognition

Liu et al. [70] proposed the concept of open-set long-tailed recognition (OLTR), which aligns the natural data settings. Similar as the novelty detection methods, the proposed OLTR approach maps the images to feature space. To separate the majority and minority classes, a direct feature and a memory feature are aggregated and mutually enhanced for each known class. Open-set samples are detected by their calibrated feature distances to the learned visual memory. [70] is the first work tackles long-tailed and open-set jointly. Inflated Episodic Memory (IEM) [130] further improves the design of visual memory, since the single vector in OLTR might not be sufficient to model the distribution. IEM records the most discriminative features for each class by a group of memory bank modules. Both OLTR and IEM focus on the representation learning and use feature distances to detect novel data, while the quantitative measurement of novelty are oversimplified. Besides, same learning paradigm to all categories (e.g., dimension of embeddings, learning rate and loss weights) might lead to over-fitting or under-representation due to the difference in frequency.

# Chapter 3

# ACE: CLOSED-SET LONG-TAILED RECOGNITION

## 3.1 Overall Framework

The architecture of the proposed Ally Complementary Experts (ACE) network is shown in Figure 3.1. Followed a shared backbone, multiple experts are branched out with individual learnable blocks and a prediction layer. A distribution-aware planner distributes diverse but overlapping category splits for each expert, including target categories (TC) and interfering categories (IC). These experts complement each other from three aspects:

(1) the dominating categories in their TCs are different, so that the predictions have their own strengths;

(2) the TCs are overlapping, especially on sample-few categories, thus the predictions support each other;

(3) the experts learn to suppress the output of IC so that they will not bring ambiguity in the categories that have never been seen. To further accommodate the disparity in data, a distribution-adaptive optimizer is designed to guide the experts to update at their own paces.

We use classification loss $L_{cls}$ and complement loss $L_{com}$ to train the model end-to-end in only one-shot. Finally, the predictions from the experts are aggregated by averaging over the re-scaled logits in each data split.

## 3.2 Methodology

### 3.2.1 Distribution-aware Planner

The experimental fact that classifiers tend to have better performance on the majority categories than the minority on an imbalanced set is considered a drawback and avoided by existing methods. However, if each split's prediction is obtained from a classifier that biases on it, we could expect accuracy gains everywhere. Therefore, we design a distribution-aware planner to assign each expert with a subset of the training set, which is also imbalanced and dominated by different splits,

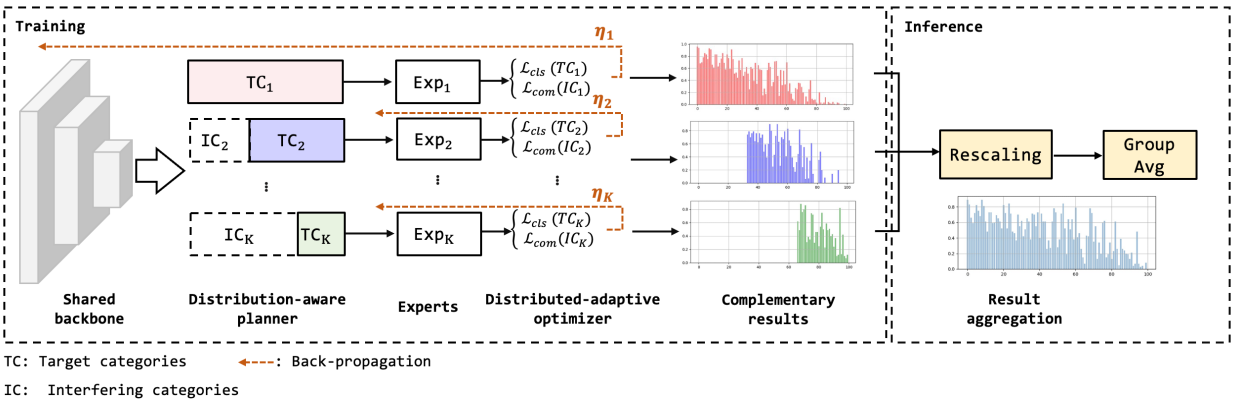



Figure 3.1: Network architecture of ACE. There are four components: (1) a shared backbone for representation learning; (2) a distribution-aware planner assigns diverse target categories (TC) and interfering categories to each expert, respectively; (3) a group of experts that learns to identify the TC with classification loss $L_{cls}$ and eliminate their effect on IC with complementary loss $L_{com}$; (4) a distribution-adaptive loss that adjust learning pace $\eta$ of each expert for simultaneous convergence. By allying complementary experts (ACE) in a group average manner, the aggregated prediction compromises the merits of all experts.

respectively. Formally, the process is as follows,

Given a training set $\mathbb{D} = \{X; Y\}$ with $C$ categories in total, for $K$ experts $\mathbb{E} = \{\mathscr{E}_1, \mathscr{E}_2, ..., \mathscr{E}_K\}$, $\mathscr{E}_i$ is assigned categories $\mathscr{C}_i$, such that $|\mathscr{C}_1 \cap \mathscr{C}_2 \cap ... \cap \mathscr{C}_K| = C$ and $\mathscr{C}_i \cup \mathscr{C}_j \neq \varnothing$.

Denoted $\mathscr{N} = \{n_1, n_2, ...n_C\}$ as the number of samples in each class, to be simple, suppose $\mathscr{N}$ is in the descending order. Similar to the spirit of re-balancing, sample-few categories should be more exposed. Therefore, the $i$-th expert $\mathscr{E}_i$ is assigned target and interfering classes

$$\begin{aligned} \mathscr{C}_i &= \{\frac{C}{K}(i-1), \frac{C}{K}(i-1)+1, \frac{C}{K}(i-1)+2, ..., C\}, \\ \widetilde{\mathscr{C}}_i &= \{1, 2, ..., \frac{C}{K}(i-1)-1\}. \end{aligned} \tag{3.1}$$

For a randomly-sampled mini-batch $\mathrm{B} \subset \mathbb{D}$, $\mathscr{E}_i$ uses a sub-batch $\mathrm{B}_i = \{(x, y) : (x, y) \in \mathrm{B}, y \in \mathscr{C}_i\}$. In this case, there is always an expert be presented with all the samples, and the smaller the class, the more experts are assigned. Besides, the medium-shot or few-shot classes have chances to dominate an expert, thus eliminating the bias towards the sample-rich classes. The network degenerates to a plain classifier if $K = 1$.

Similar to the existing methods, we use ResNet [38] as our backbone. The last residual block

is duplicated for each expert, and followed by a learnable weight scaling (LWS) classifier [47]. The output logits (before `SoftMax` layer) of $\mathscr{E}_i$ is $\mathbf{z}^i \in \mathbb{R}^{1\times|C|}$, which are further adjusted to be $\hat{\mathbf{z}}_i$ by the norm of the fully-connected layers' weights to have comparable scales:

$$\hat{\mathbf{z}}_i = \frac{\|\mathbf{w}_i\|^2}{\|\mathbf{w}_1\|^2} \cdot \mathbf{z}_i \tag{3.2}$$

The set of experts that trained with class $c$ is $\mathrm{S}^c$, then the output logit of class $c$ is the average among the outputs from $\mathrm{S}^c$, i.e.,

$$\mathbf{o}^c = \frac{1}{|\mathrm{S}_c|} \sum_{\mathscr{E}_i \in \mathrm{S}^c} \hat{\mathbf{z}}_i \tag{3.3}$$

`SoftMax` operation is applied on $\mathbf{o}$ to obtain the classification confidence.

### 3.2.2 Objective Functions

Loss functions are applied on each expert separately instead of on the aggregated output $\mathbf{o}$ to avoid a mixture of expect-specific features. We use the cross-entropy loss as the classification loss, with the sub-batch $\mathrm{B}_i$ for $\mathscr{E}_i$,

$$L^i_{cls}(\mathrm{B}_i) = -\sum^{\mathscr{C}_i} y \log(\sigma(\mathbf{z}_i)), \tag{3.4}$$

where $\sigma(\cdot)$ represents the `SoftMax` operation.

In addition to classifying the assigned targeted class, each expert is required not to affect the other experts on the classes they have never seen, i.e., the interfering categories (IC). For the experts themselves, categories in IC are the main source of confusion as well. By eliminating the effect of IC, the experts work in a complementary way rather than competitive. Hence, a regularization term to suppress the output of IC is necessary. We define the complement loss $L_{com}$ as

$$L^i_{com}(\mathrm{B}_i) = \sum_{c_j \in \widetilde{\mathscr{C}_i}}^{\mathscr{C}} \|\mathbf{z}_i^{c_j}\|^2. \tag{3.5}$$

The complement loss minimizes the logits of non-target categories for $\mathscr{E}_i$ so as to put down their effect. Removing the logits representing the sample-rich categories empirically might also do the trick; however, there is no learning process to distinguish IC and TC. We compare this architecture with $L_{com}$, discussions could be found in Sec 3.4.

Overall, the loss function for $\mathscr{E}_i$ is

$$L_{\mathscr{E}_i}(\mathrm{B}_i) = L^i_{cls} + L^i_{com}. \tag{3.6}$$

### 3.2.3 Distribution-adaptive Optimizer

Recall the Linear Scaling Rule [31] for training networks in mini-batches with stochastic gradient descent (SGD) optimizer: *when the minibatch size is multiplied by k, multiply the learning rate by k. All other hyper-parameters (weight decay, momentum, etc.) are kept unchanged.*

By the rule, to avoid over-fitting, the optimizer should be distribution-aware to assign smaller weights to $\mathscr{E}_i$ which is trained with less data. Denoted the base learning rate as $\eta_0$, which is the learning rate for the expert presented with all categories, the $i$-th expert is trained by,

$$\eta_i = \eta_0 \cdot \frac{\sum_{c \in \mathscr{C}_i} n_c}{\sum^{\mathscr{C}} n_j}. \tag{3.7}$$

The loss of $\mathscr{E}_1$ updates the backbone and parameters of $\mathscr{E}_1$, and $L_i$ that $i > 1$ only updates the expert itself. The reason is the errors likely duplicates because of data overlapping, which means the backbone could be corrected multiple times due to the same error. This is similar to the idea of re-weighting methods, as introduced in Section 2.1.1 that hurt the representation learning. Therefore, only $\mathscr{E}_1$ updates the backbone.

## 3.3 Experimental Results

### 3.3.1 Datasets and Protocols

Generally, in long-tail recognition tasks, the classes are categorized into many (with more than 100 training samples), medium (with 20 ∼100 samples) and few (with less than 20 samples) splits [70]. The imbalance factors of the long-tailed datasets, defined as the frequency of the largest class divided by the smallest class, vary from 10 to over 500 [17, 70, 98].

**CIFAR100-LT and CIFAR10-LT** [17] are artificially created from the balanced CIFAR dataset [54] by reducing training samples according to an exponential function $n = n_i \mu^i$, where $i$ is the class index, $n_i$ is the original number of samples and $\mu \in (0, 1)$. We experiment with two commonly used IFs, 100 and 50. There are approximately 10K∼13K training images and 10K testing images for each split. ResNet-32 is used as the base network, where the last residual block is

tripled for the branches to be comparable with other methods. Following [38], for training samples, 4 pixels are padded on each side, following by a $32 \times 32$ random crop on the padded image or its horizontal flip. The network is trained by the stochastic gradient descent (SGD) optimizer with a momentum 0.9 for 400 epochs. The base learning rate is 0.05 and decreases by 0.1 at epoch 320 and 360, respectively.

**ImageNet-LT** [70] is sampled from ImageNet-2012 [20] following the Pareto distribution with the power value $\alpha = 6$. ImageNet-LT contains 115.8K images for 1000 categories, with maximally 1280 images per class and minimally 5 images per class. Following [70, 47, 113, 123], we use ResNet-10 as the backbone. To be comparable with [47, 107], we also report our results with ResNet-50 and ResNeXt-50 [114]. For data pre-processing, the training samples are resized to $256 \times 256$, then randomly cropped to $224 \times 224$ and flipped horizontally with a probability of 0.5; on testing, the aspect ratio of the testing sample is kept by first resize its shorter side to 256 then crop $224 \times 224$ in the center. The networks are trained by the SGD optimizer with momentum 0.9 for 100 epochs. The base learning rate is 0.1 and decreases by 0.1 at epoch 120 and 160.

**iNaturalist2018** [98] is a real-world large-scale dataset for species identification of animals and plants. Following the literature, we use the 2018 version. This dataset contains 438K images for over 8K categories, with extremely imbalanced distribution (IF=512) and challenging fine-grained issues. We use ResNet-50 as the backbone, and the same prepossessing and training protocol as ImageNet-LT.

### 3.3.2 Performance

#### Competing Methods

We first introduce some competing methods. Generally, there are two types by whether or not there is a backbone pre-training stage. For one-stage type of methods, re-balancing of the long-tailed dataset is either by resampling, (e.g., class-balanced and progressively-balanced [47]), or reweighting (e.g., focal loss [65], class-balanced focal loss [17], and LDAM [11]). Besides, strong augmentation tricks (e.g., mixup [120], re-balanced mixup [14], tail sample synthesis using class activation maps (CAM) [123])) can also benefit the overall accuracy, especially the heads. Moreover, transfer learning in either image domain (major-to-minor translation [51]) and in feature domain

(OLTR [70]) are proved useful. BBN [127] uses a two-branch architecture to combine normal sampling and distribution-reversed sampling progressively, improving the tail's accuracy in a large margin. The other type is two-stage methods. In the second stage, $\tau$-norm, LWS and cRT [47] retrain or fine-tune the classifier with a balanced dataset or unbiased classifier weights. LFME [113] and RIDE [107] are multi-branch assembled architectures with knowledge distillation. LFME uses teacher-student network to train experts on many-/medium-/few-shot splits, while RIDE does not fix the number of branches and uses KL-divergence loss to force them to be experts on different groups.

### CIFAR-LT

Table 3.1 shows the proposed ACE performs the best among all one-stage methods and surpasses other multi-stage methods on CIFAR100-LT-100. We outperform the previous one-stage SOTA BBN by 9.0%. Class-wise comparison is shown in Figure 3.2. ACE has significant advantages in medium and few-shot categories. It is also the only method that improves all the groups by a single stage. Table 3.3 shows the top-1 accuracy on CIFAR10-LT and CIFAR100-LT with imbalance factor 50 and 100.

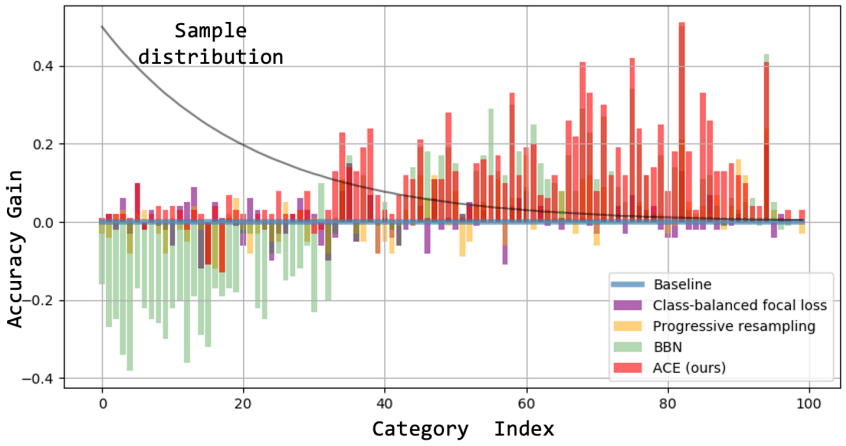



Figure 3.2: Accuracy gain comparisons between representative one-stage long-tailed recognition methods and baseline.

| Type | Method | Multiple experts | Accuracy | | | |
|---|---|---|---|---|---|---|
| | | | **All** | **Many** | **Medium** | **Few** |
| **One-Stage** | Baseline (ResNet-32) | | 38.3 | 65.2 | 37.1 | 9.1 |
| | CB resampling [44]§ | | 36.0 | 59.0 | 35.4 | 10.9 |
| | Focal loss [65] | | 37.4 | 64.3 | 37.4 | 7.1 |
| | CB Focal loss [17]§ | | 38.7 | 65.0 | 37.6 | 10.3 |
| | Progressive [47] | | 39.4 | 63.3 | 38.8 | 13.1 |
| | ReMix [14] | | 40.9 | 69.6 | 40.7 | 8.8 |
| | Mixup [120] | | 41.2 | 70.7 | 40.4 | 8.8 |
| | BBN [127] | ✓ | 39.4 | 47.2 | 49.4 | 19.8 |
| | Logit Adjustment [72] | | 43.9 | - | - | - |
| | **ACE (3 experts)** | ✓ | 49.4 | 66.1 | **55.7** | 23.5 |
| | **ACE (4 experts)** | ✓ | **49.6** | **66.3** | 52.8 | **27.2** |
| **Multi-Stage** | $\tau$-norm [47] | | 43.2 | 65.7 | 43.6 | 17.3 |
| | cRT [47] | | 43.3 | 64.0 | 44.8 | 18.1 |
| | LDAM+DRW [11] | | 42.0 | 61.5 | 41.7 | 20.2 |
| | LDAM+LFME [113] | ✓ | 43.8 | - | - | - |
| | LDAM+M2m [51] | | 43.5 | - | - | - |
| | CAM [123] | ✓ | 47.8 | - | - | - |
| | RIDE [107] (2 experts) | ✓ | 47.0 | 67.9 | 48.4 | 21.8 |
| | RIDE [107] (3 experts) | ✓ | 48.0 | 68.1 | 49.2 | 23.9 |
| | RIDE [107] (4 experts) | ✓ | 49.1 | 69.3 | 49.3 | 26.0 |

Table 3.1: Top-1 accuracy on CIFAR100-LT-100. Best performance are marked in **bold**. Our ACE is the only one-stage method with performance gain on all groups and of the best over all categories. §: CB represents class-balanced.

### ImageNet-LT and iNaturalist

We also report our performance on ImageNet-LT with various backbone models of ResNet-10, ResNet-50 and ResNeXt-50 as well as iNaturalist-LT with ResNet-50, shown in Table 3.2. Our method outperforms the BBN by 6.4% (ResNet-50) and 7.3% (ResNeXt-50) on ImageNet-LT and 3.9% on iNaturalist2018, respectively.

| Method | ImageNet-LT | | | iNaturalist |
|---|---|---|---|---|
| | Res10 | Res50 | ResX50 | Res50 |
| Baseline | 20.9 | 41.6 | 44.4 | 66.1 |
| FSLwF [29] | 28.4 | - | - | - |
| Range Loss [121] | 30.7 | - | - | - |
| Lifted Loss [77] | 30.8 | - | - | - |
| Focal loss [65] | 30.5 | - | - | 60.3 |
| CB Focal loss [17] | - | - | - | 61.1 |
| BBN [127] | - | 48.3 | 49.3 | 68.0 |
| Logit Adj.[72] | - | 51.1 | - | 66.4 |
| **ACE (3 experts)** | **44.0** | 54.7 | **56.6** | **72.9** |
| OLTR [70] | 34.1 | - | 46.3 | 63.9 |
| NCM [47] | 35.5 | 44.3 | 47.3 | - |
| LDAM+DRW [11] | 36.0 | - | - | 68.0 |
| cRT [47] | 41.8 | 47.3 | 49.5 | 65.2 |
| $\tau$-norm [47] | 40.6 | 46.7 | 49.4 | 65.6 |
| LWS [47] | 41.4 | 47.7 | 49.9 | 65.9 |
| CAM [123] | 43.1 | - | - | 70.9 |
| LFME [113] | 38.8 | - | - | - |
| RIDE [107]† | - | 54.4 | 55.9 | 71.4 |
| RIDE [107]‡ | - | **54.9** | 56.4 | 72.2 |

Table 3.2: Top-1 accuracy on ImageNetLT and iNaturalist2018, best results are marked in **bold**. Overall, it shows the multi-expert/branch architecture outperforms the re-balancing methods. Our ACE has consistent performance gain comparing with other one-stage methods with multiple backbones, and is comparable with multi-stage methods.†:2 experts, ‡:3 experts.

## 3.4 Analysis and Discussions

### 3.4.1 Effectiveness of Complementary Experts

We compare ACE with its two variants to show the effectiveness of its architecture, learning process and the loss function: one is training without $L_{com}$ and the other is the non-complementary architecture, which is called split-specific classifier (SSC). In the latter one, output dimensions of the classifier of $\mathscr{E}_i$ are the same as $|\mathscr{C}_i|$, i.e., $\mathbf{z}^i \in \mathbb{R}^{1\times|\mathscr{C}_i|}$. In other words, the weights of non-target

| Method | CIFAR100-LT | | CIFAR10-LT | |
|---|---|---|---|---|
| | **100** | **50** | **100** | **50** |
| Baseline | 38.3 | 42.1 | 69.8 | 75.2 |
| Focal loss [65] | 37.4 | 42.4 | 70.4 | 75.3 |
| Mixup [120] | 39.5 | 45.0 | 73.1 | 77.8 |
| CB Focal loss [17] | 38.7 | 46.2 | 74.6 | 79.3 |
| BBN [127] | 39.4 | 47.0 | 79.8 | 82.2 |
| Logit Adj.[72] | 43.9 | - | 77.7 | - |
| **ACE (3 experts)** | **49.4** | **50.7** | **81.2** | **84.3** |
| **ACE (4 experts)** | **49.6** | **51.9** | **81.4** | **84.9** |
| LDAM+DRW [11] | 42.0 | 45.1 | 77.0 | 79.3 |
| LFME [113] | 42.3 | - | - | - |
| LDAM+M2m [51] | 43.5 | - | 79.1 | - |
| CAM [123] | 47.8 | 51.7 | 80.0 | 83.6 |
| RIDE [107] | 49.1 | - | - | - |

Table 3.3: Top-1 accuracy on CIFAR100-LT and CIFAR10-LT with imbalance factor 100 and 50.

groups are set to be zero as a hard constraint, instead of learning to suppress them with $L_{com}$ in a soft regularization manner. Results are in Table 3.4. Figure 3.3 shows ACE with $L_{com}$ in the top row, where $\mathscr{E}_i$ learns similar scales over all the data splits and the scales of interfering classes are zeros. Therefore, all trained experts have an approximately equal contribution to the shared splits. We also observe that on the minority splits $j$ of $\mathscr{E}_i$ generates supportive results for $\mathscr{E}_j$ (e.g., $\mathscr{E}_1$ is peripheral in the few-shot split, and its scales are smaller than those of $\mathscr{E}_3$, so it is just a supplementary to $\mathscr{E}_3$'s output.). As seen from the middle row of Figure 3.3, by splitting the data to complementary batches, but without $L_{com}$, all experts compete with each other in the common splits. For example, $\mathscr{E}_1$ is strong over all categories though it is less accurate in the minority classes compared to $\mathscr{E}_3$, $\mathscr{E}_1$'s scales are still larger than $\mathscr{E}_3$'s. This explains why ACE without $L_{com}$ has the best performance in the head categories. In the experiments on SSC, where the experts learn to classify $\mathscr{C}_i$ but cannot distinguish the untrained categories, resulting in the obvious dominance of the expert trained with the full set in all splits, making other experts useless.

Results here are inspiring: different from most exiting works that try to eliminate the bias, we

utilize it. The data re-balancing is embedded in the data assignment to ensure more exposure of the minority. The individual back-propagation of each expert will not hurt the representation learning. Therefore, $L_{com}$ decouples the representation learning and classifier training in one stage.

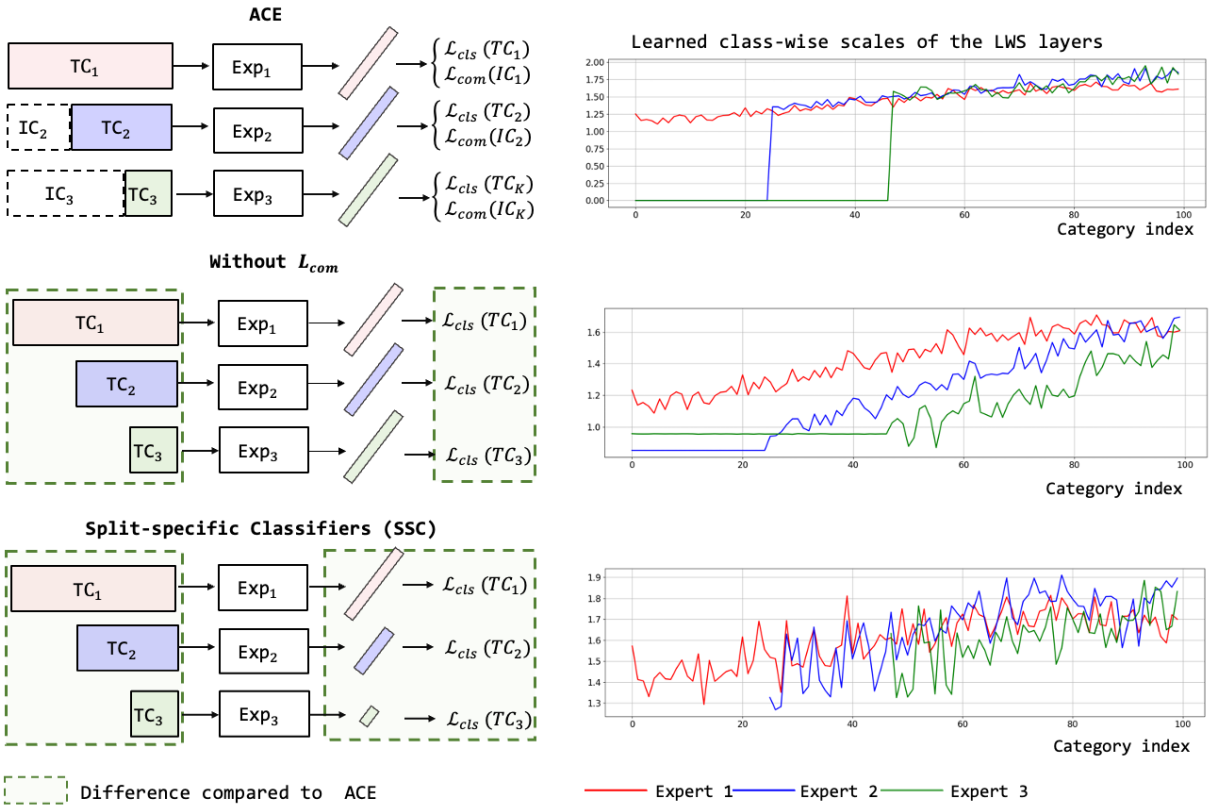



Figure 3.3: The classifier learnt scales of three model: ACE trained with complementary loss (top), ACE trained without complementary loss (middle), and split-specific classifiers (SSC) (bottom) trained on CIFAR100-LT-100. $\mathscr{E}_1$, $\mathscr{E}_2$ and $\mathscr{E}_3$ are plotted in red, blue and green colors, respectively. Complementary loss allows the experts work jointly in their common splits. Without the complementary loss, the experts trained with full batch has the largest scales on all splits and competes with the real dominating experts.

### 3.4.2 Effectiveness of Distribution-aware Optimizer

The distribution-aware optimizer controls the learning speed of each expert with various data assignments. In this section, we compare the linear scaling rule with the square-root scaling [53]

| Method | All | Many | Medium | Few |
|---|---|---|---|---|
| **ACE (With $L_{com}$)** | **49.4** | **66.1** | **55.7** | **23.5** |
| Expert 1 | 41.9 | 71.2 | 40.2 | 10.7 |
| Expert 2 | 30.7 | 19.9 | 53.7 | 17.7 |
| Expert 3 | 21.8 | 0.0 | 38.7 | 27.8 |
| **Without $L_{com}$** | **47.2** | **71.5** | **49.4** | **17.5** |
| Expert 1 | 42.0 | 71.0 | 40.9 | 10.5 |
| Expert 2 | 31.1 | 19.4 | 53.8 | 19.4 |
| Expert 3 | 22.0 | 0.0 | 38.8 | 28.3 |
| **With SSC** | **43.4** | **65.1** | **44.4** | **18.0** |
| Expert 1 | 41.6 | 68.2 | 41.2 | 12.1 |
| Expert 2 | 16.0 | 2.4 | 26.5 | 19.9 |
| Expert 3 | 21.4 | 0.0 | 38.6 | 26.7 |

Table 3.4: Overall and many-/medium-/few-shot split top-1 accuracy on CIFAR100-LT-100 of the three model. The results are consistent with our analysis that without complementary loss, the experts are competing, so the results tend to average. Split-specific classifiers (SSC) depends mostly on $E_1$.

and a uniform optimizer. [53] indicates when multiplying the batch size by $S$, one should multiply the learning rate by $\sqrt{S}$ to keep the variance in the gradient expectation constant. For a uniform optimizer, all the experts share the same $\eta$, i.e.,

$$\eta_i^{sqrt} = \eta_0 \cdot \sqrt{\frac{\sum_{c \in \mathscr{C}_i} n_c}{\sum^{\mathscr{C}} n_j}}, \eta_i^{uni} = \eta_0 \tag{3.8}$$

The training will be more sensitive to the variance of data with a larger learning rate. For the experts trained by minority splits, we have $\eta_i^{uni} >> \eta_i^{sqrt} > \eta_i^{linear}$. The comparison of the results is shown in Table 3.5. All three schemes produce better results than baseline. $\eta_i^{uni}$ promotes the higher improvements in the majority categories, while significantly decreases the tails. The reason is several experts converge too early and thus not effective due to over-fitting. $\eta_i^{sqrt}$ and $\eta_i^{linear}$ show similar performance, while $\eta_i^{linear}$ is better in many and med-shot splits. By comparing $\eta_i^{sqrt}$ and $\eta_i^{linear}$, we observe learning rate is not the principal reason for accuracy booms. We conclude that selecting a proper optimization scheme with respect to the data distribution will benefit the overall performance.

| Scheme | All | Many | Medium | Few |
|---|---|---|---|---|
| **Linear** | 49.4 | 66.1 | 55.7 | 23.5 |
| **Square-root** | 49.1 | 67.1 | 55.2 | 22.1 |
| **Uniform** | 41.7 | 69.7 | 39.9 | 10.7 |

Table 3.5: Comparisons of learning rate scaling schemes on CIFAR100-LT-100.

### 3.4.3 Effectiveness of Group Average Output Aggregation

We compare different aggregation methods of the output logits $\{\mathbf{z}^i\}$ from $K$ experts. Four variants of the aggregation methods are shown in Figure 3.4 (3 experts).

Comparisons between (ACE) and (4) in Table 3.6 shows that the design of scaling is for preserving the accuracy of the head classes. Computing the maximum by groups over the scaled logits (2) also suppresses the performance on the heads, as the experts for small classes are easier to overfit and thus overconfident. Concatenating the result of each dominating group of the experts amplifies the drawbacks of overconfidence, and experts competes each other. Overall, merging multiple experts is a trade-off for one-stage methods, in which all experts are trained from scratch. On the other hand, our ACE balances them by adjusting learning speed and with complementary loss, achieving improvements for all groups.

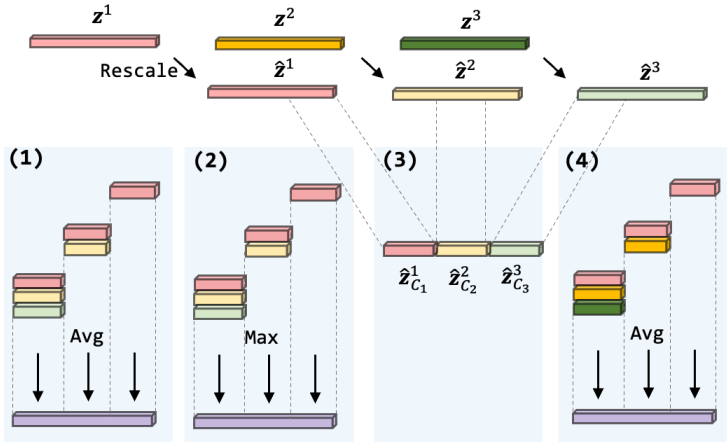



Figure 3.4: Illustration of variants of output aggregation methods.

| Aggregation | All | Many | Medium | Few |
|---|---|---|---|---|
| Group Avg w/ scaling (ACE) | 49.4 | 66.1 | 55.7 | 23.5 |
| Group Max (2) | 43.4 | 47.5 | 54.2 | 26.5 |
| Group Concat (3) | 37.7 | 30.3 | 50.2 | 22.9 |
| Group Avg w/o scaling (4) | 46.7 | 49.5 | 53.0 | 36.5 |

Table 3.6: Ablation study on aggregation of the outputs on CIFAR100-LT-100.

Chapter 4

# LUNA: OPEN-SET LONG-TAILED RECOGNITION

## 4.1 Overall Framework

In this chapter, we introduce our approach on training the classification network with a proposed distribution-sensitive loss to obtain distinctive representations, and detect novel classes based on the Localizing Unfamiliarity Near Acquaintance (LUNA, detailed illustration in Figure 4.1) measurement under the open-set setting.

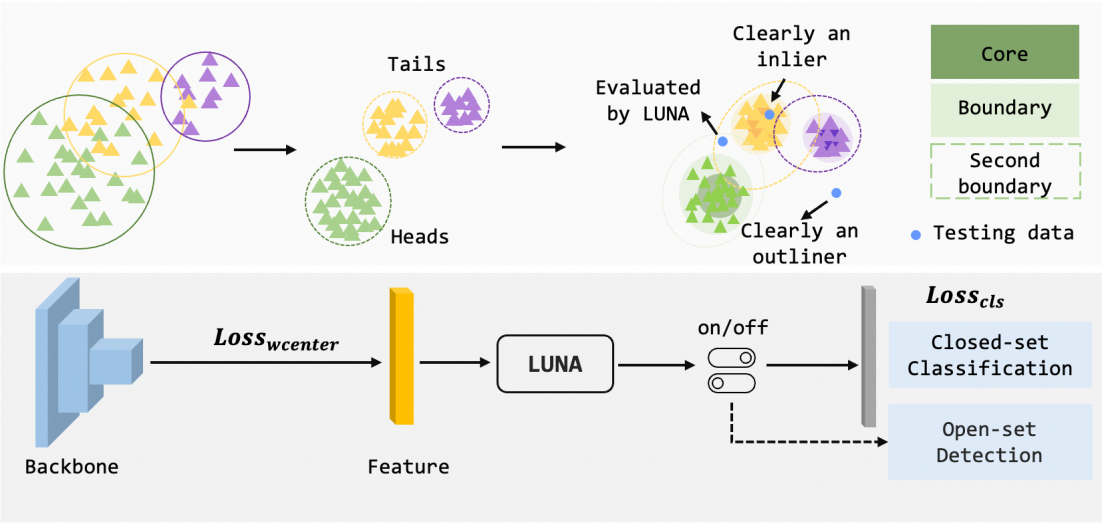



Figure 4.1: The illustration of our proposed method. Bottom: the workflow of open long-tailed recognition training and inference. Top: a brief illustration of our open-set detection method in feature space. The training samples form clusters in the feature space with the weighted center (wcenter) loss are further categorized into core, boundary and second boundary points by their relative local density. LUNA assesses the aforementioned metrics to measure the novelty of the testing samples.

## 4.2 Methodology

### 4.2.1 Problem Setup

Let the training images and the corresponding labels be represented as $\mathbf{X} = \{\mathbf{x}_i\}_{i=1:M}$ and $\mathbf{Y} = \{y_i\}_{i=1:M}$, the LUNA framework consists of three components:

- a neural network $f : \mathbf{X} \to \mathbf{Z} = \{\mathbf{z}_i \in \mathbb{R}^d\}_{i=1:M}$, where $z_i$ is the extracted feature for $x_i$,
- a classifier $g : \mathbf{Z} \to \mathbf{L} = \{\mathbf{l}_i \in \mathbb{R}^T\}_{i=1:M}$, where $l_i$ is the output logits for $T$ closed-set classes,
- a novelty measurement LUNA factor $h : \mathbf{Z} \to \mathbf{S} = \{s_i \in \mathbb{R}^1\}_{i=1:M}$.

All notations used in this section are summarized in Table 4.1 for clarification.

| Symbol | Meaning |
|---|---|
| $i, j$ | Index of input samples. |
| $t, r$ | Index of closed-set categories. |
| $d$ | Dimension of extracted features. |
| $M$ | Number of input samples. |
| $T$ | Number of closed-set categories. |
| $\mathbf{X} = \{\mathbf{x}_i\}_{i=1:M}$ | Set of $M$ input images. |
| $\mathbf{Y} = \{y_i\}_{i=1:M}$ | Set of labels w.r.t. $\mathbf{X}$. |
| $\hat{\mathbf{Y}} = \{\hat{y}_i\}_{i=1:M}$ | Set of predicted labels w.r.t. $\mathbf{X}$. |
| $\mathbf{Z} = \{\mathbf{z}_i\}_{i=1:M}$ | Set of extracted features w.r.t. $\mathbf{X}$. |
| $\mathbf{L} = \{\mathbf{l}_i\}_{i=1:M}$ | Set of output logits w.r.t. $\mathbf{X}$. |
| $\mathbf{S} = \{s_i\}_{i=1:M}$ | Set of LUNA scores w.r.t. $\mathbf{X}$. |
| $\mathbf{N} = \{n_i\}_{t=1:T}$ | Number of samples per closed-set class. |
| $\widetilde{\mathbf{N}} = \{\widetilde{n}_i\}_{t=1:T}$ | Normalized number of samples per closed-set class. |
| $\mathbf{C} = \{\mathbf{c}_t\}_{t=1:T}$ | Set of pre-generated cluster centers. |
| $\hat{\mathbf{C}} = \{\hat{\mathbf{c}}_t\}_{t=1:T}$ | Set of learned cluster centers. |

| Symbol | Meaning |
|---|---|
| $\mathbf{P}$ | Set of positive samples in contrastive loss. |
| $f(\cdot)$ | Feature extraction network. |
| $g(\cdot)$ | Classifier. |
| $h(\cdot)$ | Functions for LUNA factor calculation. |
| $b$ | Bias term in weighted center loss. |
| $d_k(\mathbf{z}_i)$ | k-distance between $z_i$ and its $k$-th nearest neighbour. |
| $rd_k(\mathbf{z}_i, \mathbf{z}_j)$ | Reachability distance between $z_i$ and its peer $z_j$. |
| $\mathscr{D}_k(\mathbf{z}_i)$ | Sub-local reachability density of $z_i$. |
| $\eta$ | Percentage of core features in a cluster. |
| $\mathbb{N}_k(\mathbf{z}_i)$ | Set ot $k$ sub-local nearest neighbors of $z_i$. |
| $\mathbb{N}_t^C$ | Set of core features for class $t$. |
| $\mathbb{N}_t^B$ | Set of boundary features for class $t$. |
| $\mathbb{N}_t^{sB}$ | Set of second boundary features for class $t$. |
| $\mathbf{p}$ | Feature of a testing sample. |
| $\mathbb{N}(\mathbf{p})$ | The targeted cluster's feature set of $\mathbf{p}$. |
| $\mathscr{F}_C(\mathbf{p})$ | Core Outlier Factor (COF) of $\mathbf{p}$. |
| $\mathscr{F}_B(\mathbf{p})$ | Boundary Outlier Factor (BOF) of $\mathbf{p}$. |
| $\mathscr{F}_{sB}(\mathbf{p})$ | Second Boundary Outlier Factor (sBOF) of $\mathbf{p}$. |
| $\alpha$ | Hyper-parameter for number of nearest neighbour. |
| $\theta$ | Classification confidence. |

Table 4.1: A summary of the symbols and their corresponding meanings.

### 4.2.2 Metric Learning with Weighted Center Loss

The center loss [110] is originally proposed for face recognition, which is formulated as

$$L_c = \frac{1}{2}\sum_i ||\mathbf{x}_i - \mathbf{c}_{y_i}||^2, \tag{4.1}$$

where $\mathbf{x}_i$ indicates the feature of the $i$-th sample with ground truth label $y_i$; $\mathbf{c}_{y_i}$ is the corresponding centroid, which is initialized randomly and updated iteratively to minimize the distance between itself and the continuously updated deep features during the training. Center loss is jointly trained with cross-entropy loss, balanced by a scalar parameter $\lambda$

$$L = L_{xent} + \lambda L_c, \tag{4.2}$$

where different $\lambda$ ($\lambda = 0.001, 0.01, 0.1, 1$) is shown to lead to different deep feature distributions [110], and features are more concentrated with larger $\lambda$. In the situation of long-tailed datasets, the tail classes tend to be sparser in distribution since there are much fewer samples, which are easier to mix up with other clusters in the feature space. Thus, we propose a weighted center (wcenter) loss that caters to imbalanced distribution.

$$L_{\mathrm{WC}} = \frac{1}{2}\sum_i \lambda_{y_i}||\mathbf{z}_i - \mathbf{c}_{y_i}||^2, \tag{4.3}$$

where $\lambda_{y_i}$ is the weight and $\mathbf{c}_{y_i}$ is the center of class $y_i$ and is obtained by taking the arithmetical average of all features belonging to that class. Denoting the number of samples per class as $\mathbf{N} = \{n_t\}_{t=1:T}$, the normalized frequency is

$$\widetilde{\mathbf{N}} = \{\tilde{n}_t\} = \{n_t/n_{max}\}_{t=1:T}, \tag{4.4}$$

where $n_{max} = \texttt{Max(N)}$ is the largest class size. Then the cost coefficient $\lambda$ is inversely proportional to the frequency to fit the long-tailed distribution of training data, i.e., errors on the tail classes are penalized more:

$$\lambda_t = \tilde{n}_t/\texttt{Max}(\widetilde{\mathbf{N}}) + b. \tag{4.5}$$

Here, $b$ is a hyper-parameter for scaling. We visualize the weights for ImageNet-Lt dataset in Figure 4.2. Noted that no open-set samples are generated or presented to $f$ during training.

Therefore, the output of classifier $g$ are logits for $T$ closed-set categories. We use a single-layer linear classifier followed by the softmax operation in LUNA, the prediction is represented as

$$\hat{\mathbf{Y}} = \texttt{SoftMax}(\mathbf{L}) = \texttt{SoftMax}(g(\mathbf{Z})). \tag{4.6}$$

The overall loss function is the combination of wcenter loss and cross-entropy (CE) loss

$$L = L_{\text{WC}}(\mathbf{Z}, \mathbf{Y}) + L_{\text{CE}}(\hat{\mathbf{Y}}, \mathbf{Y}), \tag{4.7}$$

where $b$ in wcenter loss is set to 1 to balance the two terms.

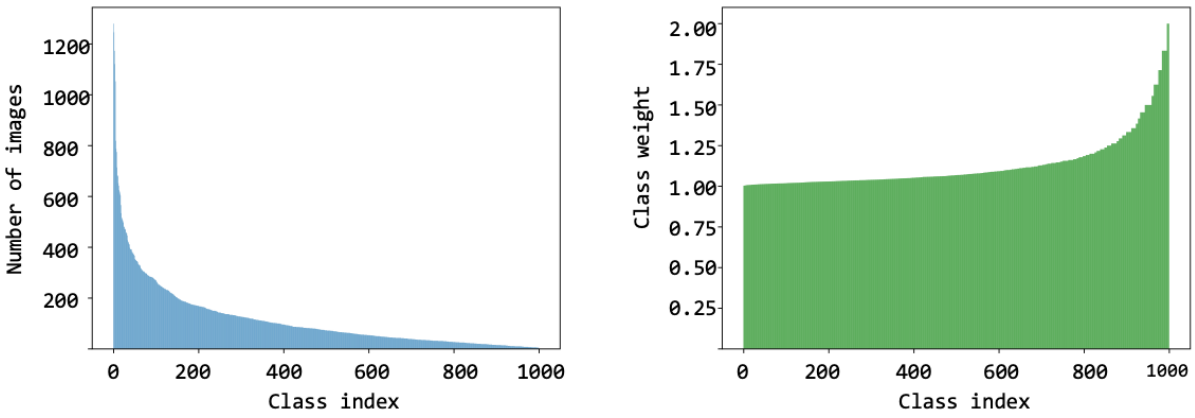



Figure 4.2: The long-tailed distribution of the ImageNet-LT dataset and the corresponding weights for the weighted center loss.

### 4.2.3 Metric Distance Measurement

With the well-clustered features, unlike prior methods [70, 130] that use the Euclidean distance to the closest centroid or memorized prototype as the criteria for novelty, LUNA uses the feature density instead. Following the famous single-class outlier detection algorithm Local Outlier Factor (LOF) [8], we define $d_k(\mathbf{z_i})$ as the *k-distance* between a feature $\mathbf{z_i}$ to its $k$-th nearest neighbour. The *reachability distance rd* between anchor feature $\mathbf{z_i}$ and a peer feature $\mathbf{z_j}$ is then defined as the maximum of the *regular distance* (usually use the distance metric as $d_k$, e.g, both are cosine

distance) $d$ between them and *k-distance* of $\mathbf{z_i}$:

$$rd_k(\mathbf{z}_i, \mathbf{z}_j) = \texttt{Max}(d(\mathbf{z}_i, \mathbf{z}_j), d_k(\mathbf{z_i})). \tag{4.8}$$

$rd_k$ is a more robust measurement in a feature set than direct distance, we recommend readers referring to [8] for mathematical derivation. With $rd_k$, we define the *sub-local reachability density* (hereafter denoted as $\mathscr{D}$) as the inverse of the average of the $k$-distance between $\mathbf{z_i}$ to its $k$ nearest neighbors $\mathbb{N}_k$ in the same cluster (so-called sub-local neighbors).

$$\mathscr{D}_k(\mathbf{z_i}) = 1 \Bigg/ \left( \frac{\sum_{\mathbf{z_j} \in \mathbb{N}_k} rd_k(\mathbf{z_i}, \mathbf{z_j})}{|\mathbb{N}_k(\mathbf{z}_i)|} \right) \tag{4.9}$$

If a feature is farther away from its neighbors, its density $\mathscr{D}_k$ is smaller, indicating it is at a sparser neighborhood and likely to be a outlier. The density within a cluster also varies: more concentrated as closer to the center and looser at the boundary. To model the feature space comprehensively, features in each cluster are separated into *core features* and *boundary features* based on their $\mathscr{D}_k$ with threshold $\eta$. Therefore, the size of *core features* and *boundary features* are $n_i^C = \eta n_i$ and $n_i^B = (1 - \eta) n_i$ for class $i$, respectively.

In addition to the intra-cluster density, LUNA evaluate the inter-cluster density *second boundary features*. The *second boundary features* to a cluster are defined as the data from other clusters, but regard this cluster as their second-best choice based on the regular distance to cluster center. The above definitions are visualized in Figure 4.3, we denote the *core features*, *boundary features* and *second boundary features* for class $t$ as $\mathbb{N}_t^C$, $\mathbb{N}_t^B$, and $\mathbb{N}_t^{sB}$.

### 4.2.4 LUNA Factor

To access a testing sample's novelty, LUNA compares its density with respect to the core, boundary and second boundary areas to obtain similarities.

Let the feature of a testing sample be written as $\mathbf{p}$, then its *sub-local reachability density* is $\mathscr{D}_k(\mathbf{p})$. $\mathbb{N}^C$, $\mathbb{N}^B$, and $\mathbb{N}^{sB}$ are the core, boundary and second boundary feature sets of closest cluster to $\mathbf{p}$, respectively. The targeted cluster is denoted as $\mathbb{N}$. We compare $\mathscr{D}_k(\mathbf{p})$ with the average $\mathscr{D}$ of the three feature sets. Intuitively, if the density of $\mathbf{p}$ is close to the core area or boundary area, it is likely to be a inlier (i.e., belongs to a trained category). Otherwise, if the density of $\mathbf{p}$ is comparable to the second boundary area, it means the testing sample is not well-aligned with its

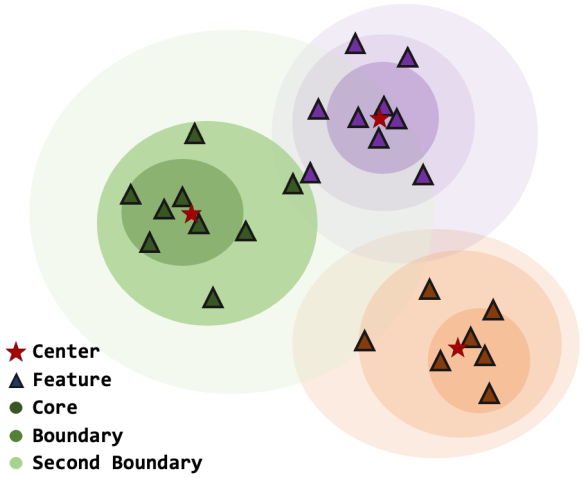



Figure 4.3: Illustration of our definitions in feature space. Three clusters are plotted in green, orange and purple respectively. Each cluster is divided into core and boundary area based on the sub-local reachability density. The second boundary covers features from other cluster that see the current cluster as second-rated choice. Darker color means larger density value. Best view in color.

predicted cluster and potentially novel. These comparisons are formulated as the following outlier factors, i.e., *Core Outlier Factor* (COF, $\mathscr{F}_C$), *Boundary Outlier Factor* (BOF, $\mathscr{F}_B$) and *Second Boundary Outlier Factor* (sBOF, $\mathscr{F}_{sB}$):

$$\begin{aligned}
\mathscr{F}_C(\mathbf{p}) &= \frac{\sum_i^{|\mathbb{N}^C|} \mathscr{D}_k(\mathbf{p})/\mathscr{D}_k(\mathbf{z}_i)}{|\mathbb{N}^C|}, \mathbf{z}_i \in \mathbb{N}^C. \\
\mathscr{F}_B(\mathbf{p}) &= \frac{\sum_i^{|\mathbb{N}^B|} \mathscr{D}_k(\mathbf{p})/\mathscr{D}_k(\mathbf{z}_i)}{|\mathbb{N}^B|}, \mathbf{z}_i \in \mathbb{N}^B. \\
\mathscr{F}_{sB}(\mathbf{p}) &= \frac{\sum_i^{|\mathbb{N}^{sB}|} \mathscr{D}_k(\mathbf{p})/\mathscr{D}_k(\mathbf{z}_i)}{|\mathbb{N}^{sB}|}, \mathbf{z}_i \in \mathbb{N}^{sB}.
\end{aligned} \tag{4.10}$$

Unlike LOF [8] that uses a fixed number of neighbors, LUNA uses a distribution-adaptive design: $k = \alpha|\mathbb{N}|$, where $\alpha \in (0, 1]$ is a parameter. In this case, small $k$ is chosen for tail classes and larger $k$ is chosen for head classes. Therefore, LUNA measure is more robust to features in the twilight zone. More experimental results are discussed in [9].

The overall LUNA factor $s(\mathbf{p})$ is a combination of density measurements and the network's classification output:

$$
\begin{aligned}
s(\mathbf{p}) =& \texttt{Min}(|1-\mathscr{F}_C(\mathbf{p})|, |1-\mathscr{F}_B(\mathbf{p})|) \\
&+ \left| 1 - \frac{1}{|\mathbb{N}^{sB}|} \sum_{i}^{\mathbb{N}^{sB}} \mathscr{F}_{sB}(p_{sb_i}) / \mathscr{F}_{sB}(\mathbf{p}) \right| + (1-\theta),
\end{aligned} \tag{4.11}
$$

In Equation 4.11, the first term measures how likely $\mathbf{p}$ is inside an existing cluster, i.e., the density is close to the core or boundary features' average density. As the density of $\mathbf{p}$, $\mathscr{D}_k(\mathbf{p})$, could be very large for inliers according to Equation 4.10, the second term compares the $\mathbf{p}$'s sBOF with the average sBOF over all second boundary features. In the last term, $\theta$ is the confidence for prediction from the classifier. Then Otsu's method [78] is applied on the entire testing set to find the threshold of novelty.

The LUNA factor is proved to be an effective novelty measurement in the OLTR setting and it is the state-of-the-art method on multiple large-scale benchmarks. Compared to the previous novelty detection or memory-based OLTR approaches, LUNA has two advantages: (1) improve closed-set long-tailed classification accuracy with weighted center loss, and (2) LUNA factor is a quantitative measurement with interpretable meanings (i.e., locations in the feature space).

## 4.3 Experimental Results

### 4.3.1 Dataset and Protocols

#### Dataset

The **ImageNet-LT** [70] dataset is re-sampled from a subset of the original ImageNet-2012 [20] following Pareto distribution. Extra 10 classes from ImageNet-2010 make up the open-set. There are 1000 classes for training, with 5 to 1280 images per class and 115.8K images in total.

The **Places-LT** [70] dataset is re-sampled from Place-2 dataset [126] for scene recognition. There are 69 new classes in Place-Extra69 used as open-set. It contains 184.5K images for 365 classes, with 5 to 4980 images per class.

Our proposed **Marine Species (MS)-LT** dataset is naturally long-tailed distributed (in Figure 4.4). There are 25.4K images for 106 marine species, with 5 to 1920 images per class. There are 25 classes for open-set. Table 4.2 shows the distribution and number of samples of MS-LT. The challenges are several classes share high inter-class similarity and some data of the same class

| Division | # of class | # of images |
|---|---|---|
| Many ($x > 100$) | 43 | 23.3K |
| Medium ($20 < x \leq 100$) | 32 | 1.7K |
| Few ($x \leq 20$) | 31 | 0.4K |
| Total (close-set) | 106 | 25.4K |
| Open-set | 25 | 0.4K |

Table 4.2: Statistics of our MS-LT dataset. Here, $x$ denotes the number of samples in the class.

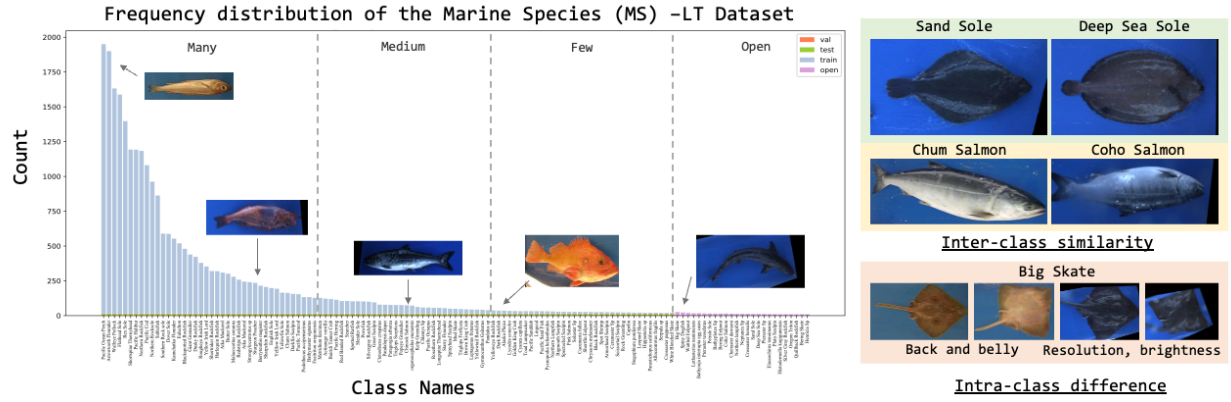



Figure 4.4: **Left**: the distribution of the proposed MS-LT dataset. There are three levels of frequency for the closed-set: many (counts $> 100$), medium ($20 <$ count $\leq 100$) and few (count $\leq 20$). The training set follows a long-tailed distribution, while the testing and validation sets are balanced following the configuration of other long-tailed datasets. **Right**: the challenging samples in the MS-LT. Some classes are similar in appearance, while some samples in the same class are different in orientation, resolution and lighting.

exhibit vast difference in appearances with different orientations or are collected in different years. The images were collected during National Marine Fisheries Service groundfish abundance surveys of the Gulf of Alaska (in 2015 and 2019) and the Aleutian Islands (in 2016). These image collections were elements of an ongoing effort to develop onboard imaging systems for fisheries monitoring, resulting in differences between the image collection systems. Captured fish were slid through enclosed camera chutes (2015 and 2019) or placed in an enclosed photo-booth (2016). These imaging systems are shown in Figure 4.5. Chutes used cool white (6000K) LED lighting. The 2016

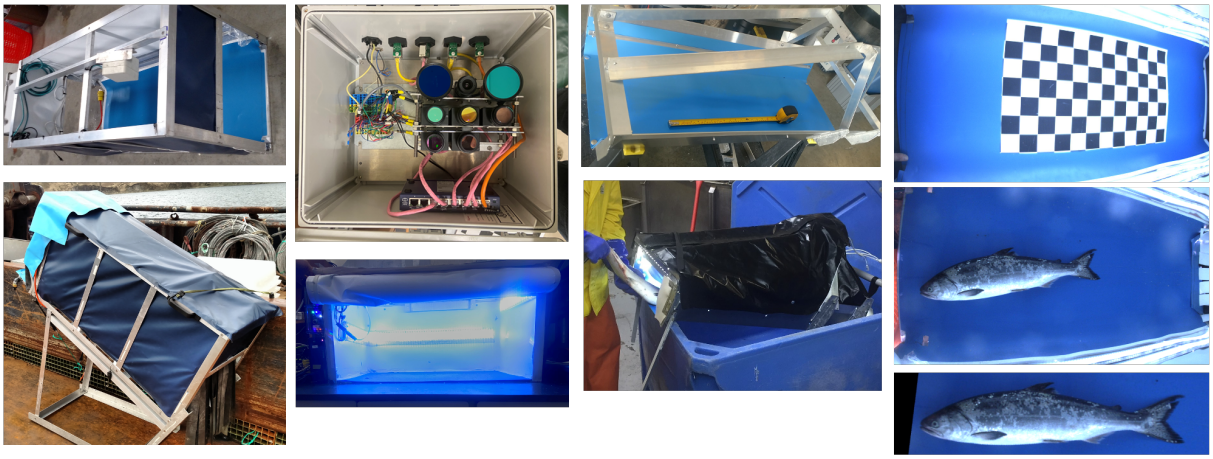
photo-booth, used for a trial of multi-spectral discrimination, was illuminated with a combination of narrow-frequency LEDs centered at a range of seven visible, infrared and ultraviolet frequencies. Single images, triggered by infrared light beam sensors, were collected with machine-vision cameras in 2015, while motion-triggered video clips were collected in 2019. While the photo-booth was equipped with seven machine-vision cameras, one filtered for each LED frequency, only images from an eighth, wide-frequency RGB camera were used for the current collection. The chute cameras both used oblique views, with the camera located at one end of the chute, while the photo-booth camera was centrally located above the fish location. While these differences likely caused difficulties for consistent species identification, we feel they represented some problems encountered when developing a general classifier for a range of image sources. The images and videos were first calibrated based on images of a checkerboard relocated to cover all areas of the imaging surface. Images were then rectified to achieve equal length per pixel across the image [7]. Fish identifications were recorded for each image or video clip by the biologists on the survey vessels. Finally, the fishes were cropped from the rectified images.

Example closed-set images in the MS-LT dataset are shown in Figure 4.6. In the closed training set, images vary yearly due to differences in the camera location, lighting condition and the way fishes move in the chute (single/group). For example, the Pacific Ocean Perch class (row-1, column-5) and the Sablefish class (row-2, column-6) have various brightness levels. In addition, some species have similar appearance, such as the Eulachon (row-2, column-2) and Pacific Herring (row-2, column-3). Besides, the fishes have different colors for the back and belly, like the Arrowtooth Flounder (row-1, column-1) and the Mud Skate (row-2, column-1). The inter-class similarity and intra-class difference make MS-LT a challenging fine-grained recognition dataset. Examples of the open-set images are shown in Figure 4.7.

The openness $O$ [5] of an open-set is defined as

$$O = 1 - \sqrt{\frac{2 \times T_{\text{train}}}{T_{\text{test}} + T_{\text{closed}}}}, \tag{4.12}$$

where $T_{\text{train}}$ and $T_{\text{test}}$ are the total number of classes in training and testing sets, respectively, and $T_{\text{closed}}$ is the number of closed-set categories for classification. In regular object recognition task, $T_{\text{test}}$ is identical to $T_{\text{closed}}$, thus the openness is 0. The more the novel classes, the higher the openness, the more difficult the task. The openness of ImageNet-LT, Places-LT and MS-LT is



(a) The interior (top) and external (bottom) view of the chute (2015).

(b) The multi-spectrum cameras (top) and external view (bottom) of the booth (2016).

(c) The interior (top) and external (bottom) view of the chute (2019).

(d) An example of the checkerboard image (top), captured image (middle), the calibrated and cropped image (bottom) in our dataset (2015).

Figure 4.5: (a)-(c) The experiment setting of the chute system in 2015, 2016 and 2019, respectively. (d) An example for the checkerboard image, captured image, and final image in 2015.

0.005, 0.085, and 0.331, respectively.

### Evaluation Metrics

Following the prior works, we report the top-1 accuracy for closed-set classification, denoted as $\text{Acc}_{\text{close}}$. Specifically, the classes are divided into many-shot, medium-shot and few-shot based on the training size for long-tailed dataset [70], i.e., classes with more than 100 samples are many-shot and with less then 20 samples are few-shot. Accuracy on each frequency split is also compared.

As for open-set (including both closed-set testing data and novel data), the open-set accuracy $\text{Acc}_{\text{open}}$ and F-score are reported. [5] defines the true positive (TP) of novelty detection as closed-set samples that are correctly classified to the ground truth labels, false positive (FP) means closed-set samples that are falsely classified (to another closed-set category or to open-set), and false negative (FN) is the open-set samples wrongly recognized as closed. Then

$$\text{Acc}_{\text{open}} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{4.13}$$

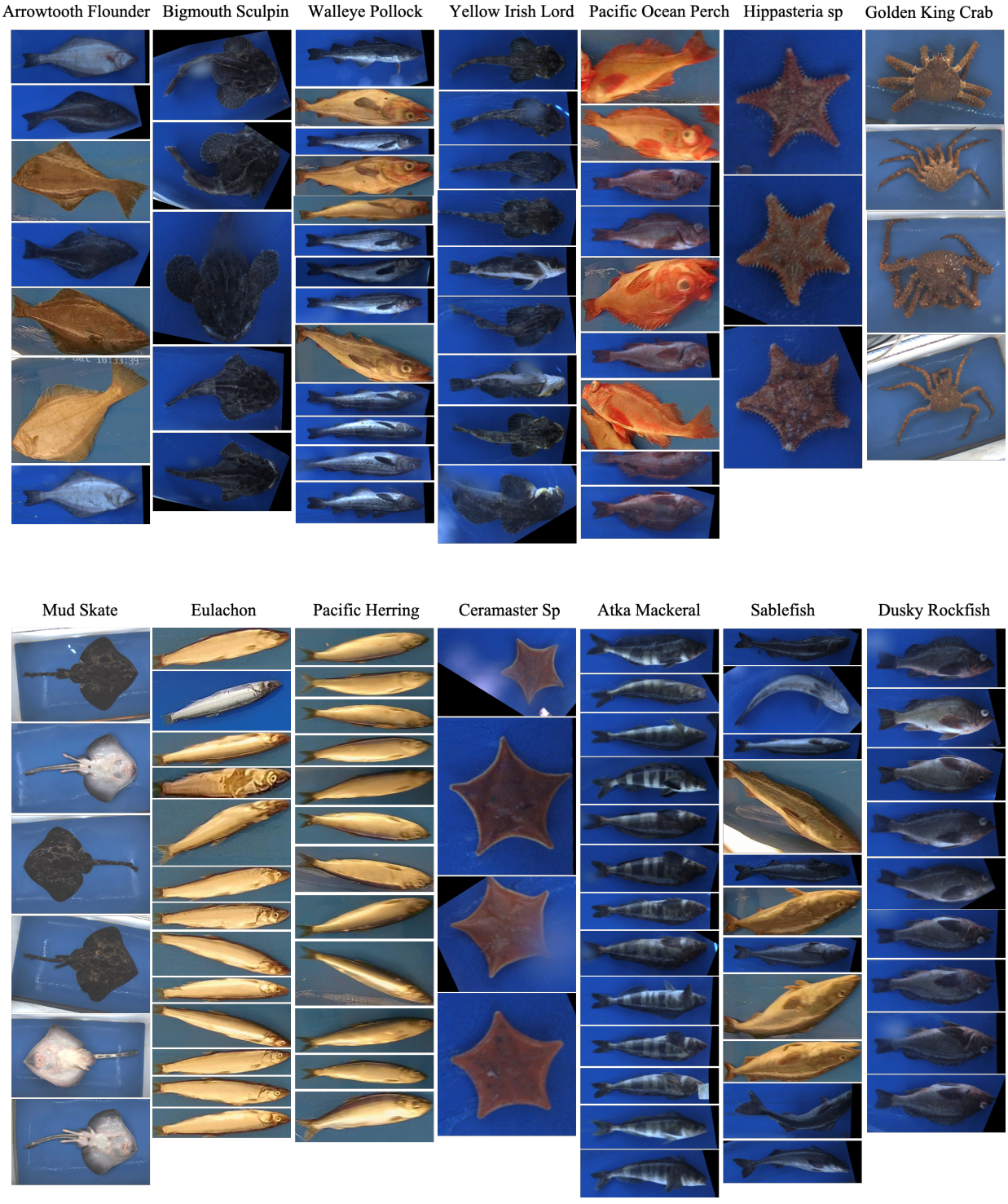



Figure 4.6: Example images in the closed-set of MS-LT.

Figure 4.7: Example images in the open-set of MS-LT.

$$\text{F-score} = \frac{\text{TP}}{\text{TP} + 0.5 \times (\text{FP} + \text{FN})}. \tag{4.14}$$

**Implementation Details**

The training samples are re-scaled by its shorter side and then resized to $224 \times 224$ with random crop and horizontal flip as data augmentation. We use ResNet-10, ResNet-152 and ResNet-32 as the backbone for ImageNet-LT, Place-LT and MS-LT, respectively. Following the two-stage decoupling training scheme [47], we first train the model with the original imbalanced dataset by stochastic gradient descent (SGD) with the momentum of 0.9 and weight decay of $2 \times 10^{-4}$ in minibatch size of 128 for 180 epochs; then continue training the model with progressively-balanced re-sampling [47] with learning rate 0.05 for an extra 50 epochs. The wcenter loss is applied only at the second stage. There is no extra parameter to weigh different loss components.

### 4.3.2 Performance

#### Performance on Public Benchmarks

Following the baseline experiments in the OLTR network [70], we report the performance of the proposed method in the open-set and closed-set settings, respectively. The base model denotes the plain ResNet [38] without any adaptation on long-tailed or open-set configurations. Lifted loss [77], focal loss [64] and range loss [121] are metric learning techniques to pull features of the same categories closer, where the range loss is designed for the long-tailed face recognition task. OpenMax [5] is a statistical fitting method to predict the probability of novelty in a similar manner as the SoftMax. FSLwF [29] is a few-shot learning algorithm. OLTR [70] is the first work to formally define the OLTR problem and propose a network with visual memory and weight regularization to transfer knowledge from heads to tails as well as separate knowns and unknowns. IEM [130] designs region self-attention to improve the quality of memorized features.

Table 4.3 shows the performance on the ImageNet-LT, Places-LT and MS-LT, respectively. With our emphasis on long-tailed learning, our model outperforms the OLTR network by 3.4%, 1.5%, and 6.8%in overall accuracy of multi-class recognition (closed-set). It also improves the F-measure by 5.4%, 0.5%, and 5.4%, respectively. Our advantages lie in the improvements in the many and medium segments. We think it is critical to balance the heads and tails properly. Applying methods for the balanced set, like the base model, yields promising performance on the sample-rich categories but performs poorly on the tails. Another extreme attempt is to use a few-shot learning scheme, like FSLwF in the tables, to promote the tails' performance, but this is not advantageous in open-set testing. With various metric learning losses, such as the lifted loss, focal loss and range loss, the performance under the open-set setting is comparable with that of the closed-set. This supports our argument that representation learning is an effective tool for transferring the closed-set knowledge to applications of open-set. Therefore, we use input re-balancing and progressively adapt the classifier to handle the long-tailed problem. With the frequency-sensitive wcenter loss, the feature space is adequately organized and separable.

| Dataset | Model | Closed-set | | | | Open-set | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Many | Medium | Few | Overall | Many | Medium | Few | F-measure |
| ImageNet-LT (ResNet-10) | Base Model [38] | 40.9 | 10.7 | 0.4 | 20.9 | 40.1 | 10.4 | 0.4 | 0.295 |
| | Lifted Loss [77] | 35.8 | 30.4 | 17.9 | 30.8 | 34.8 | 29.3 | 17.4 | 0.374 |
| | Focal Loss [64] | 36.4 | 29.9 | 16.0 | 30.5 | 35.7 | 29.3 | 15.6 | 0.371 |
| | Range Loss [121] | 35.8 | 30.3 | 17.6 | 30.7 | 34.7 | 29.4 | 17.2 | 0.373 |
| | OpenMax [5] | - | - | - | - | 35.8 | 30.3 | 17.6 | 0.368 |
| | FSLwF [29] | 40.9 | 22.1 | 15.0 | 28.4 | 40.8 | 21.7 | 14.5 | 0.347 |
| | OLTR [70] | 43.2 | 35.1 | 18.5 | 35.6 | 41.9 | 33.9 | 17.4 | 0.474 |
| | IEM [130] | 48.9 | 44.0 | 24.4 | 43.2 | 46.1 | 42.3 | 20.1 | 0.525 |
| | **LUNA (Ours)** | **51.8** | **48.6** | **26.2** | **46.6** | **48.2** | **44.7** | **23.6** | **0.579** |
| Places-LT (ResNet-152) | Base Model [38] | 45.9 | 22.4 | 0.4 | 27.2 | 45.9 | 22.4 | 0.4 | 0.366 |
| | Lifted Loss [77] | 41.1 | 35.4 | 24 | 35.2 | 41.0 | 35.2 | 23.8 | 0.459 |
| | Focal Loss [64] | 41.1 | 34.8 | 22.4 | 34.6 | 41.0 | 34.8 | 22.3 | 0.453 |
| | Range Loss [121] | 41.1 | 35.4 | 23.2 | 35.1 | 41.0 | 35.3 | 23.1 | 0.457 |
| | OpenMax [5] | - | - | - | - | 41.1 | 35.4 | 23.2 | 0.458 |
| | FSLwF [29] | 43.9 | 29.9 | 29.5 | 34.9 | 38.1 | 19.5 | 14.8 | 0.375 |
| | OLTR [70] | 44.7 | 37 | 25.3 | 35.9 | 44.6 | 36.8 | 25.2 | 0.464 |
| | IEM [130] | 46.8 | 39.2 | 28.0 | 39.7 | **48.8** | **42.4** | 28.9 | 0.486 |
| | **LUNA (Ours)** | **48.7** | **42.4** | **30.2** | **42.1** | 48.1 | 41.6 | **29.0** | **0.491** |
| MS-LT (ResNet-32) | Base Model [38] | 56.1 | 35.1 | 8.0 | 35.7 | 56.1 | 35.1 | 11.4 | 0.537 |
| | Lifted Loss [77] | 53.2 | 42.3 | 12.6 | 38.0 | 53.0 | 42.2 | 12.4 | 0.549 |
| | Focal Loss [64] | 57.3 | 44.6 | 18.5 | 42.1 | 57.0 | 42.8 | 15.4 | 0.576 |
| | Range Loss [121] | 55.8 | 43.8 | 15.7 | 40.5 | 55.8 | 43.6 | 15.6 | 0.575 |
| | OpenMax [5] | - | - | - | - | 54.2 | 44.9 | 12.8 | 0.564 |
| | OLTR [70] | 57.8 | 49.8 | 28.6 | 46.8 | 56.7 | 45.3 | 23.6 | 0.603 |
| | **LUNA (Ours)** | **61.2** | **56.6** | **34.6** | **52.0** | **60.4** | **51.8** | **30.4** | **0.657** |

Table 4.3: OLTR performance of top-1 accuracy on the ImageNet-LT, Places-LT, and MS-LT datasets. Best results are marked in **bold**.

### Discussions on the Open-set Performance

F-measure evaluates both the classification accuracy and novelty detection recall rate. However, the novelty detection does not make a great difference if the open-set is small, i.e., small false

negative value in Equation 4.14. The novelty detection accuracy, which is the portion of open-set being correctly identified, is a better metric to purely evaluate the model's ability of identifying the new classes. In Table 4.3, we report closed-set overall accuracy (false positives are the closed-set samples that are incorrectly classified as another closed-set class or open-set), open-set accuracy (false positives are the open-set samples that are misclassified as closed-set), and the F-measure. Comparing to the prior leading method, our LUNA achieves better performance on the novel classes, without sacrificing the long-tailed classification accuracy. The results also show MS-LT is a challenging OSR dataset due to its high openness and fine-grained properties.

**Visualize the Feature Space**

Figure 4.8 shows the t-SNE [71] visualization of the MS-LT dataset. With the wcenter loss, it is observed that the intra-class distances are reduced and each class is nicely clustered together, especially for the tails. The more concentrated they are, the more precise for LUNA factor estimation. The open-set samples are spread over empty region of the feature space rather than inside the clusters. Therefore, wcenter loss significantly benefits open-set detection.

**Visualize the LUNA Components**

Figure 4.9 is the visualization of COF, BOF, sBOF and classification confidence of each sample in the MS-LT dataset. The COF and BOF of known classes are usually around 1, meaning they are close to the neighbors inside the cluster. They also have large sBOF as they have higher local density than the second boundary group. As for the novel points, the COF and BOF are smaller than 1 while the sBOF is close to 1, indicating they are far away from the cluster center like they are in another cluster. Besides, the network's outputs of maximum confidence, which is included in LUNA as well, are high for known classes and unstable for the novel classes.

**LOF and LUNA**

To compare the novelty detection of LOF and LUNA, we use $K = 5$ for LOF, which is the size of the minimum cluster. As shown in Figure 4.10, the novel samples are more distinctive with LUNA. The F-measure of LOF is 0.357 while the LUNA is 0.657. The reason is LOF selects a constant

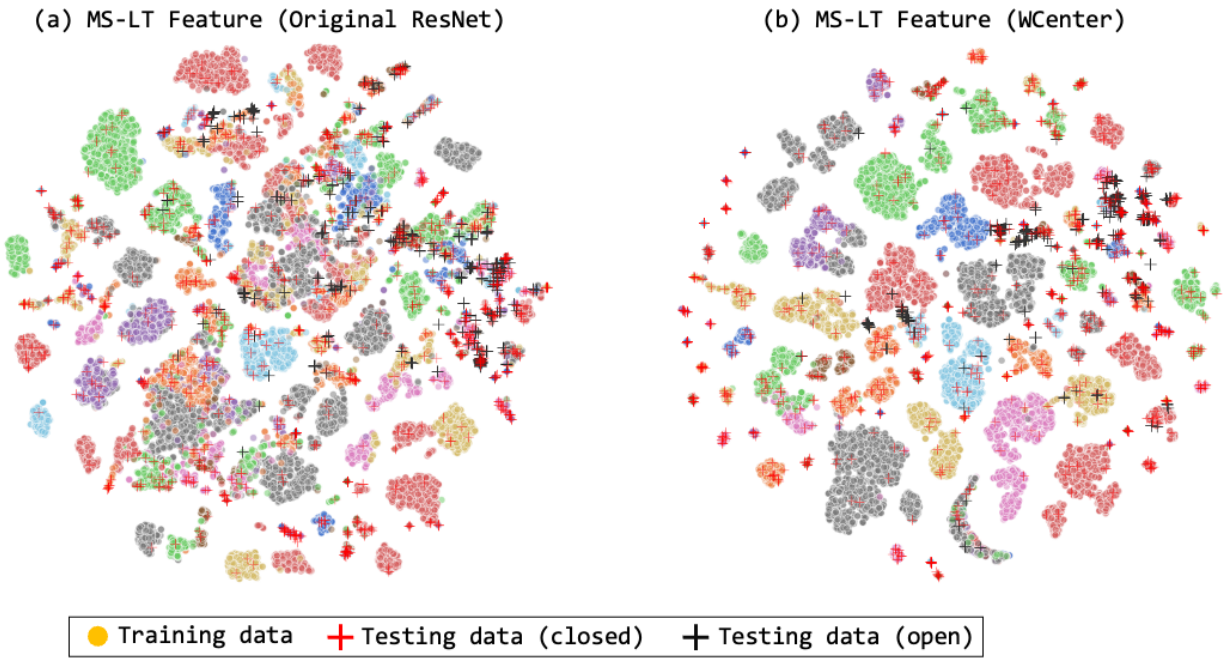



Figure 4.8: t-SNE visualization of the MS-LT dataset. Left: the original model; right: model with wcenter loss. The class of training samples (dots) are marked in different colors. The testing set (closed) and open set are marked in red and black crosses, respectively.

number of neighbors over the whole dataset for each testing sample, regardless of the clustering size or its potential category. On the other hand, LUNA uses variable sizes of neighbors that are adaptive to the clusters' sizes and different regions.

## 4.4 Analysis and Discussions

### 4.4.1 Effectiveness of Wcenter Loss

The role of wcenter loss is in two aspects: (1) re-weighting on the minority classes to benefit long-tailed recognition; (2) concentrating clusters in the feature space for outlier detection. Therefore, we compare the following schemes to show its effectiveness. Denoted $\lambda_j$ as the weight of class $j$, which is a function of its frequency $\widetilde{n_j}$.

(a) None: $\lambda_j = 0$, which is training without center loss.

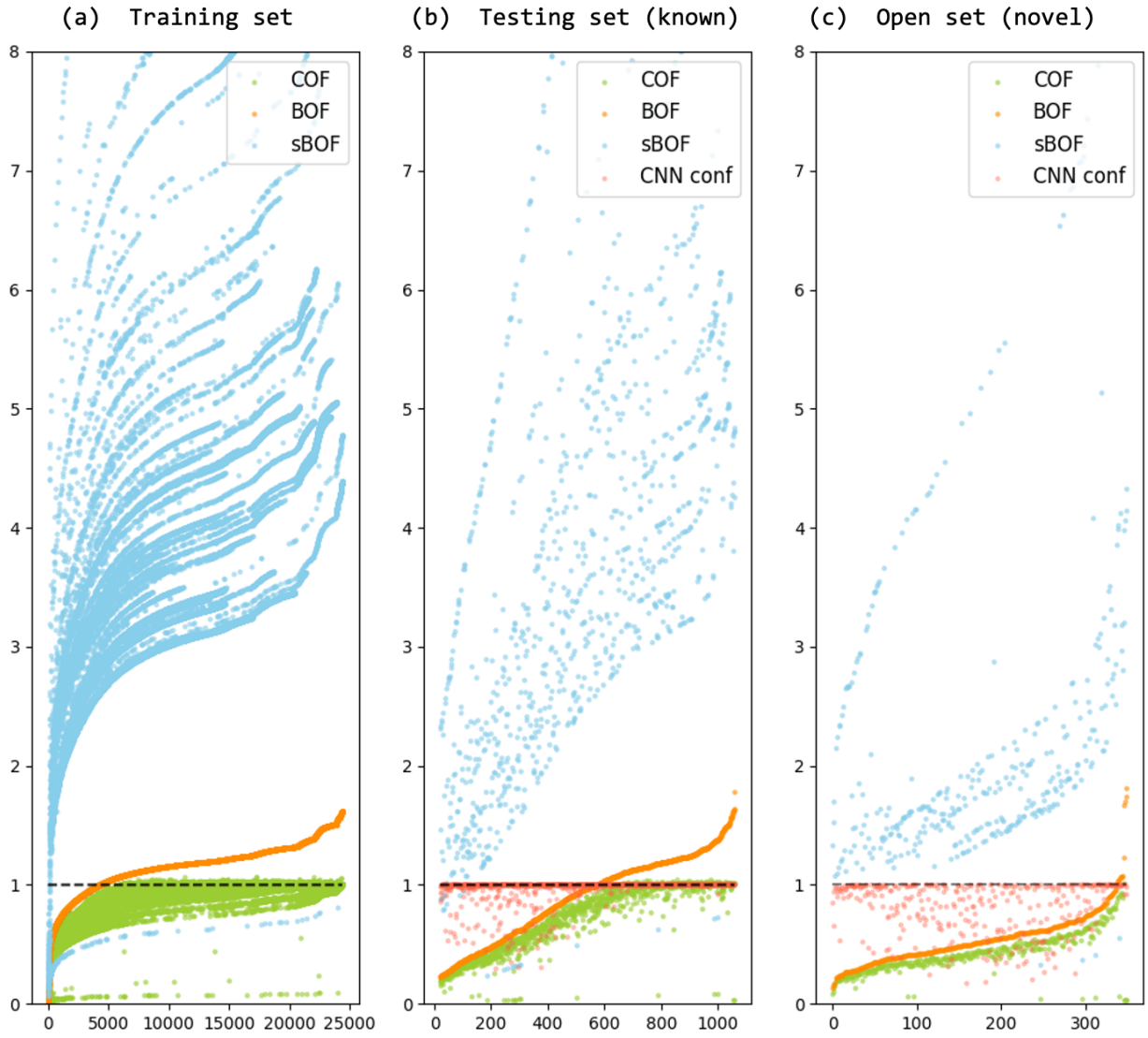



Figure 4.9: Visualization of COF, BOF, sBOF and network confidence. The $x$-axis is the data point index, which is independent of each other. They are ordered by the value of BOF to show the trend. The black dash, $y = 1$, indicates the location of the point in the feature space. For example, a sample with COF near 1 is likely to be a core point of seen classes.

(b) Center: $\lambda_j = 1$, which is the vanilla center loss.

(c) Same: $\lambda_j = 1/\widetilde{n_j}$, which is the same as the frequency distribution.

(d) Inverse: $\lambda_j = \frac{\widetilde{n_j}}{\max_c\{\widetilde{n_c}\}}$, which is the inverse of the frequency distribution.

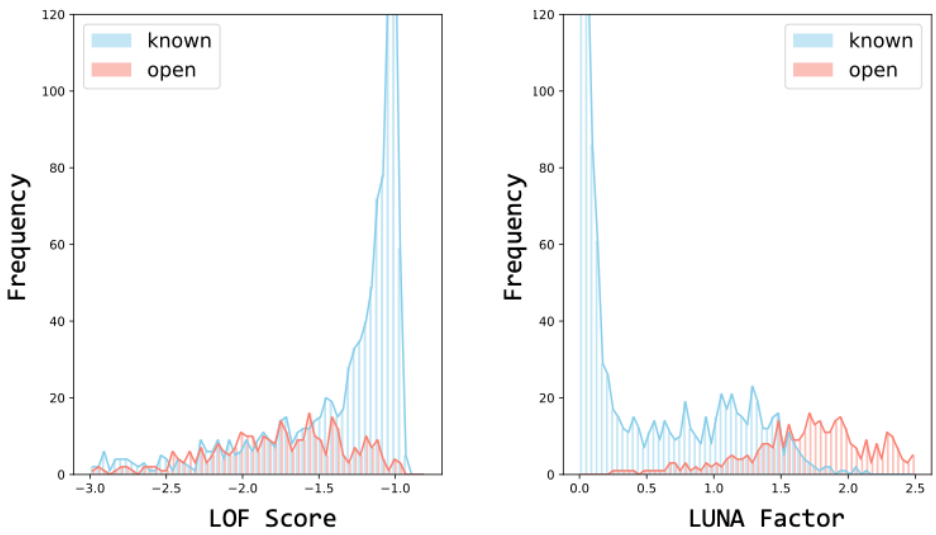



Figure 4.10: Comparison of the LOF score and LUNA score on MS-LT.

(e) Wcenter: $\lambda_j = \frac{\widetilde{n_j}}{\max_c\{\widetilde{n_c}\}} + 1$

The results, as shown in Table 4.4 are reported on MS-LT under the open-set setting. The result indicates bias on the feature domain affects the classification accuracy proportionally. The inverse loss and wcenter loss weight more on the tails, thus have more gains on the few-shot split. However, they sacrifice the accuracy on heads. Wcenter loss with a scaling term preserves the performance on head relatively. On the other hand, center loss and wcenter loss emphasis the clustering requirement, resulting in more desirable open-set detection performance. The result suggests metric learning is the key to solve open-set recognition problem with long-tailed training data: it is capable of helping imbalanced classification and automates feature selection in high dimension space for open-set recognition.

### 4.4.2 Effectiveness of LUNA

LUNA measures the relative density of the testing sample with respect to the densities of core, boundary and second boundary in each cluster, as well as the network confidence. In this section, we show the experimental results by removing each component in Equation 4.11. To simplify, the three components are represented with its most important metric, i.e.,, $\mathscr{F}_C, \mathscr{F}_B$, $\mathscr{F}_{sB}$ and $\theta$,

| Weight | Many | Medium | Few | Overall | F-measure |
|---|---|---|---|---|---|
| None | 62.4 | 49.8 | 29.4 | 48.8 | 0.632 |
| Center | 60.8 | 54.2 | 31.9 | 50.4 | 0.651 |
| Same | 63.5 | 55.8 | 28.4 | 50.8 | 0.608 |
| Inverse | 57.6 | 57.0 | 35.2 | 50.9 | 0.614 |
| Wcenter | 61.2 | 56.6 | 34.6 | 52.0 | 0.657 |

Table 4.4: Ablation study on weighting schemes on MS-LT. Many-/medium-/few-shot and overall accuracy are reported in closed-set; F-measurement is under open-set setting.

respectively. The open-set performance on MS-LT is shown in Table 4.5.

The result indicates all the components in LUNA are necessary and effective. The first term evaluates the density regarding the core and boundary samples (inliers), which is shown to separate the majority from the novel samples. Removing it causes misclassification of the many-shot split. The second term and third term are responsible for separating the minority classes from the novel ones, as the clusters of minority classes are not as concentrated as the majorities', relaxing the metrics to the second boundary is beneficial.

Existing open-set recognition methods rely on the classification confidence (or the classifier output logits), which do not work well on long-tailed dataset as they do in balanced sets. Figure 4.9 (b) shows that confidence is not always high for closed-set samples, and it is the least important one comparing to the $\mathscr{F}_C, \mathscr{F}_B$ and $\mathscr{F}_{sB}$. Utilizing the sample's property itself (confidence), and the difference compared to the nearby acquaintance (trained samples) is more stable and interpretable.

| Component | Many | Medium | Few | F-measure |
|---|---|---|---|---|
| **LUNA** | **60.4** | **51.8** | **30.4** | **0.657** |
| $-\mathscr{F}_C, \mathscr{F}_B$ | 54.5 | 46.8 | 25.1 | 0.607 |
| $-\mathscr{F}_{sB}$ | 57.8 | 48.2 | 23.6 | 0.620 |
| $-\theta$ | 59.2 | 48.4 | 27.4 | 0.636 |

Table 4.5: Ablation study on each component of LUNA on MS-LT.

### 4.4.3 Sensitivity of the Hyper-parameter

We conduct experiments on the potion of each cluster ($\eta$) that is selected as core samples; results are shown in Table 4.6. As a joint evaluation of the density of multiple well-defined sample groups, the proposed LUNA is very robust in OLTR task and not sensitive to the selection of $\eta$. From our own test, we would recommend the $\eta$ value from 0.2 to 0.5.

| $\eta$ | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.8 |
|---|---|---|---|---|---|---|
| F-measure | 0.654 | 0.657 | 0.655 | 0.648 | 0.635 | 0.622 |

Table 4.6: Ablation study on the potion of core samples on MS-LT.

# Chapter 5

# LUNA+: IMPROVING OPEN-SET LONG-TAILED RECOGNITION

## 5.1 Overall Framework

Inspired by a recent LTR research on the separability in the feature space [46, 60], we see the following deficiencies in LUNA:

- **Uniformity**: the center of each class (optimization targets) is randomly initialized in Equation 4.3, and are updated with the learned features. There is no uniformity constrains applied on the centers. Thus the feature space is not guaranteed to be balanced.

- **Interpretability**: supposing there is a set of uniformly distributed centers and are assigned to the classes. However, due to the random ordering of classes, the semantic closeness are not preserved in feature space. This may hurt the detection performance or cause ambiguities in novelty detection.

- **Diversity**: Equation 4.11 use a empirical ratio to divide each cluster into core and boundary splits. The same parameters might not fit all classes because of the difference in cluster size, density and the relationship to neighboring clusters.

To this end, we hereby propose an improved metric learning approach on top of the original LUNA framework, as shown in Figure 5.1. There are two isolated training stages in the workflow, one for center generation and the other one is to learn features and closed-set classifier simultaneously, which will be addressed in Section 5.2.1 and 5.2.2, respectively. Then we introduce the new LUNA factor measurement, called LUNA+, in Section 5.2.3. LUNA+ contains four major revisions to mitigate the mentioned problems:

- **Optimization target:** utilize a center generation process to obtain a set of targeted centers that are uniformly distributed, and then assign the generated centers to classes as the guiding initialization during feature learning to preserve semantic relationships.

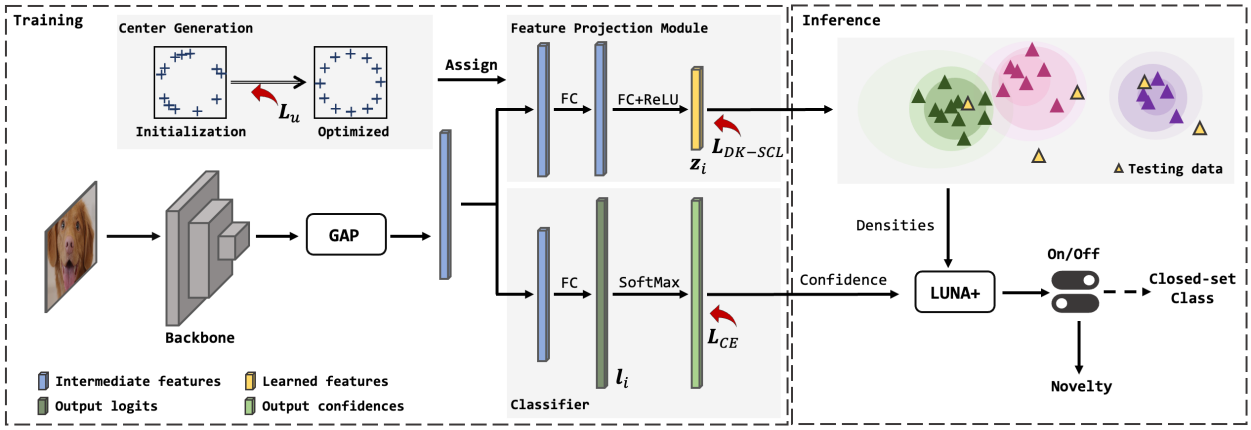



Figure 5.1: Illustration of the proposed workflow. There are two training stages: center generation and joint feature-classifier learning. In the second one, backbone features are pooled by a Global Average Pooling (GAP) layer, and sent to the feature projection module and classifier. In testing, the testing sample's feature is evaluated w.r.t. the training set's features using LUNA+ factor to access novelty. For closed-set samples, we trust the classifier's output.

- **Network architecture:** decouple the representation learning and classifier training by adding a feature projection module. The purpose of using a separate feature branch is to avoid the trade-off effects between classification and metric learning tasks.

- **Metric learning:** use supervised contrastive loss instead of center loss, which is more stable and leads to better novelty detection performance.

- **LUNA factor:** replace the network confidence term in LUNA factor by an uncertainty measurement to bypass the overconfidence issue (?? not mentioned the reason why previously in the beginning of this chapter??).

## 5.2 Methdology

### 5.2.1 Pre-generated Centers and Online Assignment

As the optimization targets for feature learning, the distribution of the centers are critical, especially for imbalanced training data. The reason lies in the discrepancies of each class, in the aspects of cluster size, intra-class similarity, and how they are initialized. Typical center-based metric learning

methods, e.g., center loss [110], randomly initialize the centers and then updates them as the deep features change. However, as larger classes occupy more feature space than the smaller ones, the clusters (centers) could hardly be distributed uniformly. An intuitive idea is to use pre-computed centers that are evenly distributed, and force training features to be close to their class centers while keeping the centers fixed. The problem of positioning vectors in a hypersphere is well-studied. It is usually tackled by minimizing pairwise distance with respect to some kernel functions. Following targeted supervised contrastive (TSC) learning [60] and [104] for LTR, we choose the Radial Basis Function (RBF) kernel to define the uniformity loss as:

$$L_{\mathrm{u}} = \frac{1}{T}\sum_{t}^{T} \texttt{log} \sum_{r,r\neq t}^{T} e^{(\mathbf{c}_r \cdot \mathbf{c}_t^T)\cdot\tau}, \tag{5.1}$$

where $\tau$ is a predefined and fixed temperature parameter. The cosine distance between each distinct center pair $c_r, c_t^T$ is minimized with $L_u$. Note that simply minimizing the pairwise cosine distance (without using a kernel function) or Euclidean distance could potentially reach any distribution with zero mean, but unnecessary to be a uniform distribution. In-depths mathematical derivations are discussed in [15, 3, 104].

Once the centers $\mathbf{C} = \{\mathbf{c}_t \in \mathbb{R}^d\}_{t=1:T}$ are obtained, another question is how to assign them to the classes. Since only the number of closed-set classes and feature dimension are used in Equation 5.1, $\mathbf{C}$ does not have semantic meanings. A consensus in metric learning is that images look alike are usually also closer in feature space. Therefore, the ideal assignment is based on semantic relevancy. However, it is hard to find a criteria that jointly models image similarity from both semantics and appearance aspects, and it is even harder in multi-task training (i.e., classification and feature learning).

TSC [107] propose to use the Hungarian Algorithm[57] to match the learned feature centers $\hat{\mathbf{C}} = \{\hat{\mathbf{c}}_t \in \mathbb{R}^d\}_{t=1:T}$ to target $\mathbf{C}$ with their distances used as costs. Specifically, $\hat{\mathbf{C}}$ is the running average for each epoch (i.e., over the entire training set) and the assignment is performed for every mini-batch. We follow the same scheme.

### 5.2.2 Dynamic-K Supervised Contrastive Loss

Observations in K-positive contrastive loss (KCL) [46] reveal that contrastive loss can obtain more balanced feature space and show strong generalizability. Therefore, as a variant of supervised

contrastive loss (SupCon) [49], KCL empirically keeps the number of positive instances equal ($K$) instead of using all samples from the same classes as positive in SupCon. TSC [107] further adds the generated centers as samples on top of KCL. However, in both KCL and TSC, $K$ is dataset-specific and needs to be carefully tuned with other hyper-parameters. As the experiments shown in [46], the accuracy significantly fluctuates as $K$ varies. Moreover, all classes sharing the same $K$ is regardless of the difference in their distribution, even though this helps to balance the feature space, it is easy to overfit or underfit certain classes.

To make the loss distribution-sensitive and alleviate the effort in parameter tuning, we propose a dynamic-K supervised contrastive loss (DK-SCL) that automatically chooses the value of $K$ per distribution. Note that various re-sampling recipes have been used in long-tailed datasets, meaning the sampling probability might change for every iteration. Therefore, the dynamic selection of $K$ not only fits the original data distribution, but can also apply to imbalanced re-sampled data.

More specifically, for each minibatch, following the common settings of supervised contrastive learning, we obtain two augmented copies of the original images. The extracted pairwise features are denoted as $\mathbf{Z}'$ and $\mathbf{Z}''$, respectively. The precomputed centers (see Eq. 5.1 are also included in the entire feature set $\mathbf{V} = \mathbf{Z}' \cup \mathbf{Z}'' \cup \mathbf{C}$. Taking a sample as *anchor*, its *cardinality* is defined as the number of samples from the same class in that minibatch, i.e., the number of positive samples. We define the dynamic-$K$ value as the average cardinality over that batch. Let the set of positive samples be denoted as $\mathbf{P}$, then $K = |\mathbf{P}|$. The DK-SCL can thus be formulated as

$$L_{\text{DK-SCL}} = -\frac{1}{|\mathbf{V}|}\sum_{i=1}^{|\mathbf{V}|}\frac{1}{K+1}\sum_{\mathbf{z}_j^+\in\mathbf{P}}\texttt{log}\frac{e^{(\mathbf{z}_i^T\cdot\mathbf{z}_j^+)\cdot\tau}}{\sum\limits_{\mathbf{z}_j\in\mathbf{V}\setminus\mathbf{P}} e^{(\mathbf{z}_i^T\cdot\mathbf{z}_j)\cdot\tau}}. \tag{5.2}$$

Besides, as suggested in [49], decoupling features for the classifier and visual representation can benefit both tasks. Therefore, we use two extra fully connected (`FC`) layers after the backbone's features to project the features to another high-dimensional space, called Feature Projection Module (FPM). The learned features $\mathbf{Z}$ are normalized before sent to DK-SCL for training or LUNA+ factor measurement for novelty detection.

### 5.2.3 LUNA+ Factor for Novelty Measurement

In the original LUNA factor, the potion of core features in a cluster is calculated from an empirical parameter $\eta$ and it is consistent for all classes. However, the clusters' intra-class distances are not equal, i.e., the optimal $\eta$ varies for every class. Therefore, we use Otsu's method [78] to find the cut-off threshold of boundary and core features within each cluster:

$$\eta_t = \texttt{Otsu}(\{\mathscr{D}_k(\mathbf{z}_i)\}_{y_i=t}), \tag{5.3}$$

where $\{\mathscr{D}_k(\mathbf{z}_i)\}_{y_i=t}$ represents the *sub-local reachability density* of all training features of class $t$. Then the samples with their densities above $\eta_t$ are core features. Core point and boundary point selection for Tiny-ImageNet-LT dataset is visualized in Figure 5.2. The rest parts of LUNA factor calculations are unchanged.

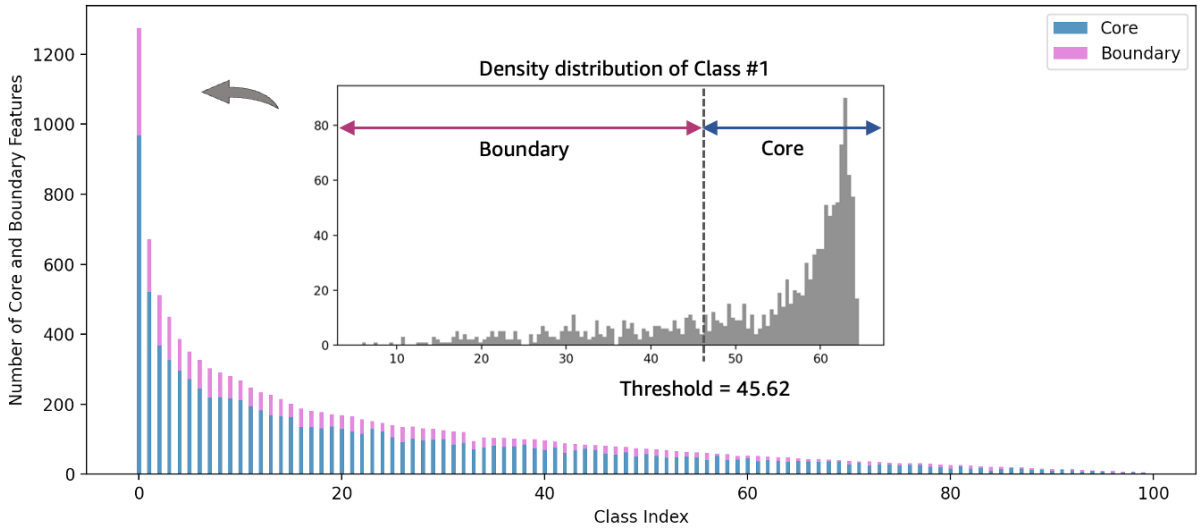



Figure 5.2: Visualization of core/boundary feature separation for Tiny-ImageNet-LT dataset. The zoom-in figure is a histogram on sub-local reachability density distribution of the largest class.

## 5.3 Experimental Results

### 5.3.1 Datasets

The proposed method is evaluated on three public OLTR benchmarks.

**ImageNet-LT** [70] is the long-tailed version of original ImageNet-2012 [55] classification dataset by re-sampling (Pareto distribution, power value $\alpha$ is 6). ImageNet-LT contains 115.8k images from 1000 closed-set categories, with a maximum of 1280 images per class and a minimum of 5 images per class. 10 extra classes in ImageNet-2010 form the open-set, adding another 18k images. Its imbalance factor is 256 and openness is 0.0025.

**Tiny-ImageNet-LT** is a down-scaled version created from ImageNet-LT by us to get higher openness. The closed-set is trimmed by randomly selecting 100 closed-set classes. We purposely keep the largest and smallest class to preserve the same imbalance factor. The open-set is also downsized 10 times, containing 1.8k images from the same 10 open classes as in ImageNet-LT (i.e., we do not use the dropped closed-set classes to expand the open-set). The openness of Tiny-ImageNet-LT is 0.024.

**Marine-Species-LT version 2 (MS-LT-v2)** is a natural marine species (mainly fish species) identification dataset. In MS-LT-v1 [9], there are 25.4k images for 106 species, with 1920 to 5 images per class. 0.4k images from 25 classes are viewed as open-set. MS-LT is the first naturally-collected OLTR dataset, and is challenging in representation learning because of the fine-grained properties and high openness. MS-LT-v2 merges 6 classes that are annotated as sub classes in MS-LT-v1, and increases the number of testing images per class from 10 to 40. There are 17.5k images for 62 closed-set classes, including 29 many-shot, 19 medium-shot and 14 few-shot classes. The largest class has 1057 training images and the smallest only has 4. Its imbalance factor is 264 and openness is 0.14. The number of novel classes increases to 54 (1.3k images). MS-LT-v2 provides more precise ground truth labels and is designed to evaluate the open-set detection issue given the large openness.

### 5.3.2 Experimental Setups

We use the same backbone as the competing methods for fair comparison, i.e., ResNet-10 for all the three datasets. The training images are augmented using AutoAug [16], then resized by the shorter side to $256 \times 256$ and randomly cropped to $224 \times 224$. We also apply input Mixup [120] ($\alpha$=1 in the Beta distribution) simultaneously.

For center generation with uniformity loss in Equation 5.1, we use SGD optimizer with a weight

decay of 1e-4 and a learning rate of 0.005. We train 400K iterations for 100 and 62 centers, and 2.4M for 1000 centers. Centers are of 64, 256 and 512 dimensions.

The backbone, feature projection module and classifier are jointly trained end-to-end, unlike the previous OLTR methods that use multiple training stages. For ImageNet-LT, our model is trained for 300 epochs with SGD optimizer and batch size 512. The initial learning rate is 0.2 and decreases by 0.1 at the 200th and the 260th epoch, respectively. Tiny-ImageNet-LT is trained with the same number of epochs and learning rate scheduler. Differently, the batch size is 256 and initial learning rate is 0.1. MS-LT-v2 is trained for 160 epochs with batch size 256. The learning rate starts from 0.1 and decreases by 0.1 at the 120th and the 140th epoch, respectively.

We use a progressively-balanced re-sampling scheme[47], where the sampling probability of class $t$ at epoch $e$, $p_t(e)^{\mathrm{PB}}$, is

$$\begin{aligned} p_t(e)^{\mathrm{PB}} &= (1-\frac{e}{E})p_t^{\mathrm{IB}} + \frac{e}{E}p_t^{\mathrm{CB}} \\ &= (1-\frac{e}{E})\frac{n_t}{\sum_r^T n_r} + \frac{e}{E}\frac{1}{T}, \end{aligned} \tag{5.4}$$

where $E$ denotes the total number of epochs,$p_t^{\mathrm{IB}}$ represents instance-balanced re-sampling (i.e., random sampling w.r.t. the training data distribution) and $p_t^{\mathrm{CB}}$ represents class-balanced re-sampling (i.e., each category has equal probability). Such a sampling approach enables the model to be trained in a single stage, which is more efficient than multi-stage training in [47, 70, 130].

### 5.3.3 Benchmark Performance

We first compare the purposed LUNA+ approach with several state-of-the-art competing methods. Base model [38] is the plain ResNet trained with cross-entropy loss. Lifted loss [77], focal loss [64] and range loss [121] are re-weighting methods to handle the data imbalance issue, for example, range loss is proposed for long-tailed face recognition task. Since the above methods are designed for closed-set recognition, following [70], we use 0.1 as threshold for novelty detection, i.e., samples with classification confidence below 0.1 are considered novel. OpenMax [5] is a well-known method for multi-class novelty detection by fitting the output logits to Weibull distribution. We replace the `SoftMax` layer in the base model with the OpenMax module. OLTR [70] and IEM [130] are two memory-based metric learning methods that detect novel data by feature distance. We also compare with LUNA [9] introduced in the previous chapter.

| Dataset | Model | Feature Dim | Closed-set | | | | Open-set | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Many | Medium | Few | All | Many | Medium | Few | All | F-score |
| ImageNet-LT (ResNet-10) | Base Model [38] | 512 | 40.9 | 10.7 | 0.4 | 20.9 | 40.1 | 10.4 | 0.4 | 26.2 | 0.295 |
| | Lifted Loss [77] | 512 | 35.8 | 30.4 | 17.9 | 30.8 | 34.8 | 29.3 | 17.4 | 29.7 | 0.374 |
| | Focal Loss [64] | 512 | 36.4 | 29.9 | 16.0 | 30.5 | 35.7 | 29.3 | 15.6 | 29.8 | 0.371 |
| | Range Loss [121] | 512 | 35.8 | 30.3 | 17.6 | 30.7 | 34.7 | 29.4 | 17.2 | 29.7 | 0.373 |
| | OpenMax [5] | 512 | - | - | - | - | 35.8 | 30.3 | 17.6 | 30.6 | 0.368 |
| | FSLwF [29] | 512 | 40.9 | 22.1 | 15.0 | 28.4 | 40.8 | 21.7 | 14.5 | 28.0 | 0.347 |
| | OLTR [70] | 512 | 43.2 | 35.1 | 18.5 | 35.6 | 41.9 | 33.9 | 17.4 | 34.6 | 0.474 |
| | IEM [130] | 512 | 48.9 | 44.0 | 24.4 | 43.2 | 46.1 | 42.3 | 20.1 | 40.5 | 0.525 |
| | LUNA [9] | 512 | 51.8 | **48.6** | 26.2 | 46.6 | 48.2 | **44.7** | 23.6 | 43.0 | 0.579 |
| | **LUNA+** | 64 | 52.4 | 47.5 | 27.1 | 47.2 | 49.9 | 43.8 | 23.5 | 43.8 | 0.612 |
| | **LUNA+** | 256 | **53.1** | 47.9 | 26.9 | **47.6** | **50.0** | 44.2 | 23.1 | 43.8 | 0.613 |
| | **LUNA+** | 512 | 52.3 | 48.1 | **27.5** | 47.5 | 49.7 | 43.5 | **23.8** | **44.2** | 0.621 |

Table 5.1: OLTR performance of top-1 accuracy on the ImageNet-LT. Best results are marked in **bold**.

| Dataset | Model | Feature Dim | Closed-set | | | | Open-set | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Many | Medium | Few | All | Many | Medium | Few | All | F-score |
| Tiny-ImageNet-LT (ResNet-10) | Base Model [38] | 512 | 40.7 | 13.0 | 0.03 | 22.3 | 40.5 | 13.0 | 0.03 | 22.2 | 0.282 |
| | Lifted Loss [77] | 512 | 38.3 | 9.2 | 0.0 | 19.5 | 38.2 | 8.9 | 0.0 | 19.4 | 0.251 |
| | Focal Loss [64] | 512 | 38.3 | 18.0 | 1.9 | 23.9 | 38.2 | 18.0 | 1.7 | 23.8 | 0.299 |
| | OpenMax [5] | 512 | - | - | - | - | 39.2 | 17.6 | 3.1 | 24.2 | 0.347 |
| | OLTR [70] | 512 | 48.5 | 43.2 | 23.3 | 42.5 | 48.4 | 43.1 | 23.1 | 42.4 | 0.477 |
| | LUNA [9] | 512 | 65.3 | 51.3 | 22.3 | 52.7 | 62.5 | 48.0 | 16.7 | 49.3 | 0.629 |
| | **LUNA+** | 64 | **69.0** | 53.0 | 20.3 | 54.7 | **68.8** | 51.6 | 16.1 | 53.3 | 0.687† |
| | **LUNA+** | 256 | 68.7 | **55.4** | **24.4** | **56.2** | 68.3 | 53.0 | 17.7 | 54.0 | 0.687† |
| | **LUNA+** | 512 | 68.1 | 54.9 | 22.0 | 55.4 | 67.9 | **53.7** | **19.3** | **54.4** | **0.701** |

Table 5.2: OLTR performance of top-1 accuracy on Tiny-ImageNet-LT. Best results are marked in **bold**.

| Dataset | Model | Feature Dim | Closed-set | | | | Open-set | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Many | Medium | Few | All | Many | Medium | Few | All | F-score |
| MS-LT-v2 (ResNet-10) ‡ | Base Model [38] (a=0.2) | 512 | 50.6 | 39.2 | 21.8 | 40.3 | 50.5 | 39.0 | 21.8 | 40.2 | 0.421 |
| | Lifted Loss [77] (a=0.2) | 512 | 51.0 | 38.6 | 23.5 | 40.8 | 51.0 | 38.5 | 23.3 | 40.7 | 0.423 |
| | Focal Loss [64] (a=0.3) | 512 | 47.4 | 41.4 | 29.7 | 41.4 | 45.9 | 39.7 | 29.0 | 40.0 | 0.429 |
| | OpenMax [5] | 512 | - | - | - | - | 52.8 | 40.1 | 31.8 | 45.3 | 0.544 |
| | OLTR [70] (a=0.5) | 512 | 53.5 | 78.8 | 78.7 | 66.9 | 44.7 | 75.6 | 75.8 | 61.2 | 0.620 |
| | LUNA [9] | 64 | 66.3 | 84.3 | 76.3 | 74.0 | 64.7 | 83.2 | 70.0 | 71.4 | 0.800 |
| | LUNA [9] | 256 | 66.1 | 84.6 | 78.0 | 74.4 | 64.1 | 83.6 | 71.3 | 71.5 | 0.791 |
| | LUNA [9] | 512 | 67.9 | 85.4 | **79.0** | **75.4** | 63.9 | 83.5 | 69.8 | 71.0 | 0.776 |
| | **LUNA+** | 64 | 67.3 | 85.4 | 78.2 | 75.2 | 67.1 | 85.1 | 71.5 | 73.4 | 0.822 |
| | **LUNA+** | 256 | **68.4** | 85.4 | 77.5 | 75.5 | **68.4** | 85.4 | 77.3 | **75.5** | **0.857** |
| | **LUNA+** | 512 | 67.8 | **85.8** | 77.3 | **75.4** | 67.8 | **85.7** | **77.2** | 75.3 | 0.854 |

Table 5.3: OLTR performance of top-1 accuracy on MS-LT-v2 benchmarks. Best results are marked in **bold**. †: F-scores on 64-dim and 256-dim features are 0.6870 and 0.6873 respectively, and both are rounded to 0.687. ‡: a represents the open-set probability threshold in [70], and the default value is 0.1 for ImageNet-LT. We choose the best threshold for competing methods on MS-LT-v2 and present more details in the analysis.

Quantitative results in Table 5.1, Table 5.2 and Table 5.3 show that LUNA+ achieves SOTA performance on all the three benchmarks. Comparing to the original LUNA with WCenter loss, both closed-set (+1.0%, +2.7%, +1.1% on ImageNet-LT, Tiny-ImageNet-LT and MS-LT with feature dimension 256, respectively) and open-set (accuracy +0.8%, +2.7%, +4.0% and F-score +0.044, +0.072, +0.066), performance are significantly improved. Specifically, LUNA+ achieves better performance with smaller feature dimension compared to OLTR and IEM, suggesting the our method is more computation-efficient and memory-friendly. Moreover, we observe that LUNA+ constantly outperforms other approaches in the many-shot and medium-shot splits. We believe the reason is we do not adopt a class-balanced re-sampling strategy but use the progressively re-balancing instead, which hurts less to the accuracy on majority classes.

We also evaluate each competing method thoroughly on the newly proposed MS-LT-v2 dataset. As defined in OLTR [70], a fixed threshold $a$ is applied on the output confidence to determine open-set data. $a$ is set to 0.1 for ImageNet-LT and Tiny-ImageNet-LT. As shown in Figure 5.6, other approaches are more sensitive to this hyper-parameter. As $a$ increases from 0.1 to 0.3, the F-score improves moderately and accuracy decreases slightly. And if $a$ keeps increasing, both F-score and accuracy drop significantly. However, LUNA and LUNA+ do not choose the thresholds empirically, thus are more robust and with higher generalizability.

## 5.4 Analysis and Discussions

### 5.4.1 Effectiveness of Center Generation and Online Assignment

Center generation and the online assignment are critical steps for learning a balanced feature space in LUNA+. Qualitatively, the randomly initialized positions and optimized positions for 100 points in 2D and 3D space are visualized in Figure 5.3, where the distribution is shown to be more uniform after optimization. Though higher-dimensional hyper-sphere cannot be plotted in 2D, it validates the uniformity loss conceptually.

We project the pre-generated centers, learned features and centers using the T-SNE visualization, as shown in Figure 5.5, where the learned centers align well with the pre-generated centers.

In addition, we also use random centers, which are not updated in training without online assignment either, the results are shown in Table 5.4. Besides the improvements in classification

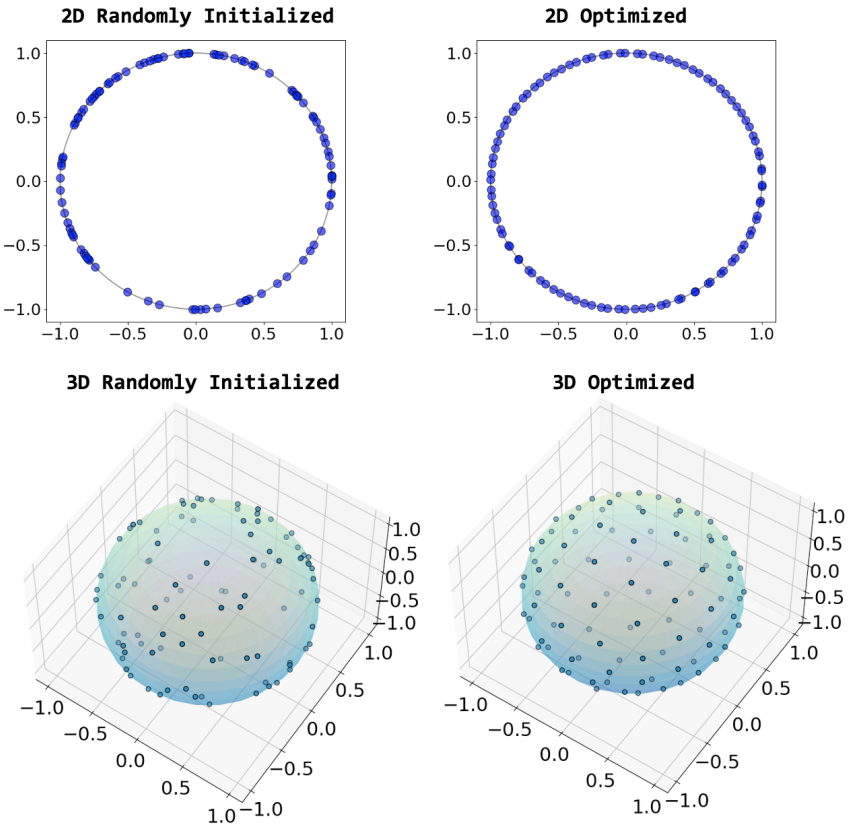



Figure 5.3: Visualization of center generation in 2D and 3D. Each point is represented in blue dots. Initialization of the centers are on the left, and the optimized results are on the right.

| Center Generation | Online Assignment | Closed-set Accuracy | Open-set Accuracy | F-score | $R_k \downarrow$ | $U_k \uparrow$ |
|---|---|---|---|---|---|---|
| ✓ | ✓ | 54.7 | 53.3 | 0.687 | 13.09 | 1.39 |
| ✓ | | 54.9 | 53.2 | 0.685 | 14.11 | 1.37 |
| | | 53.9 | 52.8 | 0.676 | 14.24 | 1.28 |

Table 5.4: Ablation study on center generation and online assignment on Tiny-ImageNet-LT with feature dimension 64. The number of neighbouring classes $k$ in $U_k$ and $R_k$ is 3.

accuracy and novel detection, we also find the center generation and online assignment are crucial in preserving the semantic closeness in feature space. As defined in [60], *neighborhood uniformity*

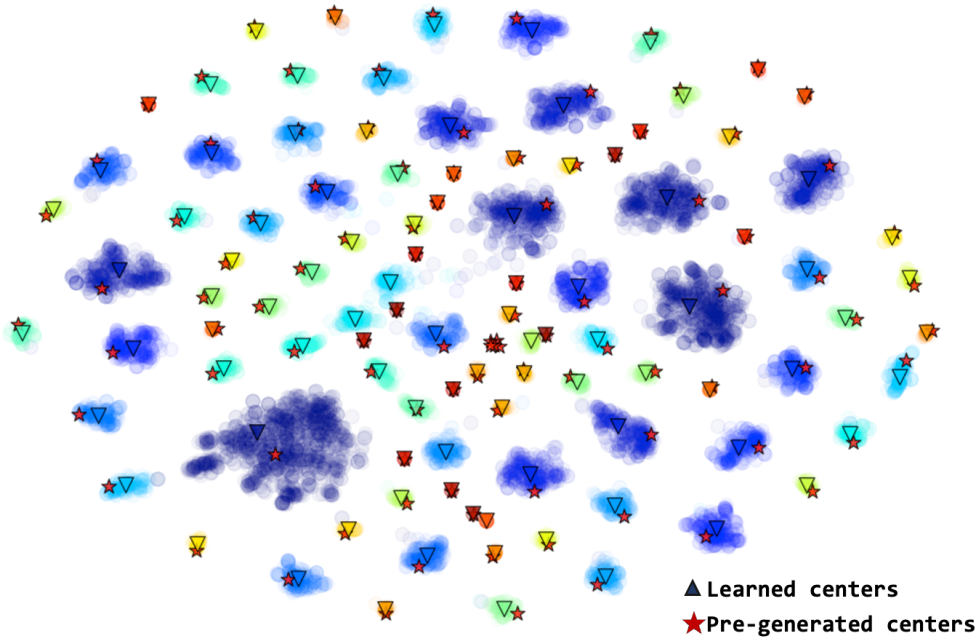



Figure 5.4: Visualization of 64 dimensional features using T-SNE method. Training data features from different classes are represented in different colors.

$U_k$ can be used to evaluate the closeness between a class to its top-$k$ closest neighbours, i.e.,

$$U_k = \frac{1}{T \cdot k} \sum_{T} \min_{r_1, r_2, \ldots, r_k} \left( \sum_{k} d(\hat{\mathbf{c}}_t, \hat{\mathbf{c}}_{r_k}) \right), \tag{5.5}$$

where $T$ is the total number of classes, $\hat{\mathbf{c}}_t$ is the learned center of class $t$ (average of all features from class $t$), and $d$ is the euclidean distance. A higher $U_k$ value indicates the class is further away from its neighbours, thus more separable.

Besides, the *reasonability* $R_k$ represents the semantic distance between a class to its top-$k$ closest neighbours. Semantic distance is the length of the shortest path between the two classes in WordNet hierarchy [74]. Smaller $R_k$ values are more desirable because the semantic meanings are better maintained.

Results in Table 5.4 show that with the optimized centers and online assignment, $R_3$ is reduced by 1.15 and $U_3$ is improved by 0.11. Specifically, center generation benefits the uniformity ($U_3$ + 0.02) and online assignment helps the reasonability ($R_3$ - 1.02).

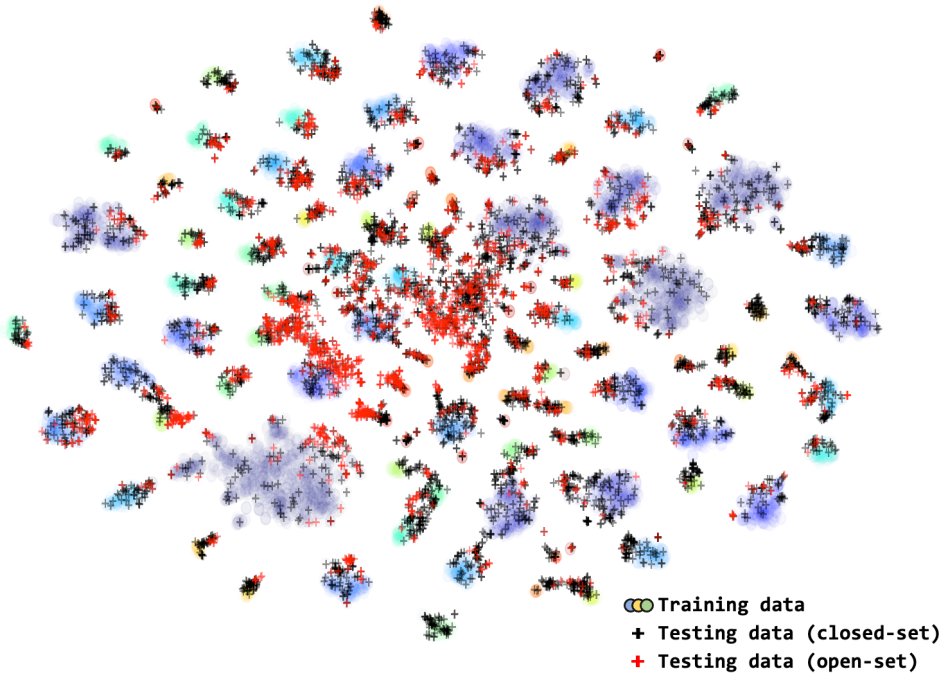



Figure 5.5: Visualization of training, closed-set testing and open-set testing features using T-SNE method. Training data features from different classes are represented in different colors.

### 5.4.2 Effectiveness of DK-SCL

In addition to the OLTR approaches, we also compare with KCL [46] and TSC [60], which use similar supervised contrastive loss as our proposed DK-SCL. Both KCL and TSC randomly select $K$ positive samples for each anchor image ($K$=6 for ImageNet-LT and Tiny-ImageNet-LT), while DK-SCL chooses the dynamic $K$ values automatically per mini-batch. The other difference is KCL and TSC are trained with class-balanced re-sampling to make the feature space more balanced while we use progressively-balanced re-sampling. We use our own implementation for both methods since their source codes are not publicly available. For fair comparison, we also apply performance-boosters to them, including the Feature Projection Module, AutoAug and Mixup.

Comparisons between TSC and KCL confirm the conclusion in Section 5.4.1 that the feature space is more uniform and separable with pre-computed centers (open-set accuracy +1.5% and F-score +0.022 with about identical closed-set accuracy). The proposed DK-SCL improves many-shot

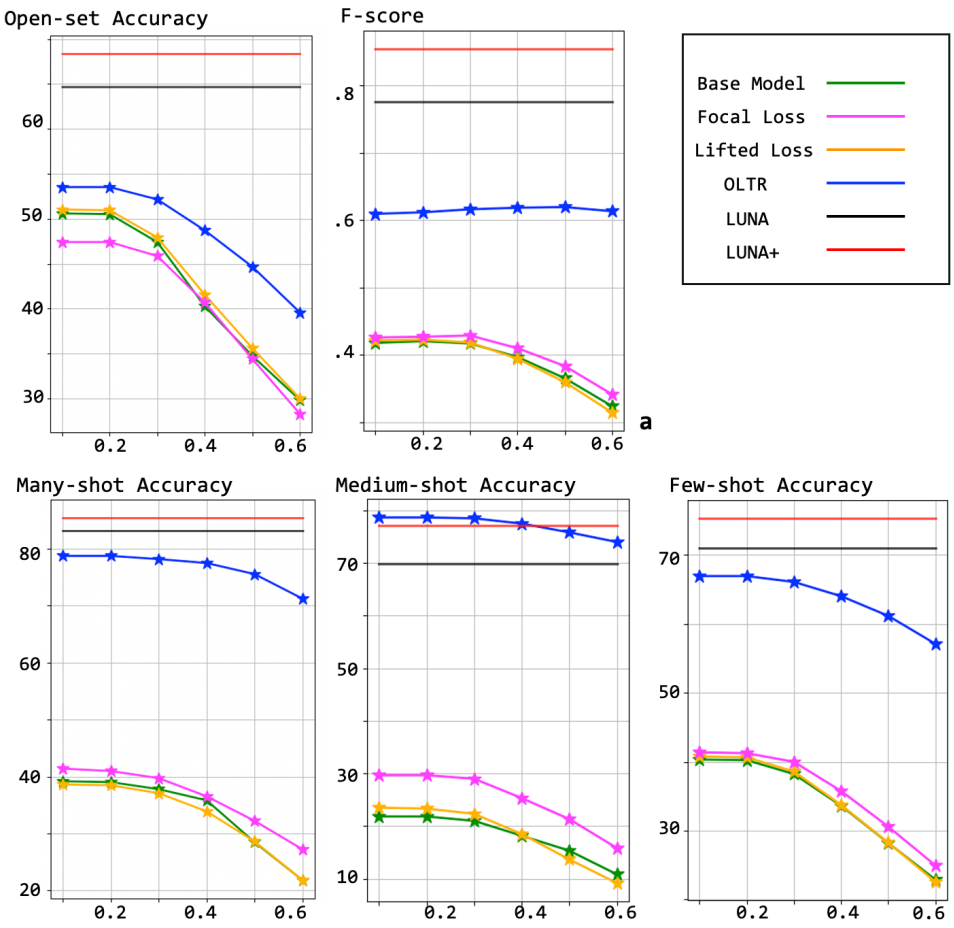



Figure 5.6: Open-set performance comparison on MS-LT-v2 with different open-set thresholds. All methods are with feature dimension 512.

| Loss | Closed-set Accuracy | | | | Open-set Accuracy | F-score |
|---|---|---|---|---|---|---|
| | Many | Medium | Few | All | | |
| KCL [47] | 65.4 | 52.9 | 19.1 | 53.0 | 49.4 | 0.643 |
| TSC [60] | 64.7 | 52.8 | **21.9** | 53.1 | 50.9 | 0.665 |
| **DK-SCL** | **69.0** | **53.0** | 20.3 | **54.7** | **53.3** | **0.687** |

Table 5.5: Ablation study on DK-SCL on Tiny-ImageNet-LT with feature dimension 64. We use LUNA+ for all three methods for open-set detection. Best results are marked in **bold**.

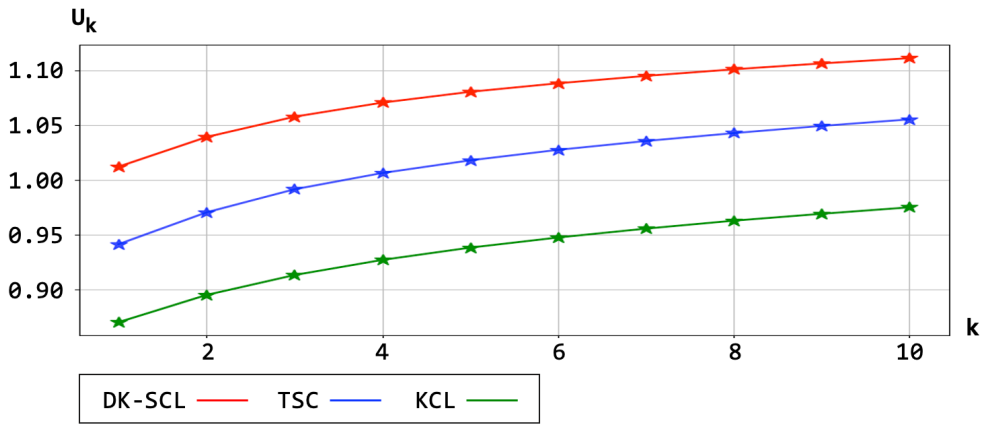



Figure 5.7: Comparison between TSC, KCL and the proposed DK-SCL on $U_k$ with the number of neighbouring classes $k \in [1, 10]$ on Tiny-ImageNet-LT. Larger $U$ means better uniformity.

and medium-shot accuracy significantly. Moreover, it drops less (-1.4% from closed-set to open-set) in accuracy, meaning it is more robust to outliers in the closed testing set. Because the models are trained with long-tailed dataset and tested with a balanced set, testing samples from tail classes are prone to be identified as outliers because the model's bias. To validate this, we compare the uniformity of the three approaches and visualize in Figure 5.7. The results show that with center generation (DK-SCL and TSC), the uniformity is better than randomly initialized centers (KCL). And the dynamic $K$ in DK-SCL further contributes to the superiority.

### 5.4.3 Effectiveness of LUNA+

To evaluate the quality of learned features, we compare the proposed LUNA+ factor with a widely used density-based novelty detection method, Local Outluer Factor (LOF) [8]. The parameters in LOF are determined as in the original paper. Two conclusions can be drawn from results in Table 5.6: (1) LUNA+ factor consistently outperforms LOF with various metric learning methods (+0.151 with KCL, +0.220 with TSC and +0.135 with DK-SCL). (2) DK-SCL features are more density-sensitive when compared to features trained by other losses. If feature dimension is 64, DK-SCL is +0.022 and +0.067 better than TSC with LUNA+ and LOF, respectively.

| Loss | Feature Dim | LUNA+ | | | | | LOF [8] | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Many | Med | Few | All | F-score | Many | Med | Few | All | F-score |
| KCL [47] | 64 | 65.0 | 47.0 | 14.1 | 49.4 | 0.643 | 32.9 | 41.4 | 11.1 | 33.8 | 0.492 |
| TSC [60] | 64 | 64.4 | 50.3 | 15.0 | 50.9 | 0.665 | 28.5 | 38.8 | 11.4 | 30.9 | 0.445 |
| **DK-SCL** | 64 | **69.0** | 51.6 | 16.1 | 53.3 | 0.687 | 38.8 | 38.9 | 12.0 | **35.1** | 0.512 |
| **DK-SCL** | 256 | 68.3 | 53.0 | 17.7 | 54.0 | 0.687 | **38.8** | **42.2** | **14.4** | **38.1** | **0.544** |
| **DK-SCL** | 512 | 67.9 | **53.7** | **19.3** | **54.4** | **0.701** | 41.2 | 41.2 | 12.9 | 37.4 | 0.534 |

Table 5.6: Comparisons between LUNA+ and LOF on Tiny-ImageNet-LT with multiple loss functions under the open-set setting. Best results are marked in **bold**.

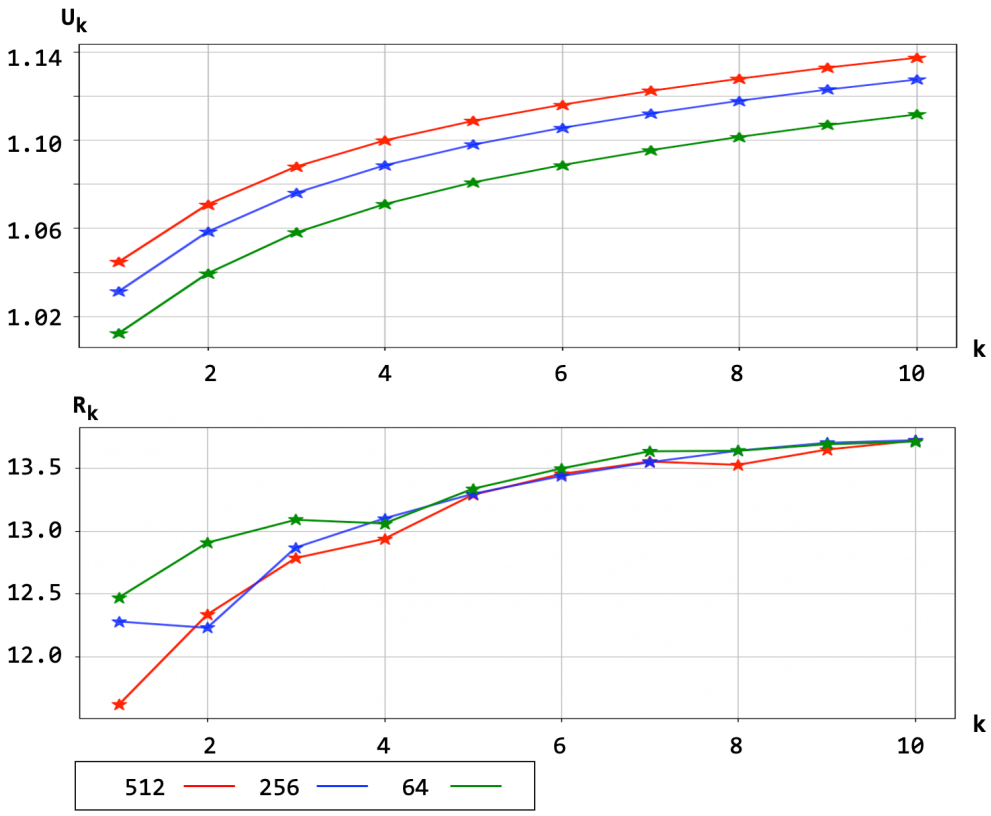



Figure 5.8: Comparison between DK-SCL with different feature dimensions on $R_k$ with the number of neighbouring classes $k \in [1, 10]$ on Tiny-ImageNet-LT. Larger $U$ means better uniformity, smaller $R$ means better reasonability.

### 5.4.4 Dimension of Features

As shown in Table 5.6, we also observe better novelty detection performance with higher feature dimension. One of the reasons is due to the fact that higher dimensional features preserve semantic closeness better. Figure 5.8 indicates that $R$ is better with larger dimension given the same $K$, especially when $K$ is small. The uniformity $U$ is very close for different dimensions, and 512-dim achieves slightly improved performance, where the pre-computed centers that are more uniformly distributed is used.

# Chapter 6

# CONCLUSION AND FUTURE WORKS

## 6.1 Conclusion

In this thesis, three research works on long-tailed recognition under both closed-set and open-set settings are introduced. In ACE, extensive experiments on existing long-tailed recognition algorithms reveal the contradiction between biased representation learning and unbiased classifier learning. We proposed a multi-expert network that optimizes the two in a uniform network. Complementary constraints in data and objective function are applied to suppress the effects of non-targeted groups and promote both of the dominating and minority groups. Besides, a distribution-adaptive optimization scheme helps to adjust the learning paces of each expert to avoid over-fitting. ACE becomes the new SOTA among all one-stage long-tailed recognition methods with 3$\sim$10% accuracy gain, and is the first one that improves performance on all three frequency splits. With the equivalent strong performance to the multi-stage methods, there is great potential to extend well-formulated one-stage ACE to complex computer vision tasks like detection and segmentation.

For open-set long-tailed recognition, we achieved "killing three birds with one stone" with LUNA framework. By introducing a fine-grained natural OLTR dataset about ocean fish species, researchers can engage to the real OLTR challenges in lab. Such a dataset can be a solid supplement to the existing, manually re-sampled OLTR benchmarks. Secondly, a new wcenter loss is designed to minimize the intra-class distance in the feature space, which preserves classification accuracy while optimizing the clustering for both the heads and tails. In addition, we propose the LUNA, which is an effective measure of novelty based on the relative local density of the learned representation. LUNA+ further addresses the feature space imbalance issue by introducing uniformity of the centers. Moreover, the semantic closeness is preserved by online assignment between the targeted centers and learned features. Besides, a distribution-sensitive contrastive loss is proposed, which is more stable and performs better than the wcenter loss. Our proposed LUNA and LUNA+ significantly outperforms the SOTA OLTR algorithms in all public and customized benchmarks.

## 6.2 Future Works

### 6.2.1 Imbalance and Open-set Problem in Other Vision Tasks

The object detection and segmentation are commonly studied in a few categories regime, where annotation per class is rich and relatively balanced over the whole dataset, for example, the Microsoft COCO dataset [66]. The large-scale few-shot detection and segmentation is a new opportunity to push the existing algorithms towards more complex real-world scenes that a huge number of rare and popular objects co-exist. Some pilot studies [33, 63] imply the off-the-shelf detectors, including Faster-RCNN [85], Mask-RCNN [37] and Hybrid task cascade (HTC) [13], are not sufficient for long-tailed object detection. Similar to long-tailed recognition, existing works address the imbalance issue mainly from three perspectives: (1) re-balancing techniques, including data re-sampling [33, 102] and loss re-weighting [94, 40, 101, 103]; (2) multi-stage re-training [95]; and (3) classifier ensemble [63, 102].

All these methods are purely applied on the classification head, the bounding box regression and mask heads are unchanged. Finding the principle obstacles preventing the state-of-the-art detectors from decent performance on imbalance dataset and studying the effect of re-balancing techniques to the regression and mask heads are directions to find the optimal solution to the long-tailed detection/segmentation task. Due to the limitations in computational resources, we did not integrate ACE framework into detection task at this point. Future Research could be conducted to study the performance of multi-expert architecture in other computer vision tasks.

### 6.2.2 Novelty Detection, Actively Learning and Beyond

In this thesis, we present the novelty detection approaches LUNA and LUNA+, which are capable to identify out-of-distribution samples. The next step of research will be how to update the deep feature extractor and classifier to learn form the novel classes. Prior research [61, 100] show that the strategies of choosing samples for model update are critical. Ideally, the samples are representative enough and should be as few as possible to reduce human annotation. Therefore, the testing sample's diversity and uncertainty are significant indicators. Our proposed LUNA factor is a measurement of uncertainty, moreover, the intermediate feature densities and outlier factors are also correlated with diversity. In future study, adding active learning will be a huge benefit to

real-world applications: out-of-distribution and novel samples can further improve the model's representation learning capability in the long run.

# BIBLIOGRAPHY

[1] Summary of notifiable diseases — centers for disease control and prevention, united states, https://www.cdc.gov/mmwr/preview/mmwrhtml.

[2] The mnist database, http://yann.lecun.com/exdb/mnist/.

[3] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. *Advances in neural information processing systems*, 32, 2019.

[4] Hessam Bagherinezhad, Maxwell Horton, Mohammad Rastegari, and Ali Farhadi. Label refinery: Improving imagenet classification through label progression. *arXiv preprint arXiv:1805.02641*, 2018.

[5] Abhijit Bendale and Terrance E Boult. Towards open set deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1563–1572, 2016.

[6] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020.

[7] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[8] Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. Lof: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pages 93–104, 2000.

[9] Jiarui Cai, Yizhou Wang, Hung-Min Hsu, Jenq-Neng Hwang, Kelsey Magrane, and Carig Rose. Luna: Localizing unfamiliarity near acquaintance for open-set long-tailed recognition. In *AAAI*, 2022.

[10] Jiarui Cai, Yizhou Wang, and Jenq-Neng Hwang. Ace: Ally complementary experts for solving long-tailed recognition in one-shot. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 112–121, 2021.

[11] 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.

[12] 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.

[13] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 4974–4983, 2019.

[14] Hsin-Ping Chou, Shih-Chieh Chang, Jia-Yu Pan, Wei Wei, and Da-Cheng Juan. Remix: Rebalanced mixup. In *European Conference on Computer Vision*, pages 95–110. Springer, 2020.

[15] Henry Cohn and Abhinav Kumar. Universally optimal distribution of points on spheres. *Journal of the American Mathematical Society*, 20(1):99–148, 2007.

[16] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation policies from data. *arXiv preprint arXiv:1805.09501*, 2018.

[17] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 9268–9277, 2019.

[18] 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/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[19] Jia Deng, Nan Ding, Yangqing Jia, Andrea Frome, Kevin Murphy, Samy Bengio, Yuan Li, Hartmut Neven, and Hartwig Adam. Large-scale object classification using label relation graphs. In *European conference on computer vision*, pages 48–64. Springer, 2014.

[20] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[21] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2019.

[22] Anita Elberse and Felix Oberholzer-Gee. *Superstars and underdogs: An examination of the long tail phenomenon in video sales*. Citeseer, 2006.

[23] Eleazar Eskin. Anomaly detection over noisy data using learned probability distributions. 2000.

[24] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer Vision and Pattern Recognition Workshop*, 2004.

[25] Geoff French, Michal Mackiewicz, Mark Fisher, Helen Holah, Rachel Kilburn, Neil Campbell, and Coby Needle. Deep neural networks for analysis of fisheries surveillance video and automated monitoring of fish discards. *ICES Journal of Marine Science*, 77(4):1340–1353, 2020.

[26] Peter W Frey and David J Slate. Letter recognition using holland-style adaptive classifiers. *Machine learning*, 6(2):161–182, 1991.

[27] Mikel Galar, Alberto Fernández, Edurne Barrenechea, and Francisco Herrera. Eusboost: Enhancing ensembles for highly imbalanced data-sets by evolutionary undersampling. *Pattern recognition*, 46(12):3460–3471, 2013.

[28] ZongYuan Ge, Sergey Demyanov, Zetao Chen, and Rahil Garnavi. Generative openmax for multi-class open set classification. *arXiv preprint arXiv:1707.07418*, 2017.

[29] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4367–4375, 2018.

[30] Wonjoon Goo, Juyong Kim, Gunhee Kim, and Sung Ju Hwang. Taxonomy-regularized semantic deep convolutional neural networks. In *European Conference on Computer Vision*, pages 86–101. Springer, 2016.

[31] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

[32] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *European conference on computer vision*, pages 87–102. Springer, 2016.

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

[34] 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.

[35] Mahmudul Hasan, Jonghyun Choi, Jan Neumann, Amit K Roy-Chowdhury, and Larry S Davis. Learning temporal regularity in video sequences. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 733–742, 2016.

[36] Mehadi Hassen and Philip K Chan. Learning a neural-network-based representation for open set recognition. In *Proceedings of the 2020 SIAM International Conference on Data Mining*, pages 154–162. SIAM, 2020.

[37] 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.

[38] 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.

[39] Matthias Hein, Maksym Andriushchenko, and Julian Bitterwolf. Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 41–50, 2019.

[40] Ting-I Hsieh, Esther Robb, Hwann-Tzong Chen, and Jia-Bin Huang. Droploss for long-tail instance segmentation. *arXiv preprint arXiv:2104.06402*, 2021.

[41] 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 (CVPR)*, June 2016.

[42] Tsung-Wei Huang, Jenq-Neng Hwang, and Craig S Rose. Chute based automated fish length measurement and water drop detection. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1906–1910. IEEE, 2016.

[43] Muhammad Abdullah Jamal, Matthew Brown, Ming-Hsuan Yang, Liqiang Wang, and Boqing Gong. Rethinking class-balanced methods for long-tailed visual recognition from a domain adaptation perspective. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7610–7619, 2020.

[44] Nathalie Japkowicz and Shaju Stephen. The class imbalance problem: A systematic study. *Intelligent data analysis*, 6(5):429–449, 2002.

[45] Pedro R Mendes Júnior, Roberto M De Souza, Rafael de O Werneck, Bernardo V Stein, Daniel V Pazinato, Waldir R de Almeida, Otávio AB Penatti, Ricardo da S Torres, and Anderson Rocha. Nearest neighbors distance ratio open-set classifier. *Machine Learning*, 106(3):359–386, 2017.

[46] Bingyi Kang, Yu Li, Sa Xie, Zehuan Yuan, and Jiashi Feng. Exploring balanced feature spaces for representation learning. In *International Conference on Learning Representations*, 2020.

[47] 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.

[48] Ira Kemelmacher-Shlizerman, Steven M Seitz, Daniel Miller, and Evan Brossard. The megaface benchmark: 1 million faces for recognition at scale. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4873–4882, 2016.

[49] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33:18661–18673, 2020.

[50] Hyo Jin Kim and Jan-Michael Frahm. Hierarchy of alternating specialists for scene recognition. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 451–467, 2018.

[51] Jaehyung Kim, Jongheon Jeong, and Jinwoo Shin. M2m: Imbalanced classification via major-to-minor translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13896–13905, 2020.

[52] Hans-Peter Kriegel, Peer Kröger, Erich Schubert, and Arthur Zimek. Loop: local outlier probabilities. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 1649–1652, 2009.

[53] Alex Krizhevsky. One weird trick for parallelizing convolutional neural networks. *arXiv preprint arXiv:1404.5997*, 2014.

[54] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[55] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[56] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[57] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[58] Brian Kulis et al. Metric learning: A survey. *Foundations and trends in machine learning*, 5(4):287–364, 2012.

[59] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010.

[60] Tianhong Li, Peng Cao, Yuan Yuan, Lijie Fan, Yuzhe Yang, Rogerio Feris, Piotr Indyk, and Dina Katabi. Targeted supervised contrastive learning for long-tailed recognition. *arXiv preprint arXiv:2111.13998*, 2021.

[61] Xin Li and Yuhong Guo. Adaptive active learning for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 859–866, 2013.

[62] 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 (CVPR)*, June 2020.

[63] 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.

[64] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[65] 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.

[66] 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.

[67] Jialun Liu, Yifan Sun, Chuchu Han, Zhaopeng Dou, and Wenhui Li. Deep representation learning on long-tailed data: A learnable embedding augmentation perspective. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2970–2979, 2020.

[68] Xu-Ying Liu, Jianxin Wu, and Zhi-Hua Zhou. Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(2):539–550, 2008.

[69] Yuntao Liu, Yong Dou, Ruochun Jin, and Peng Qiao. Visual tree convolutional neural network in image classification. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 758–763. IEEE, 2018.

[70] 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.

[71] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[72] Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. *arXiv preprint arXiv:2007.07314*, 2020.

[73] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. *arXiv preprint arXiv:1310.4546*, 2013.

[74] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[75] Athma Narayanan, Isht Dwivedi, and Behzad Dariush. Dynamic traffic scene classification with space-time coherence. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 5629–5635. IEEE, 2019.

[76] Hien M Nguyen, Eric W Cooper, and Katsuari Kamei. Borderline over-sampling for imbalanced data classification. *International Journal of Knowledge Engineering and Soft Data Paradigms*, 3(1):4–21, 2011.

[77] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4004–4012, 2016.

[78] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1):62–66, 1979.

[79] Yoon-Joo Park and Alexander Tuzhilin. The long tail of recommender systems and how to leverage it. In *Proceedings of the 2008 ACM conference on Recommender systems*, pages 11–18, 2008.

[80] Pramuditha Perera, Ramesh Nallapati, and Bing Xiang. Ocgan: One-class novelty detection using gans with constrained latent representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2898–2906, 2019.

[81] Stanislav Pidhorskyi, Ranya Almohsen, and Gianfranco Doretto. Generative probabilistic novelty detection with adversarial autoencoders. *Advances in neural information processing systems*, 31, 2018.

[82] Ariadna Quattoni and Antonio Torralba. Recognizing indoor scenes. In *2009 IEEE conference on computer vision and pattern recognition*, pages 413–420. IEEE, 2009.

[83] Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. Training deep neural networks on noisy labels with bootstrapping. *arXiv preprint arXiv:1412.6596*, 2014.

[84] William J Reed. The pareto, zipf and other power laws. *Economics letters*, 74(1):15–19, 2001.

[85] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv preprint arXiv:1506.01497*, 2015.

[86] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[87] Mohammad Sabokrou, Mohammad Khalooei, Mahmood Fathy, and Ehsan Adeli. Adversarially learned one-class classifier for novelty detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3379–3388, 2018.

[88] Mayu Sakurada and Takehisa Yairi. Anomaly detection using autoencoders with nonlinear dimensionality reduction. In *Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis*, MLSDA'14, page 4–11, 2014.

[89] Walter J Scheirer, Lalit P Jain, and Terrance E Boult. Probability models for open set recognition. *IEEE transactions on pattern analysis and machine intelligence*, 36(11):2317–2324, 2014.

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

[91] Ning Su, Natalia Levina, and Jeanne W Ross. The long-tail strategy of it outsourcing. *MIT Sloan Management Review*, 57(2):81, 2016.

[92] 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.

[93] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[94] Jingru Tan, Xin Lu, Gang Zhang, Changqing Yin, and Quanquan Li. Equalization loss v2: A new gradient balance approach for long-tailed object detection. *arXiv preprint arXiv:2012.08548*, 2020.

[95] Jingru Tan, Gang Zhang, Hanming Deng, Changbao Wang, Lewei Lu, Quanquan Li, and Jifeng Dai. 1st place solution of lvis challenge 2020: A good box is not a guarantee of a good mask. *arXiv preprint arXiv:2009.01559*, 2020.

[96] Kaihua Tang, Jianqiang Huang, and Hanwang Zhang. Long-tailed classification by keeping the good and removing the bad momentum causal effect. In *NeurIPS*, 2020.

[97] Bing Tu, Chengle Zhou, Wenlan Kuang, Longyuan Guo, and Xianfeng Ou. Hyperspectral imagery noisy label detection by spectral angle local outlier factor. *IEEE Geoscience and Remote Sensing Letters*, 15(9):1417–1421, 2018.

[98] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8769–8778, 2018.

[99] Chen Wang, Chengyuan Deng, and Suzhen Wang. Imbalance-xgboost: leveraging weighted and focal losses for binary label-imbalanced classification with xgboost. *Pattern Recognition Letters*, 136:190–197, 2020.

[100] Gaoang Wang, Jenq-Neng Hwang, Craig Rose, and Farron Wallace. Uncertainty-based active learning via sparse modeling for image classification. *IEEE Transactions on Image Processing*, 28(1):316–329, 2018.

[101] 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. *arXiv preprint arXiv:2008.10032*, 2020.

[102] 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 *European conference on computer vision*, pages 728–744. Springer, 2020.

[103] Tong Wang, Yousong Zhu, Chaoyang Zhao, Wei Zeng, Jinqiao Wang, and Ming Tang. Adaptive class suppression loss for long-tail object detection. *arXiv preprint arXiv:2104.00885*, 2021.

[104] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pages 9929–9939. PMLR, 2020.

[105] Wei Wang and Peizhong Lu. An efficient switching median filter based on local outlier factor. *IEEE Signal Processing Letters*, 18(10):551–554, 2011.

[106] Xinggang Wang, Xianbo Deng, Qing Fu, Qiang Zhou, Jiapei Feng, Hui Ma, Wenyu Liu, and Chuansheng Zheng. A weakly-supervised framework for covid-19 classification and lesion localization from chest ct. *IEEE transactions on medical imaging*, 39(8):2615–2625, 2020.

[107] Xudong Wang, Long Lian, Zhongqi Miao, Ziwei Liu, and Stella X Yu. Long-tailed recognition by routing diverse distribution-aware experts. *arXiv preprint arXiv:2010.01809*, 2020.

[108] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Learning to model the tail. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 7032–7042, 2017.

[109] Friedrich-Wilhelm Wellmer and Jens Dieter Becker-Platen. Global nonfuel mineral resources and sustainability. In *Proceedings for a Workshop on Deposit Modeling, Mineral Resource Assessment, and Their Role in Sustainable Development*, page 1, 2000.

[110] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *European conference on computer vision*, pages 499–515. Springer, 2016.

[111] Jialian Wu, Liangchen Song, Tiancai Wang, Qian Zhang, and Junsong Yuan. Forest r-cnn: Large-vocabulary long-tailed object detection and instance segmentation. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1570–1578, 2020.

[112] Tz-Ying Wu, Pedro Morgado, Pei Wang, Chih-Hui Ho, and Nuno Vasconcelos. Solving long-tailed recognition with deep realistic taxonomic classifier. In *European Conference on Computer Vision (ECCV)*, 2020.

[113] Liuyu Xiang, Guiguang Ding, and Jungong Han. Learning from multiple experts: Self-paced knowledge distillation for long-tailed classification. In *European Conference on Computer Vision*, pages 247–263. Springer, 2020.

[114] 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.

[115] Dan Xu, Elisa Ricci, Yan Yan, Jingkuan Song, and Nicu Sebe. Learning deep representations of appearance and motion for anomalous event detection. *arXiv preprint arXiv:1510.01553*, 2015.

[116] Zhenbo Xu, Wei Zhang, Xiao Tan, Wei Yang, Huan Huang, Shilei Wen, Errui Ding, and Liusheng Huang. Segment as points for efficient online multi-object tracking and segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[117] Kenji Yamanishi, Jun-Ichi Takeuchi, Graham Williams, and Peter Milne. On-line unsupervised outlier detection using finite mixtures with discounting learning algorithms. *Data Mining and Knowledge Discovery*, 8(3):275–300, 2004.

[118] Weiping Yu, Taojiannan Yang, and Chen Chen. Towards resolving the challenge of long-tail distribution in uav images for object detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3258–3267, 2021.

[119] Yuhang Zang, Chen Huang, and Chen Change Loy. Fasa: Feature augmentation and sampling adaptation for long-tailed instance segmentation. *arXiv preprint arXiv:2102.12867*, 2021.

[120] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

[121] 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.

[122] Yifan Zhang, Bryan Hooi, Lanqing Hong, and Jiashi Feng. Test-agnostic long-tailed recognition by test-time aggregating diverse experts with self-supervision. *arXiv preprint arXiv:2107.09249*, 2021.

[123] Yongshun Zhang, Xiu-Shen Wei, Boyan Zhou, and Jianxin Wu. Bag of tricks for long-tailed visual recognition with deep convolutional neural networks. 2021.

[124] Chensu Zhao, Yang Xin, Xuefeng Li, Yixian Yang, and Yuling Chen. A heterogeneous ensemble learning framework for spam detection in social networks with imbalanced data. *Applied Sciences*, 10(3):936, 2020.

[125] Bin Zhou, Li Fei-Fei, and Eric P Xing. Large-scale category structure aware image categorization. 2011.

[126] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464, 2017.

[127] Boyan Zhou, Quan Cui, Xiu-Shen Wei, and Zhao-Min Chen. Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9719–9728, 2020.

[128] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. In *arXiv preprint arXiv:1904.07850*, 2019.

[129] Jinlin Zhu, Youqing Wang, Donghua Zhou, and Furong Gao. Batch process modeling and monitoring with local outlier factor. *IEEE Transactions on Control Systems Technology*, 27(4):1552–1565, 2018.

[130] Linchao Zhu and Yi Yang. Inflated episodic memory with region self-attention for long-tailed visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4344–4353, 2020.

[131] Xinqi Zhu and Michael Bain. B-cnn: branch convolutional neural network for hierarchical classification. *arXiv preprint arXiv:1709.09890*, 2017.