Large-Scale Long-Tailed Recognition in an Open World

Ziwei Liu[1,2*] Zhongqi Miao[2*] Xiaohang Zhan[1] Jiayun Wang[2] Boqing Gong[2†] Stella X. Yu[2]
[1] The Chinese University of Hong Kong [2] UC Berkeley / ICSI
{zwliu,zx017}@ie.cuhk.edu.hk, {zhongqi.miao,peterwg,stellayu}@berkeley.edu, bgong@outlook.com

Abstract

Real world data often have a long-tailed and open-ended distribution. A practical recognition system must classify among majority and minority classes, generalize from a few known instances, and acknowledge novelty upon a never seen instance. We define Open Long-Tailed Recognition (OLTR) as learning from such naturally distributed data and optimizing the classification accuracy over a balanced test set which include head, tail, and open classes.

OLTR must handle imbalanced classification, few-shot learning, and open-set recognition in one integrated algorithm, whereas existing classification approaches focus only on one aspect and deliver poorly over the entire class spectrum. The key challenges are how to share visual knowledge between head and tail classes and how to reduce confusion between tail and open classes.

We develop an integrated OLTR algorithm that maps an image to a feature space such that visual concepts can easily relate to each other based on a learned metric that respects the closed-world classification while acknowledging the novelty of the open world. Our so-called dynamic meta-embedding combines a direct image feature and an associated memory feature, with the feature norm indicating the familiarity to known classes. On three large-scale OLTR datasets we curate from object-centric ImageNet, scene-centric Places, and face-centric MS1M data, our method consistently outperforms the state-of-the-art. Our code, datasets, and models enable future OLTR research and are publicly available at https://liuziwei7.github.io/projects/LongTail.html.

1. Introduction

Our visual world is inherently long-tailed and open-ended: The frequency distribution of visual categories in our daily life is long-tailed [41], with a few common classes and many more rare classes, and we constantly encounter new visual concepts as we navigate in an open world.

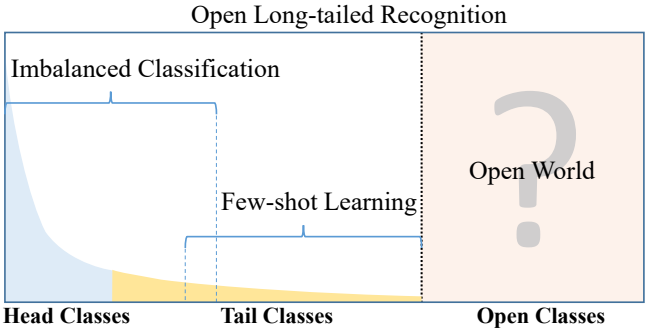


Figure 1: Our task of open long-tailed recognition must learn from long-tail distributed training data in an open world and deal with imbalanced classification, few-shot learning, and open-set recognition over the entire spectrum.

While the natural data distribution contains head, tail, and open classes (Fig. 1), existing classification approaches focus mostly on the head [8, 30], the tail [54, 27], often in a closed setting [58]. Traditional deep learning models are good at capturing the big data of head classes [26, 20]; more recently, few-shot learning methods have been developed for the small data of tail classes [51, 18].

We formally study *Open Long-Tailed Recognition* (OLTR) arising in natural data settings. A practical system shall be able to classify among a few common and many rare categories, to generalize the concept of a single category from only a few known instances, and to acknowledge novelty upon an instance of a never seen category. We define OLTR as learning from long-tail and open-end distributed data and evaluating the classification accuracy over a balanced test set which include head, tail, and open classes in a continuous spectrum (Fig. 1).

OLTR must handle not only imbalanced classification and few-shot learning in the closed world, but also open-set recognition with one integrated algorithm (Tab. 1). Existing classification approaches tend to focus on one aspect and deliver poorly over the entire class spectrum.

The key challenges for OLTR are tail recognition robustness and open-set sensitivity: As the number of training instances drops from thousands in the head class to the few

*Equal contribution.
†Work done in part at Tencent AI Lab.

Task Setting	Imbalanced Train/Base Set	#Instances in Tail Class	Balanced Test Set	Open Class	Evaluation: Accuracy Over ?
Imbalanced Classification	✓	20∼50	×	×	all classes
Few-Shot Learning	×	1∼20	✓	×	novel classes
Open-Set Recognition	×	N/A	✓	✓	all classes
Open Long-Tailed Recognition	✓	1∼20	✓	✓	all classes

Table 1: Comparison between our proposed OLTR task and related existing tasks.

in the tail class, the recognition accuracy should maintain as high as possible; on the other hand, as the number of instances drops to zero in the open set, the recognition accuracy relies on the sensitivity to distinguish unknown open classes from known tail classes.

An integrated OLTR algorithm should tackle the two seemingly contradictory aspects of recognition robustness and recognition sensitivity on a continuous category spectrum. To increase the recognition robustness, it must share visual knowledge between head and tail classes; to increase recognition sensitivity, it must reduce the confusion between tail and open classes.

We develop an OLTR algorithm that maps an image to a feature space such that visual concepts can easily relate to each other based on a learned metric that respects the closed-world classification while acknowledging the novelty of the open world.

Our so-called *dynamic meta-embedding* handles tail recognition robustness by combining two components: a direct feature computed from the input image, and an induced feature associated with the visual memory. **1)** Our direct feature is a standard embedding that gets updated from the training data by stochastic gradient descent over the classification loss. The direct feature lacks sufficient supervision for the rare tail class. **2)** Our memory feature is inspired by meta learning methods with memories [54, 12, 2] to augment the direct feature from the image. A visual memory holds discriminative centroids of the direct feature. We learn to retrieve a summary of memory activations from the direct feature, combined into a meta-embedding that is enriched particularly for the tail class.

Our dynamic meta-embedding handles open recognition sensitivity by dynamically calibrating the meta-embedding with respect to the visual memory. The embedding is scaled inversely by its distance to the nearest centroid: The farther away from the memory, the closer to the origin, and the more likely an open set instance. We also adopt *modulated attention* [55] to encourage the head and tail classes to use different sets of spatial features. As our meta-embedding relates head and tail classes, our modulated attention maintains discrimination between them.

We make the following major contributions. **1)** We formally define the OLTR task, which learns from natural long-tail and open-end distributed data and optimizes the overall accuracy over a balanced test set. It provides a comprehensive and unbiased evaluation of visual recognition algorithms in practical settings. **2)** We develop an integrated OLTR algorithm with dynamic meta-embedding. It handles tail recognition robustness by relating visual concepts among head and tail embeddings, and it handles open recognition sensitivity by dynamically calibrating the embedding norm with respect to the visual memory. **3)** We curate three large OLTR datasets according to a long-tail distribution from existing representative datasets: object-centric ImageNet, scene-centric MIT Places, and face-centric MS1M datasets. We set up benchmarks for proper OLTR performance evaluation. **4)** Our extensive experimentation on these OLTR datasets demonstrates that our method consistently outperforms the state-of-the-art.

Our code, datasets, and models are publicly available at https://liuziwei7.github.io/projects/LongTail.html. Our work fills the void in practical benchmarks for imbalanced classification, few-shot learning, and open-set recognition, enabling future research that is directly transferable to real-world applications.

2. Related Works

While OLTR has not been defined in the literature, there are three closely related tasks which are often studied in isolation: imbalanced classification, few-shot learning, and open-set recognition. Tab. 1 summarizes their differences.

Imbalanced Classification. Arising from long-tail distributions of natural data, it has been extensively studied [44, 65, 4, 32, 66, 37, 31, 52, 7]. Classical methods include under-sampling head classes, over-sampling tail classes, and data instance re-weighting. We refer the readers to [19] for a detailed review. Some recent methods include *metric learning* [24, 36], *hard negative mining* [11, 29], and *meta learning* [17, 58]. The lifted structure loss [36] introduces margins between many training instances. The range loss [63] enforces data in the same class to be close and those in different classes to be far apart. The focal loss [29] induces an online version of hard negative mining. MetaModelNet [58] learns a meta regression net from head classes and uses it to construct the classifier for tail classes.

Our dynamic meta-embedding combines the strengths of both metric learning and meta learning. On one hand, our direct feature is updated to ensure centroids for different classes are far from each other; On the other hand, our memory feature is generated on-the-fly in a meta learning fashion to effectively transfer knowledge to tail classes.

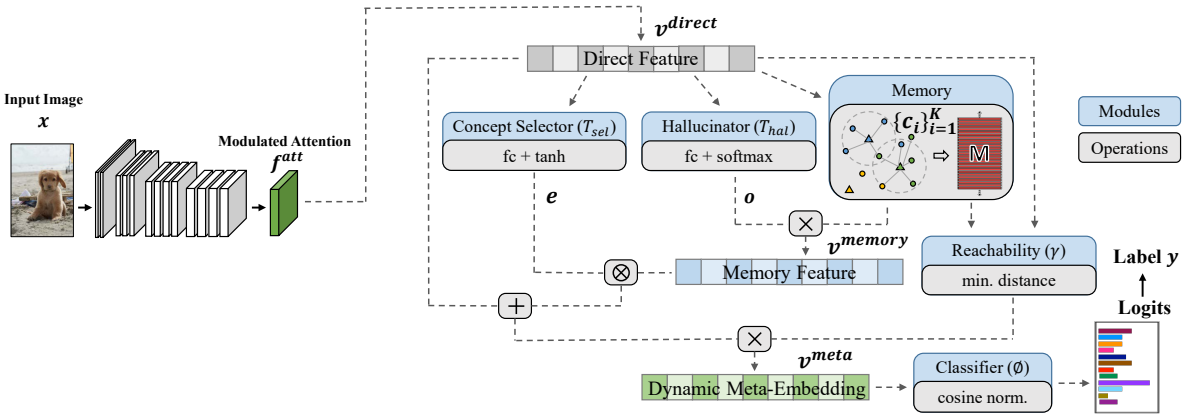


Figure 2: **Method overview.** There are two main modules: dynamic meta-embedding and modulated attention. The embedding relates visual concepts between head and tail classes, while the attention discriminates between them. The *r*eachability separates tail and open classes.

Few-Shot Learning. It is often formulated as meta learning [49, 6, 40, 45, 14, 60]. Matching Network [54] learns a transferable feature matching metric to go beyond given classes. Prototypical Network [51] maintains a set of separable class templates. Feature hallucination [18] and augmentation [56] are also shown effective. Since these methods focus on novel classes, they often suffer a moderate performance drop for head classes. There are a few exceptions. The few-shot learning without forgetting [15] and incremental few-shot learning [42] attempt to remedy this issue by leveraging the duality between features and classifiers' weights [39, 38]. However, the training set used in all of these methods are balanced.

In comparison, our OLTR learns from a more natural long-tailed training set. Nevertheless, our work is closely related to meta learning with fast weight and associative memory [22, 48, 54, 12, 2, 35] to enable rapid adaptation. Compared to these prior arts, our memory feature has two advantages: **1)** It transfers knowledge to both head and tail classes adaptively via a learned concept selector; **2)** It is fully integrated into the network without episodic training, and is thus especially suitable for large-scale applications.

Open-Set Recognition. Open-set recognition [47, 3], or out-of-distribution detection [10, 28], aims to re-calibrate the sample confidence in the presence of open classes. One of the representative techniques is OpenMax [3], which fits a Weibull distribution to the classifier's output logits. However, when there are both open and tail classes, the distribution fitting could confuse the two.

Instead of calibrating the output logits, our OLTR approach incorporates the confidence estimation into feature learning and dynamically re-scale the meta-embedding w.r.t. to the learned visual memory.

3. Our OLTR Model

We propose to map an image to a feature space such that visual concepts can easily relate to each other based on a learned metric that respects the closed-world classification while acknowledging the novelty of the open world. Our model has two main modules (Fig.2): *dynamic meta-embedding* and *modulated attention*. The former relates and transfers knowledge between head and tail classes and the latter maintains discrimination between them.

3.1. Dynamic Meta-Embedding

Our dynamic meta-embedding combines a direct image feature and an associated memory feature, with the feature norm indicating the familiarity to known classes.

Consider a convolutional neural network (CNN) with a softmax output layer for classification. The second-to-the-last layer can be viewed as the feature and the last layer a linear classifier (cf. $\phi(\cdot)$ in Fig. 2). The feature and the classifier are jointly trained from big data in an end-to-end fashion. Let v^{direct} denote the *direct feature* extracted from an input image. The final classification accuracy largely depends on the quality of this direct feature.

While a feed-forward CNN classifier works well with big training data [8, 26], it lacks sufficient supervised updates from small data in our tail classes. We propose to enrich direct feature v^{direct} with a memory feature v^{memory} that relates visual concepts in a memory module. This mechanism is similar to the memory popular in meta learning [45, 35]. We denote the resulting feature *meta embedding* v^{meta}, and it is fed to the last layer for classification. Both our memory feature v^{memory} and meta-embedding v^{meta} depend on direct feature v^{direct}.

Unlike the direct feature, the memory feature captures visual concepts from training classes, retrieved from a memory with a much shallower model.

Learning Visual Memory M**.** We follow [23] on class structure analysis and adopt discriminative centroids as the basic building block. Let M denote the visual memory of all the training data, $M = \{c_i\}_{i=1}^{K}$ where K is the number of training classes. Compared to alternatives [59, 51], this memory is appealing for our OLTR task: It is almost effortlessly and jointly learned alongside the direct features $\{v_n^{direct}\}$, and it considers both intra-class compactness and inter-class discriminativeness.

We compute centroids in two steps. **1)** Neighborhood Sampling: We sample both intra-class and inter-class examples to compose a mini-batch during training. These examples are grouped by their class labels and the centroid c_i of each group is updated by the direct feature of this mini-batch. **2)** Propagation: We alternatively update the direct feature v^{direct} and the centroids to minimize the distance between each direct feature and the centroid of its group and maximize the distance to other centroids.

Composing Memory Feature v^{memory}**.** For an input image, v^{memory} shall enhance its direct feature when there is not enough training data (as in the tail class) to learn it well. The memory feature relates the centroids in the memory, transferring knowledge to the tail class:

$$v^{memory} = o^T M := \sum_{i=1}^{K} o_i c_i, \tag{1}$$

where $o \in \mathbb{R}^K$ is the coefficients hallucinated from the direct feature. We use a lightweight neural network to obtain the coefficients from the direct feature, $o = T_{hal}(v^{direct})$.

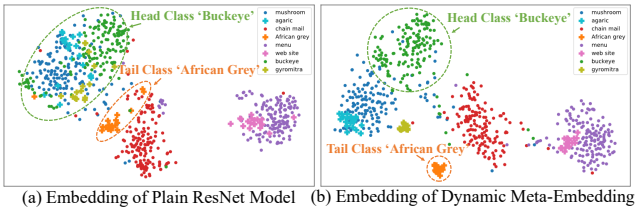
Obtaining Dynamic Meta-Embedding. v^{meta} combines the direct feature and the memory feature, and is fed to the classifier for the final class prediction (Fig. 3):

$$v^{meta} = (1/\gamma) \cdot (v^{direct} + e \otimes v^{memory}), \tag{2}$$

where $\otimes$ denotes element-wise multiplication. $\gamma > 0$ is seemingly a redundant scalar for the closed-world classification tasks. However, in the OLTR setting, it plays an important role in differentiating the examples of the training classes from those of the open-set. γ measures the reachability [46] of an input's direct feature v^{direct} to the memory M — the minimum distance between the direct feature and the discriminative centroids:

$$\gamma := \text{reachability}(v^{direct}, M) = \min_i \|v^{direct} - c_i\|_2. \tag{3}$$

When γ is small, the input likely belongs to a training class from which the centroids are derived, and a large reachability weight $1/\gamma$ is assigned to the resulting meta-embedding v^{meta}. Otherwise, the embedding is scaled down to an almost all-zero vector at the extreme. Such a property is useful for encoding open classes.



(a) Embedding of Plain ResNet Model (b) Embedding of Dynamic Meta-Embedding

Figure 3: **t-SNE feature visualization** of (a) plain ResNet model (b) our *dynamic meta-embedding*. Ours is more compact for both head and tail classes.

We now describe the concept selector e in Eq. (2). The direct feature is often good enough for the data-rich head classes, whereas the memory feature is more important for the data-poor tail classes. To adaptively select them in a soft manner, we learn a lightweight network $T_{sel}(\cdot)$ with a $\tanh(\cdot)$ activation function:

$$e = \tanh(T_{sel}(v^{direct})). \tag{4}$$

3.2. Modulated Attention

While dynamic meta-embedding facilitates feature sharing between head and tail classes, it is also vital to discriminate between them. The direct feature v^{direct}, e.g., the activation at the second-to-the-last layer in ResNet [20], is able to fulfill this requirement to some extent. However, we find it beneficial to further enhance it with spatial attention, since discriminative cues of head and tail classes seem to be distributed at different locations in the image.

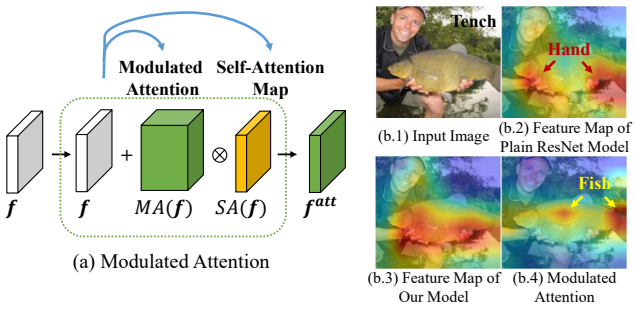
Specifically, we propose *modulated attention* to encourage samples of different classes to use different contexts. Firstly, we compute a self-attention map $SA(f)$ from the input feature map by self-correlation [55]. It is used as contextual information and added back (through skip connections) to the original feature map. The modulated attention $MA(f)$ is then designed as conditional spatial attention applied to the self-attention map: $MA(f) \otimes SA(f)$, which allows examples to select different spatial contexts (Fig. 4). The final attention feature map becomes:

$$f^{att} = f + MA(f) \otimes SA(f), \tag{5}$$

where f is a feature map in CNN, $SA(\cdot)$ is the self-attention operation, and $MA(\cdot)$ is a conditional attention function [53] with a softmax normalization. Sec. 4.1 shows empirically that our attention design achieves superior performance than the common practice of applying spatial attention to the input feature map. This modulated attention (Fig. 4b) could be plugged into any feature layer of a CNN. Here, we modify the last feature map only.

3.3. Learning

Cosine Classifier. We adopt the cosine classifier [38, 15] to produce the final classification results. Specifically, we



(a) Modulated Attention

(b.1) Input Image (b.2) Feature Map of Plain ResNet Model (b.3) Feature Map of Our Model (b.4) Modulated Attention

Figure 4: **Modulated attention** is spatial attention applied on self-attention maps ("attention on attention"). It encourages different classes to use different contexts, which helps maintain the discrimination between head and tail classes.

normalize the meta-embeddings $\{v_n^{meta}\}$, where n stands for the n-th input as well as the weight vectors $\{w_i\}_{i=1}^K$ of the classifier $\phi(\cdot)$ (no bias term):

$$v_n^{meta} = \frac{\|v_n^{meta}\|^2}{1+\|v_n^{meta}\|^2} \cdot \frac{v_n^{meta}}{\|v_n^{meta}\|}, \qquad w_k = \frac{w_k}{\|w_k\|}. \tag{6}$$

The normalization strategy for the meta-embedding is a non-linear squashing function [43] which ensures that vectors of small magnitude are shrunk to almost zeros while vectors of big magnitude are normalized to the length slightly below 1. This function helps amplify the effect of the reachability γ (cf. Eq. (2)).

Loss Function. Since all our modules are differentiable, our model can be trained end-to-end by alternatively updating the centroids $\{c_i\}_{i=1}^K$ and the *dynamic meta-embedding* v_n^{meta}. The final loss function L is a combination of the cross-entropy classification loss L_{CE} and the large-margin loss between the embeddings and the centroids L_{LM}:

$$L = \sum_{n=1}^{N} L_{CE}(v_n^{meta}, y_n) + \lambda \cdot L_{LM}(v_n^{meta}, \{c_i\}_{i=1}^K), \tag{7}$$

where λ is set to 0.1 in our experiments via observing the accuracy curve on validation set.

4. Experiments

Datasets. We curate three open long-tailed benchmarks, ImageNet-LT (object-centric), Places-LT (scene-centric), and MS1M-LT (face-centric), respectively.

1. ImageNet-LT: We construct a long-tailed version of the original ImageNet-2012 [8] by sampling a subset following the Pareto distribution with the power value α=6. Overall, it has 115.8K images from 1000 categories, with maximally 1280 images per class and minimally 5 images per class. The additional classes of images in ImageNet-2010 are used as the open set. We make the test set balanced.

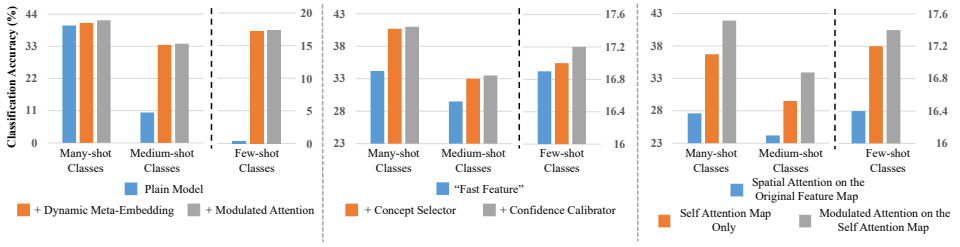
2. Places-LT: A long-tailed version of Places-2 [64] is constructed in a similar way. It contains 184.5K images from 365 categories, with the maximum of 4980 images per class and the minimum of 5 images per class. The gap between the head and tail classes are even larger than ImageNet-LT. We use the test images from Places-Extra69 as the additional open-set.

3. MS1M-LT: To create a long-tailed version of the MS1M-ArcFace dataset [16, 9], we sample images for each identity with a probability proportional to the image numbers of each identity. It results in 887.5K images and 74.5K identities, with a long-tailed distribution. To inspect the generalization ability of our approach, the performance is evaluated on the MegaFace benchmark [25], which has no identity overlap with MS1M-ArcFace.

Network Architectures. Following [18, 56, 15], we employ the scratch ResNet-10 [20] as our backbone network for ImageNet-LT. To make a fair comparison with [58], the pre-trained ResNet-152 [20] is used as the backbone network for Places-LT. For MS1M-LT, the popular pre-trained ResNet-50 [20] is the backbone network.

Evaluation Metrics. We evaluate the performance of each method under both the *closed-set* (test set contains no unknown classes) and *open-set* (test set contains unknown classes) settings to highlight their differences. Under each setting, besides the overall top-1 classification accuracy [15] over all classes, we also calculate the accuracy of three disjoint subsets: *many-shot classes* (classes each with over training 100 samples), *medium-shot classes* (classes each with 20∼100 training samples) and *few-shot classes* (classes under 20 training samples). This helps us understand the detailed characteristics of each method. For the *open-set* setting, the *F-measure* is also reported for a balanced treatment of precision and recall following [3]. For determining open classes, the $softmax$ probability threshold is initially set as 0.1, while a more detailed analysis is provided in Sec. 4.3.

Competing Methods. We choose for comparison state-of-the-art methods from different fields dealing with the open long-tailed data, including: (1) *metric learning*: Lifted Loss [36], (2) *hard negative mining*: Focal Loss [29], (3) *feature regularization*: Range Loss [63], (4) *few-shot learning*: FSLwF [15], (5) *long-tailed modeling*: Meta-ModelNet [58], and (6) *open-set detection*: Open Max [3]. We apply these methods on the same backbone networks as ours for a fair comparison. We also enable them with class-aware mini-batch sampling [50] for effective learning. Since Model Regression [57] and MetaModelNet [58] are



(a) Ablation Study of the Overall Framework (b) Ablation Study of Dynamic Meta-Embedding (c) Ablation Study of Modulated Attention

Figure 5: **Results of ablation study.** Dynamic meta-embedding contributes most on medium-shot and few-shot classes while modulated attention helps maintain the discrimination of many-shot classes. (The performance is reported with *open-set* top-1 classification accuracy on ImageNet-LT.)

Method	Error (%)
Softmax Pred. [21]	43.6
Ours	29.9
ODIN [28]†	24.6
Ours†	**18.0**

Table 2: **Open class detection error (%) comparison.** It is performed on the standard open-set benchmark, CIFAR100 + TinyImageNet (resized). "†" denotes the setting where open samples are used to tune algorithmic parameters.

the most related to our work, we directly contrast our results to the numbers reported in their paper.

4.1. Ablation Study

We firstly investigate the merit of each module in our framework. The performance is reported with *open-set* top-1 classification accuracy on ImageNet-LT.

Effectiveness of the Dynamic Meta-Embedding. Recall that the dynamic meta-embedding consists of three main components: memory feature, concept selector, and confidence calibrator. From Fig. 5 (b), we observe that the combination of the memory feature and concept selector leads to large improvements on all three shots. It is because the obtained memory feature transfers useful visual concepts among classes. Another observation is that the confidence calibrator is the most effective on few-shot classes. The reachability estimation inside the confidence calibrator helps distinguish tail classes from open classes.

Effectiveness of the Modulated Attention. We observe from Fig. 5 (a) that, compared to medium-shot classes, the modulated attention contributes more to the discrimination between many-shot and few-shot classes. Fig. 5 (c) further validates that the modulated attention is more effective than directly applying spatial attention on feature maps. It implies that adaptive contexts selection is easier to learn than the conventional feature selection.

Effectiveness of the Reachability Calibration. To further demonstrate the merit of reachability calibration for open-world setting, we conduct additional experiments following the standard settings in [21, 28] (CIFAR100 + TinyImageNet(resized)). The results are listed in Table 2, where our approach shows favorable performance over standard open-set methods [21, 28].

4.2. Result Comparisons

We extensively evaluate the performance of various representative methods on our benchmarks.

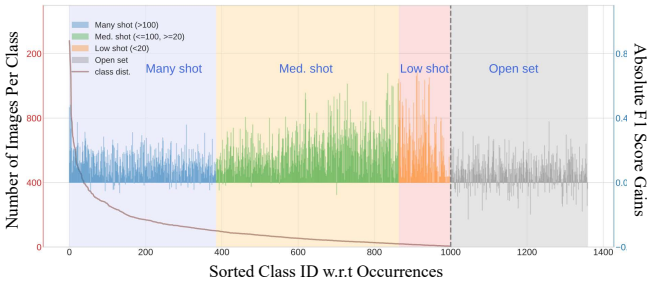


Figure 6: **The absolute F1 score of our method over the plain model.** Ours has across-the-board performance gains w.r.t. many/medium/few-shot and open classes.

ImageNet-LT. Table 3 (a) shows the performance comparison of different methods. We have the following observations. Firstly, both Lifted Loss [36] and Focal Loss [29] greatly boost the performance of few-shot classes by enforcing feature regularization. However, they also sacrifice the performance on many-shot classes since there are no built-in mechanism of adaptively handling samples of different shots. Secondly, OpenMax [3] improves the results under the open-set setting. However, the accuracy degrades when it is evaluated with *F-measure*, which considers both precision and recall in open-set. When the open classes are compounded with the tail classes, it becomes challenging to perform the distribution fitting that [3] requires. Lastly, though the few-shot learning without forgetting approach [15] retains the many-shot class accuracy, it has difficulty dealing with the imbalanced base classes which are lacked in the current few-shot paradigm. As demonstrated in Fig. 6, our approach provides a comprehensive treatment to all the many/medium/few-shot classes as well as the open classes, achieving substantial improvements on all aspects.

Places-LT. Similar observations can be made on the Places-LT benchmark as shown in Table 3 (b). With a much stronger baseline (*i.e.* pre-trained ResNet-152), our approach still consistently outperforms other alternatives un-

Backbone Net	closed-set setting				open-set setting			
ResNet-10	> 100	$\leqslant 100$ & > 20	< 20		> 100	$\leqslant 100$ & > 20	< 20	
Methods	**Many-shot**	**Medium-shot**	**Few-shot**	**Overall**	**Many-shot**	**Medium-shot**	**Few-shot**	**F-measure**
Plain Model [20]	40.9	10.7	0.4	20.9	40.1	10.4	0.4	0.295
Lifted Loss [36]	35.8	30.4	17.9	30.8	34.8	29.3	17.4	0.374
Focal Loss [29]	36.4	29.9	16	30.5	35.7	29.3	15.6	0.371
Range Loss [63]	35.8	30.3	17.6	30.7	34.7	29.4	17.2	0.373
+ OpenMax [3]	-	-	-	-	35.8	30.3	**17.6**	0.368
FSLwF [15]	40.9	22.1	15	28.4	40.8	21.7	14.5	0.347
Ours	**43.2**	**35.1**	**18.5**	**35.6**	**41.9**	**33.9**	17.4	**0.474**

(a) Top-1 classification accuracy on ImageNet-LT.

Backbone Net	closed-set setting				open-set setting			
ResNet-152	> 100	$\leqslant 100$ & > 20	< 20		> 100	$\leqslant 100$ & > 20	< 20	
Methods	**Many-shot**	**Medium-shot**	**Few-shot**	**Overall**	**Many-shot**	**Medium-shot**	**Few-shot**	**F-measure**
Plain Model [20]	**45.9**	22.4	0.36	27.2	**45.9**	22.4	0.36	0.366
Lifted Loss [36]	41.1	35.4	24	35.2	41	35.2	23.8	0.459
Focal Loss [29]	41.1	34.8	22.4	34.6	41	34.8	22.3	0.453
Range Loss [63]	41.1	35.4	23.2	35.1	41	35.3	23.1	0.457
+ OpenMax [3]	-	-	-	-	41.1	35.4	23.2	0.458
FSLwF [15]	43.9	29.9	**29.5**	34.9	38.1	19.5	14.8	0.375
Ours	44.7	**37**	25.3	**35.9**	44.6	**36.8**	**25.2**	**0.464**

(b) Top-1 classification accuracy on Places-LT.

Table 3: **Benchmarking results on (a) ImageNet-LT and (b) Places-LT.** Our approach provides a comprehensive treatment to all the many/medium/few-shot classes as well as the open classes, achieving substantial advantages on all aspects.

Backbone Net	MegaFace Identification Rate						
ResNet-50	$\geqslant 5$	< 5 & $\geqslant 2$	< 2 & $\geqslant 1$	$= 0$		Sub-Groups	
Methods	**Many-shot**	**Few-shot**	**One-shot**	**Zero-shot**	**Full Test**	**Male**	**Female**
Plain Model [20]	80.64	71.98	84.60	77.72	73.88	78.30	78.70
Range Loss [63]	78.60	71.36	83.14	77.40	72.17	-	-
Ours	**80.82**	**72.44**	**87.60**	**79.50**	**74.51**	**79.04**	**79.08**

Method	**Acc.**
Plain Model [20]	48.0
Cost-Sensitive [24]	52.4
Model Reg. [57]	54.7
MetaModelNet [58]	57.3
Ours	**58.7**

Table 4: **Benchmarking results on MegaFace (*left*) and SUN-LT (*right*).** Our approach achieves the best performance on natural-world datasets when compared to other state-of-the-art methods. Furthermore, our approach achieves across-board improvements on both 'male' and 'female' sub-groups.

der both the closed-set and open-set settings. The advantage is even more profound under the *F-measure*.

MS1M-LT. We train on the MS1M-LT dataset and report results on the MegaFace identification track, which is a standard benchmark in the face recognition field. Since the face identities in the training set and the test set are disjoint, we adopt an indirect way to partition the testing set into the subsets of different shots. We approximate the pseudo shots of each test sample by counting the number of training samples that are similar to it by at least a threshold (feature similarity greater than 0.7). Apart from many-shot, few-shot, one-shot subsets, we also obtain a zero-shot subset, for which we cannot find any sufficiently similar samples in the training set. It can be observed that our approach has the most advantage on one-shot identities (3.0% gains) and zero-shot identities (1.8% gains) as shown in Table 4 (*left*).

SUN-LT. To directly compare with [57] and [58], we also test on the SUN-LT benchmark they provided. The final results are listed in Table 4 (*right*). Instead of learning a series of classifier transformations, our approach transfers visual knowledge among features and achieves a 1.4% improvement over the prior best. Note that our approach also incurs much less computational cost since MetaModelNet [58] requires a recursive training procedure.

Indication for Fairness. Here we report the sensitive attribute performance on MS1M-LT. The last two columns in Table 4 show that our approach achieves across-board improvements on both 'male' and 'female' sub-groups, which has an implication for effective fairness learning.

4.3. Further Analysis

Finally we visualize and analyze some influencing aspects in our framework as well as typical failure cases.

What memory feature has Infused. Here we inspect the visual concepts that memory feature has infused by visualizing its top activating neurons as shown in Fig. 7. Specifically, for each input image, we identify its top-3 transferred neurons in memory feature. And each neuron is

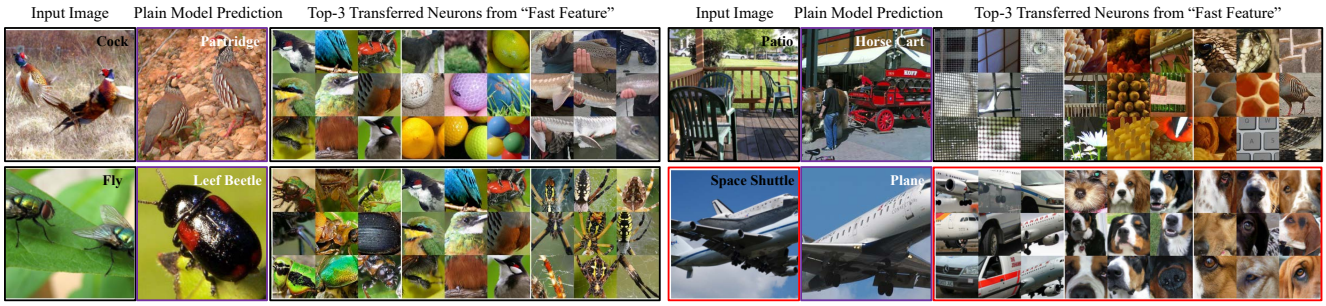


Figure 7: **Examples of the top-3 infused visual concepts from memory feature.** Except for the bottom right failure case (marked in red), all the other three input images are misclassified by the plain model and correctly classified by our model. For example, to classify the top left image which belongs to a tail class 'cock', our approach has learned to transfer visual concepts that represents "bird head", "round shape" and "dotted texture" respectively.

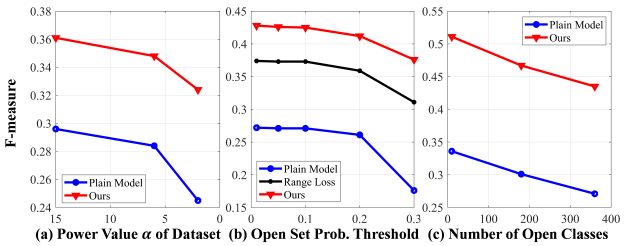


Figure 8: **The influence of (a) dataset longtail-ness, (b) open-set probability threshold, and (c) the number of open classes.** As the dataset becomes more imbalanced, our approach only undergoes a moderate performance drop. Our approach also demonstrates great robustness to the contamination of open classes.

visualized by a collection of highest activated patches [61] over the whole training set. For example, to classify the top left image which belongs to a tail class 'cock', our approach has learned to transfer visual concepts that represents "bird head", "round shape" and "dotted texture" respectively. After feature infusion, the dynamic meta-embedding becomes more informative and discriminative.

Influence of Dataset Longtail-ness. The longtail-ness of the dataset (*e.g.* the degree of imbalance of the class distribution) could have an impact on the model performance. For faster investigating, here the weights of the backbone network are freezed during training. From Fig. 8 (a), we observe that as the dataset becomes more imbalanced (*i.e.* power value α decreases), our approach only undergoes a moderate performance drop. Dynamic meta-embedding enables effective knowledge transfer among data-abundant and data-scarce classes.

Influence of Open-Set Prob. Threshold. The performance change w.r.t. the open-set probability threshold is demonstrated in Fig. 8 (b). Compared to the plain model [20] and range loss [63], the performance of our approach changes steadily as the open-set threshold rises. The reachability estimator in our framework helps calibrate the sample confidence, thus enhancing robustness to open classes.

Influence of the Number of Open Classes. Finally we investigate performance change w.r.t. the number of open classes. Fig. 8 (c) indicates that our approach demonstrates great robustness to the contamination of open classes.

Failure Cases. Since our approach encourages the feature infusion among classes, it slightly sacrifices the fine-grained discrimination for the promotion of under-representative classes. One typical failure case of our approach is the confusion between many-shot and medium-shot classes. For example, the bottom right image in Fig. 7 is misclassified into 'airplane' because some cross-category traits like "nose shape" and "eye shape" are infused. We plan to explore feature disentanglement [5] to alleviate this trade-off issue.

5. Conclusions

We introduce the OLTR task that learns from natural long-tail open-end distributed data and optimizes the overall accuracy over a balanced test set. We propose an integrated OLTR algorithm, dynamic meta-embedding, in order to share visual knowledge between head and tail classes and to reduce confusion between tail and open classes. We validate our method on three curated large-scale OLTR benchmarks (ImageNet-LT, Places-LT and MS1M-LT). Our publicly available code and data would enable future research that is directly transferable to real-world applications.

Acknowledgements. This research was supported, in part, by SenseTime Group Limited, NSF IIS 1835539, Berkeley Deep Drive, DARPA, and US Government fund through Etegent Technologies on Low-Shot Detection in Remote Sensing Imagery. The views, opinions and/or findings expressed are those of the author and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

References

[1] Lisa Anne Hendricks, Kaylee Burns, Kate Saenko, Trevor Darrell, and Anna Rohrbach. Women also snowboard: Overcoming bias in captioning models. In *ECCV*, 2018. 11

[2] Jimmy Ba, Geoffrey E Hinton, Volodymyr Mnih, Joel Z Leibo, and Catalin Ionescu. Using fast weights to attend to the recent past. In *NIPS*, 2016. 2, 3

[3] Abhijit Bendale and Terrance E Boult. Towards open set deep networks. In *CVPR*, 2016. 3, 5, 6, 7, 14

[4] Samy Bengio. The battle against the long tail. In *Talk on Workshop on Big Data and Statistical Machine Learning*, 2015. 2

[5] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *TPAMI*, 2013. 8

[6] Luca Bertinetto, João F Henriques, Jack Valmadre, Philip Torr, and Andrea Vedaldi. Learning feed-forward one-shot learners. In *NIPS*, 2016. 3

[7] Yin Cui, Yang Song, Chen Sun, Andrew Howard, and Serge Belongie. Large scale fine-grained categorization and domain-specific transfer learning. In *CVPR*, 2018. 2

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 1, 3, 5

[9] Jiankang Deng, Jia Guo, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. *arXiv preprint arXiv:1801.07698*, 2018. 5, 13

[10] Terrance DeVries and Graham W Taylor. Learning confidence for out-of-distribution detection in neural networks. *arXiv preprint arXiv:1802.04865*, 2018. 3

[11] Qi Dong, Shaogang Gong, and Xiatian Zhu. Class rectification hard mining for imbalanced deep learning. In *ICCV*, 2017. 2

[12] Yan Duan, John Schulman, Xi Chen, Peter L Bartlett, Ilya Sutskever, and Pieter Abbeel. Rl^2: Fast reinforcement learning via slow reinforcement learning. *arXiv preprint arXiv:1611.02779*, 2016. 2, 3

[13] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *The 3rd innovations in theoretical computer science conference*, 2012. 11

[14] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. *arXiv preprint arXiv:1703.03400*, 2017. 3

[15] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *CVPR*, 2018. 3, 4, 5, 6, 7, 13

[16] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *ECCV*, 2016. 5

[17] David Ha, Andrew Dai, and Quoc V Le. Hypernetworks. *arXiv preprint arXiv:1609.09106*, 2016. 2

[18] Bharath Hariharan and Ross B Girshick. Low-shot visual recognition by shrinking and hallucinating features. In *ICCV*, 2017. 1, 3, 5, 12

[19] Haibo He and Edwardo A Garcia. Learning from imbalanced data. *TKDE*, 2008. 2

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1, 4, 5, 7, 8

[21] Dan Hendrycks and Kevin Gimpel. Baseline for detecting misclassified and out-of-distribution examples in neural networks. In *ICLR*, 2017. 6

[22] Geoffrey E Hinton and David C Plaut. Using fast weights to deblur old memories. In *Proceedings of the ninth annual conference of the Cognitive Science Society*, 1987. 3

[23] Yen-Chang Hsu, Zhaoyang Lv, and Zsolt Kira. Learning to cluster in order to transfer across domains and tasks. *arXiv preprint arXiv:1711.10125*, 2017. 4

[24] Chen Huang, Yining Li, Chen Change Loy, and Xiaoou Tang. Learning deep representation for imbalanced classification. In *CVPR*, 2016. 2, 7

[25] Ira Kemelmacher-Shlizerman, Steven M Seitz, Daniel Miller, and Evan Brossard. The megaface benchmark: 1 million faces for recognition at scale. In *CVPR*, 2016. 5

[26] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 1, 3

[27] Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 2015. 1

[28] Shiyu Liang, Yixuan Li, and R Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *ICLR*, 2018. 3, 6

[29] Tsung-Yi Lin, Priyal Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017. 2, 5, 6, 7

[30] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 1

[31] Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *CVPR*, 2016. 2

[32] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, 2015. 2, 14

[33] David Madras, Elliot Creager, Toniann Pitassi, and Richard Zemel. Learning adversarially fair and transferable representations. *arXiv preprint arXiv:1802.06309*, 2018. 11

[34] Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. Model cards for model reporting. *arXiv preprint arXiv:1810.03993*, 2018. 11

[35] Tsendsuren Munkhdalai and Hong Yu. Meta networks. *arXiv preprint arXiv:1703.00837*, 2017. 3

[36] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *CVPR*, 2016. 2, 5, 6, 7

[37] Wanli Ouyang, Xiaogang Wang, Cong Zhang, and Xiaokang Yang. Factors in finetuning deep model for object detection with long-tail distribution. In *CVPR*, 2016. 2

[38] Hang Qi, Matthew Brown, and David G Lowe. Low-shot learning with imprinted weights. In *CVPR*, 2018. 3, 4

[39] Siyuan Qiao, Chenxi Liu, Wei Shen, and Alan Yuille. Few-shot image recognition by predicting parameters from activations. In *CVPR*, 2018. 3

[40] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *ICLR*, 2017. 3

[41] William J Reed. The pareto, zipf and other power laws. *Economics letters*, 2001. 1, 12

[42] Mengye Ren, Renjie Liao, Ethan Fetaya, and Richard S Zemel. Incremental few-shot learning with attention attractor networks. *arXiv preprint arXiv:1810.07218*, 2018. 3

[43] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. Dynamic routing between capsules. In *NIPS*, 2017. 5

[44] Ruslan Salakhutdinov, Antonio Torralba, and Josh Tenenbaum. Learning to share visual appearance for multiclass object detection. In *CVPR*, 2011. 2

[45] Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy Lillicrap. Meta-learning with memory-augmented neural networks. In *ICML*, 2016. 3

[46] Nikolay Savinov, Anton Raichuk, Raphaël Marinier, Damien Vincent, Marc Pollefeys, Timothy Lillicrap, and Sylvain Gelly. Episodic curiosity through reachability. *arXiv preprint arXiv:1810.02274*, 2018. 4

[47] Walter J Scheirer, Anderson de Rezende Rocha, Archana Sapkota, and Terrance E Boult. Toward open set recognition. *TPAMI*, 2013. 3

[48] Jürgen Schmidhuber. Learning to control fast-weight memories: An alternative to dynamic recurrent networks. *Neural Computation*, 1992. 3

[49] Jürgen Schmidhuber. A neural network that embeds its own meta-levels. In *ICNN*, 1993. 3

[50] Li Shen, Zhouchen Lin, and Qingming Huang. Relay backpropagation for effective learning of deep convolutional neural networks. In *ECCV*, 2016. 5

[51] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NIPS*, 2017. 1, 3, 4

[52] Grant Van Horn and Pietro Perona. The devil is in the tails: Fine-grained classification in the wild. *arXiv preprint arXiv:1709.01450*, 2017. 2

[53] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, 2017. 4

[54] Oriol Vinyals, Charles Blundell, Tim Lillicrap, and Daan Wierstra. Matching networks for one shot learning. In *NIPS*, 2016. 1, 2, 3

[55] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. *arXiv preprint arXiv:1711.07971*, 2017. 2, 4

[56] Yu-Xiong Wang, Ross Girshick, Martial Hebert, and Bharath Hariharan. Low-shot learning from imaginary data. *arXiv preprint arXiv:1801.05401*, 2018. 3, 5

[57] Yu-Xiong Wang and Martial Hebert. Learning to learn: Model regression networks for easy small sample learning. In *ECCV*, 2016. 5, 7

[58] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Learning to model the tail. In *NIPS*, 2017. 1, 2, 5, 7, 13

[59] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *ECCV*, 2016. 4

[60] Flood Sung Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *CVPR*, 2018. 3

[61] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *ECCV*, 2014. 8

[62] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In *ICML*, 2013. 11

[63] Xiao Zhang, Zhiyuan Fang, Yandong Wen, Zhifeng Li, and Yu Qiao. Range loss for deep face recognition with long-tailed training data. In *CVPR*, 2017. 2, 5, 7, 8

[64] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *TPAMI*, 2018. 5

[65] Xiangxin Zhu, Dragomir Anguelov, and Deva Ramanan. Capturing long-tail distributions of object subcategories. In *CVPR*, 2014. 2

[66] Xiangxin Zhu, Carl Vondrick, Charless C Fowlkes, and Deva Ramanan. Do we need more training data? *IJCV*, 2016. 2

Appendices

In this supplementary material, we provide details omitted in the main text including:

- Section A: intuitive explanation of our approach (Sec. 1 "Introduction" of the main paper.)
- Section B: relation to fairness analysis (Sec. 2 "Related Work" of the main paper.)
- Section C: more methodology details (Sec. 3 "Approach" of the main paper.)
- Section D: detailed experimental setup (Sec. 4 "Experiments" of the main paper.)
- Section E: additional visualization of our approach (Sec. 4.3 "Further Analysis" of the main paper.)

A. Intuitive Explanation of Our Approach

In this section, we give an intuitive explanation of our approach that tackles the problem open long-tail recognition. From the perspective of knowledge gained from observation (*i.e.* training set), head classes, tail classes and open classes form a continuous spectrum as illustrated in Fig. 9.

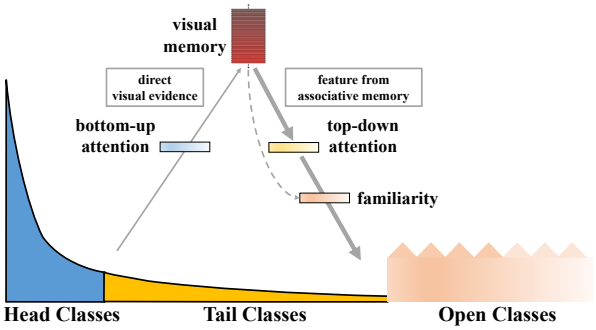


Figure 9: **Intuition explanation of our approach.**

Direct + Memory Feature	Modulated Attention	Reachability Module
Transfer knowledge between head/tail classes	Maintain discrimination between head/tail classes	Deal with open classes

Table 5: **The effects of each component in our approach.**

Firstly, we obtain a *visual memory* by aggregating the knowledge from both head and tail classes. Then the visual concepts stored in the memory are infused back as associated memory feature to enhance the original direct feature. It can be understood as using induced knowledge (*i.e.* memory feature) to assist the direct observation (*i.e.* direct feature). We further learn a *concept selector* to control the amount and type of memory feature to be infused. Since head classes already have abundant direct observation, only a small amount of memory feature is infused for them. On the contrary, tail classes suffer from scarce observation, the associated visual concepts in memory feature are extremely beneficial. Finally, we calibrate the confidence of open classes by calculating their *reachability* to the obtained visual memory. In this way, we provide a comprehensive treatment to the full spectrum of head, tail and open classes, improving the performance on all categories. To summarize, the effects of each component in our approach are listed in Table 5.

B. Relation to Fairness Analysis

The open long-tail recognition proposed in our work also has an intrinsic relationship to fairness analysis [13, 62, 33, 34, 1]. Their key differences are listed in Table 6. On the problem setting side, both open long-tail recognition and fairness analysis aim to tackle the imbalance existed in real-world data. Open long-tail recognition focuses on the longtail-ness in both known and unknown categories while fairness analysis deals with the bias in sensitive attributes such as male/female and white/black.

On the methodology side, both open long-tail recognition and fairness analysis aim to learn transferable representations. Open long-tail recognition optimizes for the overall accuracy of all categories while fairness analysis optimizes for several attribute-wise criteria. The preliminary results in Table 4 demonstrates that our proposed dynamic meta-embedding is also a promising solution to fairness analysis.

Problem	Imbalanced Asp.	Optimization Obj.
fairness analysis	sensitive attributes	attribute-wise criteria
open long-tail recog.	categories	acc. on all categories

Table 6: **Key differences between fairness analysis and open long-tail recognition.** "asp." stands for aspects while "obj." stands for objectives.

C. More Methodology Details

Notation Summary. We summarize the notations used in the paper in Table 7.

Notation	Meaning
x	input image
y	category label
f	the original feature map
f^{att}	feature map after modulated attention
$F(\cdot)$	feature extractor
$\phi(\cdot)$	classifier
c_i	discriminative centroid
G	local graph
M	visual memory
v^{direct}	direct feature
v^{memory}	memory feature
o	hallucinated coefficients from visual memory
e	concept selector
γ	confidence calibrator
v^{meta}	dynamic meta-embedding

Table 7: **Summary of notations.**

Obtaining Discriminative Centroids. The step-by-step procedure for obtaining discriminative centroids $\{c_i\}_{i=1}^{K}$ is further illustrated in Fig. 11.

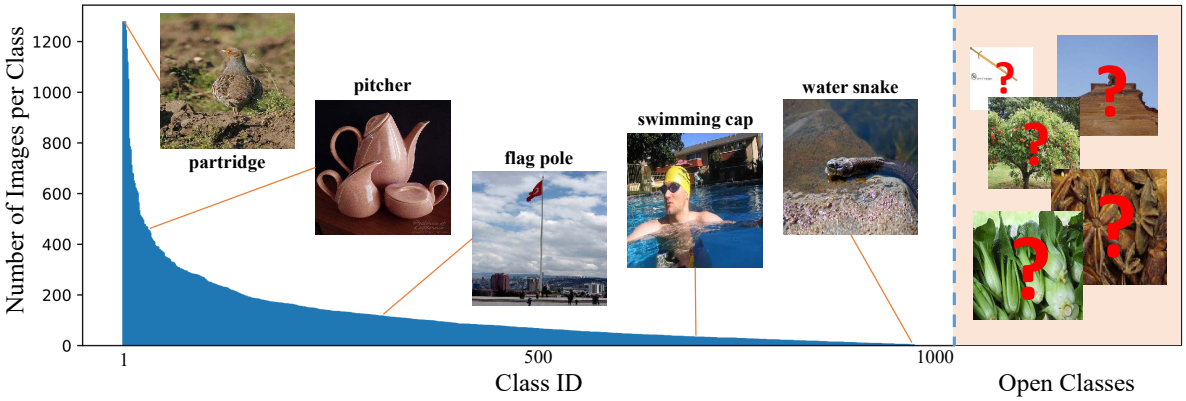


Figure 10: **The dataset statistics of ImageNet-LT.**

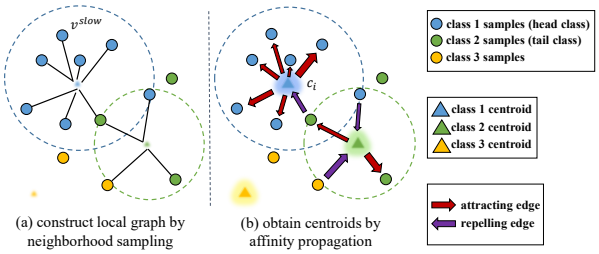


Figure 11: **The discriminative centroids constitute our visual memory**, which are obtained with two iterative steps, neighborhood sampling and affinity propagation.

Detailed Loss Functions. Here we elaborate the two loss functions L_{CE} and L_{LM} described in Eqn. 7 in the main paper. Specifically, L_{CE} is the cross-entropy loss between dynamic meta-embedding v_n^{meta} and the ground truth category label y_n:

$$L_{CE}(v_n^{meta}, y_n) = y_n \log(\phi(v_n^{meta})) + (1 - y_n) \log(1 - \phi(v_n^{meta})), \tag{8}$$

where $\phi(\cdot)$ is the cosine classifier described in Eqn. 6 in the main paper. Next we introduce the large margin loss L_{LM} between the embedding v_n^{meta} and the centroids $\{c_i\}_{i=1}^K$:

$$L_{LM}(v_n^{meta}, \{c_i\}_{i=1}^K) = \max(0, \sum_{i=y_n} \|v_n^{meta} - c_i\| - \sum_{i \neq y_n} \|v_n^{meta} - c_i\| + m), \tag{9}$$

where m is the margin and we set it as 5.0 in our experiments. With this formulation, we minimize the distance between each embedding and the centroid of its group and meanwhile maximize the distance between the embedding and the centroids it does not belong to.

D. Experimental Setup

D.1. Open Long-Tail Dataset Preparation

ImageNet-LT. The training data set was generated using a Pareto distribution [41] with a power value α=6 and 1,280∼5 images per class from the 1000 classes of ImageNet dataset. Images were randomly selected based on the distribution values of each class. The classes were sorted following the benchmark proposed by Bharath & Girshick [18], where the 1000 classes were randomly split into 389 base classes and 611 novel classes. The first 389 largest classes in ImageNet-LT are the same as the base classes in the benchmark, and the rest 611 classes are the same as the novel classes. We randomly selected 20 training images per class from the origin training set as validation set. The original validation set of ImageNet was used as testing set in this paper. The dataset specifications are shown in Fig. 10.

Places-LT. The training data set was generated similarly to ImageNet-LT using a Pareto distribution with a power value α=6 and 4,980∼5 images per class from the 365 classes of Places-365-standard data set. We used the distribution order of Places-365-challenge data set (which is imbalanced) to sort the training data classes. We also randomly selected 20 images per class from the original training set as validation set. The original validation set of Places-365 was used as testing set in this paper. The dataset specifications are shown in Fig. 12.

MS1M-LT. This dataset was generated from a large-scale face recognition dataset, named MS1M-ArcFace. The original dataset contains about 5.8M images with 85K identities. To create a long-tail version, we sampled images for each identity with a probability proportional to the image numbers of each identity. It results in 887.5K images and 74.5K identities, with a long-tail distribution.

For the evaluation set, MegaFace is one of the largest face recognition benchmarks. It contains 3,530 images from FaceScrub dataset as a probe set and 1M images as a gallery set. The identification task is to find top-1 nearest image

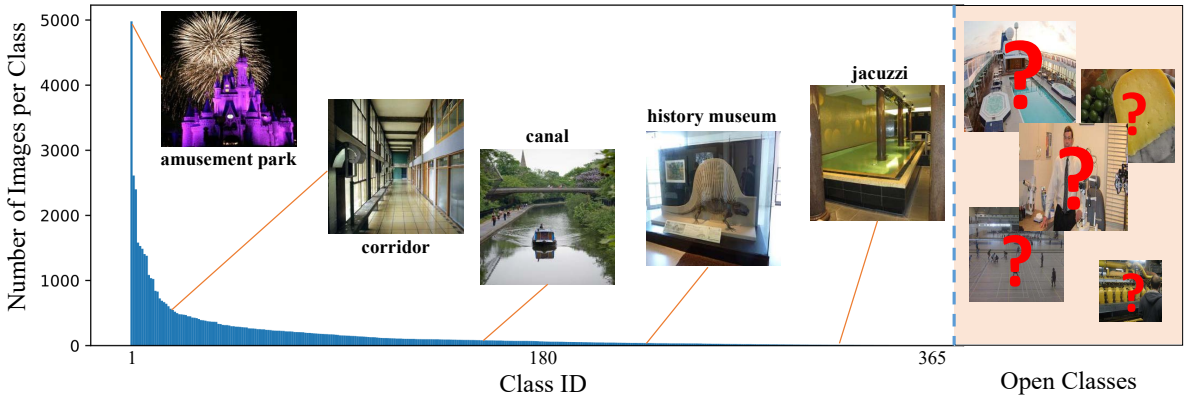


Figure 12: **The dataset statistics of Places-LT.**

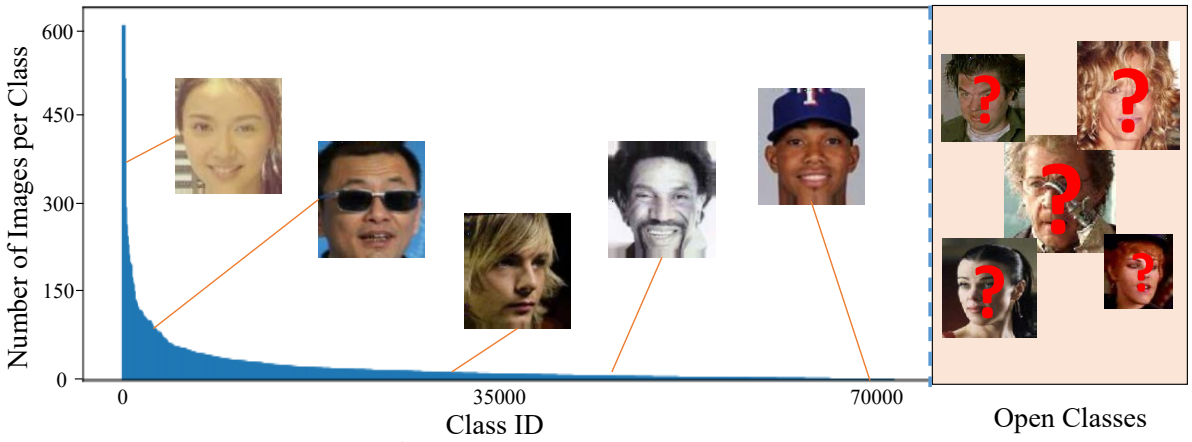


Figure 13: **The dataset statistics of MS1M-LT.**

from the 1M gallery for each sample in the probe set. Then the identification rate is the mean of hit rates. Since the identities in training set and testing set are non-overlapped, we adopt an indirect way to partition the testing set into subsets with different shots. We approximate the pseudo occurrences of each test sample by counting the number of the similar (similarity greater than 0.7) training samples. The similarity is calculated as the feature distance produced by a state-of-the-art face recognition system [9]. Apart from many-shot, few-shot and one-shot subset, we also define a zero-shot subset, for which we cannot find similar samples in the training set. The dataset specifications are shown in Fig. 13.

SUN-LT. We used the same training and testing data set as provided by [58], where there were 1,132∼1 images per class in the training set and 40 images per class in the testing set. We randomly selected 5 images from un-used training data as our validation set.

D.2. Data Pre-processing

All the images were firstly resized to 256×256. During training, the images were randomly cropped to 224×224, then augmented with random horizontal flip at probability $p = 0.5$ and random color jitter on brightness, contrast, and saturation with jitter factor of 0.4. During validation and testing, images were center cropped to 224×224 without further augmentation.

D.3. Training Details

ImageNet-LT. The feature extractor model used in the experiments on ImageNet-LT was a ResNet-10 model initialized from scratch (i.e., random initialization). All different classifiers were also initialized from scratch. Some major hyper-parameters can be found in Table 8.

Places-LT & SUN-LT. We used a two-stage training protocol following [15] when conducting experiments on both Places-LT and SUN-LT. (1) In the first stage, we used the ImageNet pre-trained ResNet-152 feature model with a dot-product classifier to fine-tune on the training data of Places-LT and SUN-LT. (2) In the second stage, we used the Places-LT/SUN-LT pre-trained model as our feature model and freezed the convolutional weights. Finally we fine-tuned the classifiers initialized from scratch to produce the experimental results. Some major hyper-parameters can be found in Table 8.

MS1M-LT. We used the ImageNet pre-trained ResNet-50 with a linear classifier and cross-entropy loss to train the

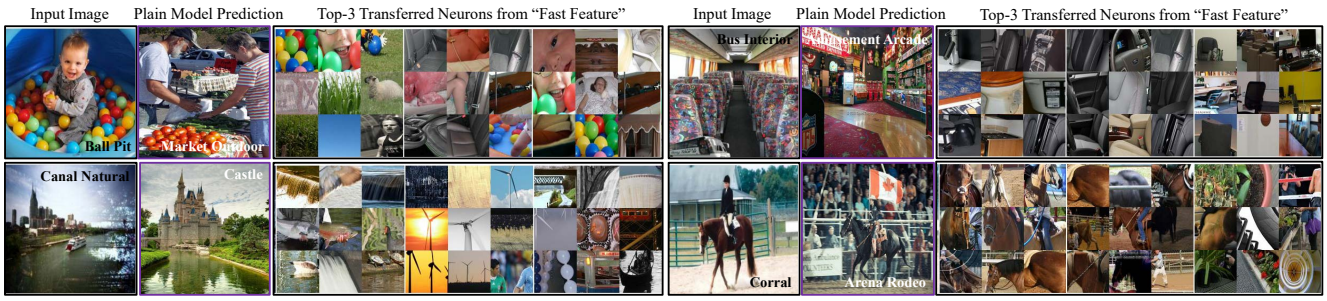


Figure 14: **Examples of the infused visual concepts from memory feature in Places-LT.**

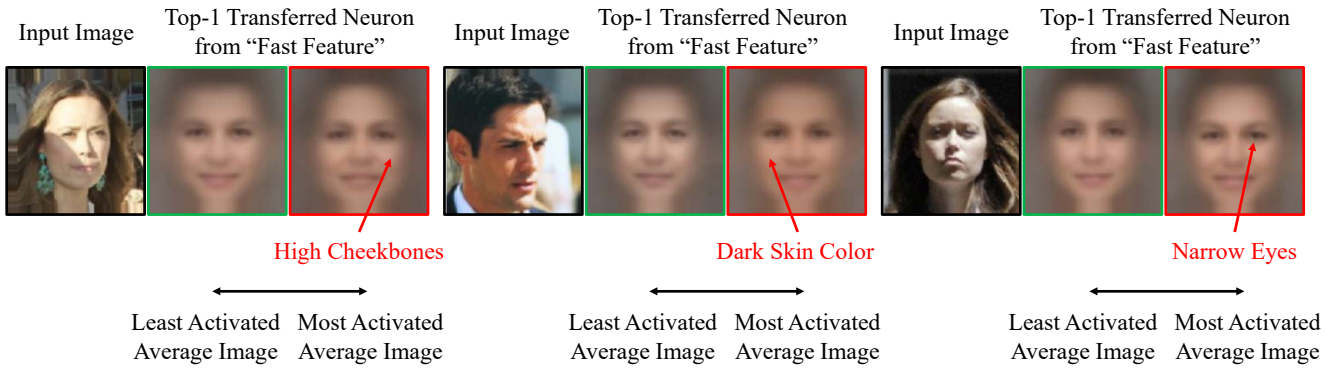


Figure 15: **Examples of the infused visual concepts from memory feature in MS1M-LT.**

face recognition model. Some major hyper-parameters can be found in Table 8.

Dataset	Initial LR.	Epoch	LR. Schedule
ImageNet-LT	0.1	30	drop 10% every 10 epochs
Places-LT	0.01	30	drop 10% every 10 epochs
MS1M-LT	0.01	30	drop 10% every 10 epochs

Table 8: The major hyper-parameters used in our experiments. "LR." stands for learning rate.

D.4. Evaluation Protocols

Top-1 Classification Accuracy. For ImageNet-LT, Places-LT, and SUN-LT, since the testing sets are balanced, the top-1 classification accuracy are calculated as the mean accuracy over all close-set categories with the contamination of open classes. All open classes are regared as one unknown class. Predictions of data are obtained as the classes with the highest $softmax$ probabilities.

F-measure. Following [3], the F-measure (F) is calculated as 2 times the product of precision (p) and recall (r) divided by the sum of p and r:

$$F = 2 \cdot \frac{p \cdot r}{p + r}. \tag{10}$$

p is calculated as true positive (T_p, defined as correct predictions on the closed testing set) over the sum of T_p and false positive (F_p, defined as incorrect predictions on closed testing set):

$$p = \frac{T_p}{T_p + F_p}. \tag{11}$$

r is calculated as T_p over the sum of T_p and false negative (F_n, defined as number of images from the open set that are predicted as known categories):

$$r = \frac{T_p}{T_p + F_n}. \tag{12}$$

E. More Visualization

Memory Feature in Places-LT. We visualize the memory feature in Places-LT similarly to ImageNet-LT as described in Sec. 4.3 in the main paper. Examples of the infused visual concepts from memory feature in Places-LT are presented in Fig. 14. We observe that memory feature indeed encodes discriminative visual traits for the underlying scene.

Memory Feature in MS-1M. Following [32], we visualize the memory feature in MS1M-LT by contrasting the least activated average image and the most activated average image of the top firing neuron. From Fig. 15, we observe that memory feature in MS1M-LT infuses several identity-related attributes (*e.g.* "high cheekbones", "dark skin color" and "narrow eyes") for precise recognition.