GistNet: a Geometric Structure Transfer Network for Long-Tailed Recognition

Bo Liu
UC, San Diego
boliu@ucsd.edu

Haoxiang Li
Wormpex AI Research
lhxustcer@gmail.com

Hao Kang
Wormpex AI Research
haokheseri@gmail.com

Gang Hua
Wormpex AI Research
ganghua@gmail.com

Nuno Vasconcelos
UC, San Diego
nuno@ece.ucsd.edu

Abstract

The problem of long-tailed recognition, where the number of examples per class is highly unbalanced, is considered. It is hypothesized that the well known tendency of standard classifier training to overfit to popular classes can be exploited for effective transfer learning. Rather than eliminating this overfitting, e.g. by adopting popular class-balanced sampling methods, the learning algorithm should instead leverage this overfitting to transfer geometric information from popular to low-shot classes. A new classifier architecture, GistNet, is proposed to support this goal, using constellations of classifier parameters to encode the class geometry. A new learning algorithm is then proposed for GeometrIc Structure Transfer (GIST), with resort to a combination of loss functions that combine class-balanced and random sampling to guarantee that, while overfitting to the popular classes is restricted to geometric parameters, it is leveraged to transfer class geometry from popular to few-shot classes. This enables better generalization for few-shot classes without the need for the manual specification of class weights, or even the explicit grouping of classes into different types. Experiments on two popular long-tailed recognition datasets show that GistNet outperforms existing solutions to this problem.

1. Introduction

The availability of large-scale datasets, with large numbers of images per class [3], has been a major factor in the success of deep learning for tasks such as object recognition. However, these datasets are manually curated and artificially balanced. This is unlike most real world applications, where the frequencies of examples from different classes can be highly unbalanced, leading to skewed distributions with long tails.

This has motivated recent interest in the problem of long-tailed recognition [13], where the training data is highly unbalanced but the test set is kept balanced, so that equally good performance on all classes is crucial to achieve high overall accuracy.

Success in the long-tailed recognition setting requires specific handling of class imbalance during training, since a classifier trained with the standard cross-entropy loss will overfit to highly populated classes and perform poorly on low-shot classes. This has motivated several works to fight class overfitting with methods, like data re-sampling [27] or cost-sensitive losses [10], that place more training emphasis on the examples of lower populated classes.

It is, however, difficult to design augmentation or class weighting schemes that do not either under or over-emphasize the few-shot classes. In this work, we seek an approach that is fully data driven and *leverages* overfitting to the popular classes, rather than combat it. The idea is to transfer some properties of these classes, which are well learned by the standard classifier, to the classes with insufficient data, where this is not possible.

For this, we leverage the interpretation of a deep classifier as the composition of an embedding, or feature extractor, implemented with several neural network layers and a parametric classifier, implemented with a logistic regression layer, at the top of the network. While the embedding is shared by all classes, the classifier parameters are class-specific, namely a weight-vector per class, as shown in Figure 1.

We exploit the fact that the configuration of these weight vectors determines the geometry of the embedding. This consists of the class-conditional distribution, and associated metric, of the feature vectors of each class, which define the class boundaries. For a well learned network, this geometry is identical for all classes. In the long-tailed setting, the geometry is usually well learned for many-shot classes, but not for classes with insufficient training samples, as shown in the left of Figure 1.

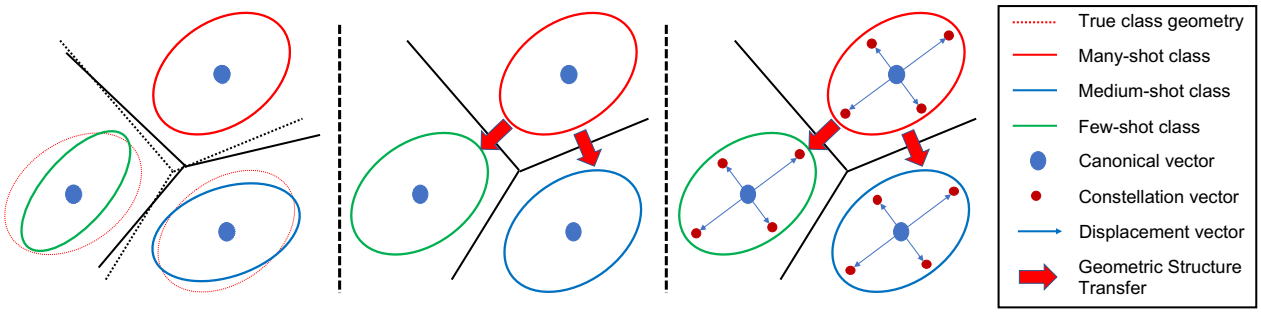


Figure 1. Left: in long-tailed recognition, the small number of samples from medium- and few-shot classes make it difficult to learn their geometry, leading to inaccurate class boundaries. This is unlike many-shot classes, whose natural geometry can usually be learned. Middle: the boundaries are corrected by transferring the geometric structure of the many-shot classes to the classes with few examples. Right: GistNet implements geometric structure transfer by implementing constellations of classification parameters. These consist of a class-specific center and a set of displacements shared by all classes. Under GIST training, these tend to follow the natural geometry of the many-shot classes, which is transferred to the medium- and few-shot classes.

The goal is to transfer the geometric structure of the many-shot classes to the classes with few examples, as shown in the middle of the figure, to eliminate this problem. The challenge is to implement this transfer using only the available training data, i.e. without manual specification of class-weights or heuristic recipes, such as equating these weights to class frequency.

We address this challenge with a combination of contributions. First, we enforce a globally learned geometric structure, which is shared by all classes. To avoid the complexity of learning a full-blown distance function, which frequently requires a large covariance matrix, we propose a structure composed by a constellation of classifier parameters, as shown on the right of Figure 1. This consists of a class-specific center, which encodes the location of the class, and a set of displacements, which are shared by all classes and encode the class geometry.

Second, we rely on a mix of randomly sampled and class-balanced mini-batches to define two losses that are used to learn the different classifier parameters. Class-balanced sampling is used to learn the class-specific center parameters. This guarantees that the learning is based on the same number of examples for all classes, avoiding a bias towards larger classes. Random sampling is used to learn the shared geometry parameters (displacements). This leverages the tendency of the standard classifier to overfit to the popular classes, making them dominant for the learning of class geometry, and thus allowing the transfer of geometric structure from these to the few-shot classes. In result, the few shot classes are learned equally to the popular classes with respect to location but inherit their geometric structure, which enables better generalization.

We propose a new learning algorithm, denoted *GeometrIc Structure Transfer* (GIST), that combines the two types of sampling, so as to naturally account for all the data in the training set, without the need for the manual specification of class weights, or even the explicit grouping of classes into different types. While we adopt the standard division into many-, medium-, and few-shot classes for evaluation, this is not necessary for training.

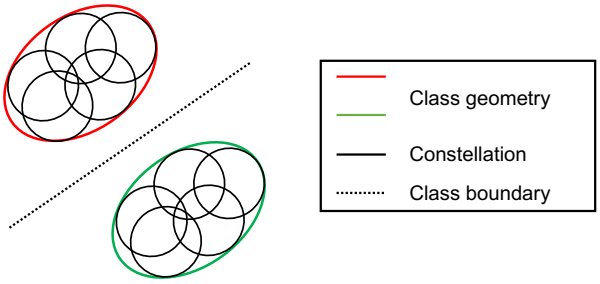


Figure 2. GistNet approximates the shared geometry by a constellation (mixture) of spherical Gaussians.

A deep network that implements the parameter constellations of Figure 1 and GIST training is then introduced and denoted as the GistNet. Experiments on two popular long-tailed recognition datasets show that it outperforms previous approaches to long-tailed recognition.

Overall, this work makes several contributions to long-tailed recognition. First, we point out that the tendency of the standard classifier to overfit to popular classes can be advantageous for transfer learning. The goal should not be to eliminate this overfitting, e.g. by uniquely adopting the now popular class-balanced sampling, but leverage it to transfer geometric information from the popular to the low-shot classes.

Second, we propose a new GistNet classifier architecture to support this goal, using constellations of classifier parameters to encode the class geometry.

Third, we introduce a new learning algorithm, GIST, that combines class-balanced and random sampling to leverage overfitting to the popular classes and enable the transfer of class geometry from popular to few-shot classes.

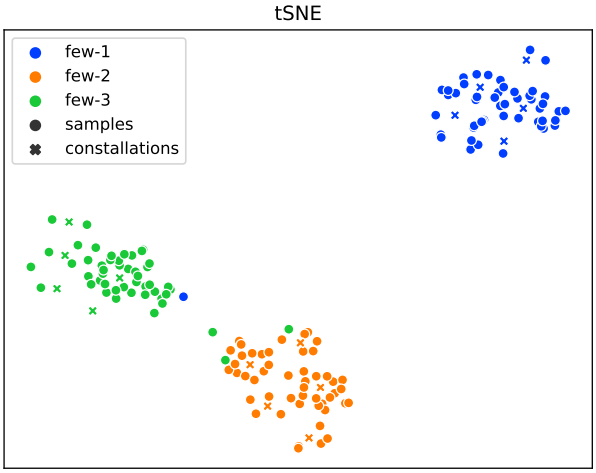


Figure 3. t-SNE visualization of 3 few-shot classes on ImageNet-LT test set, together with the constellations $\mathbf{w}_{kj}$.

2. Related Work

Long-tailed recognition has received increased attention in the recent past [25, 15, 10, 27, 13, 24]. Several approaches have been proposed, including metric learning [15, 27], hard negative mining [10], or meta-learning [24]. Some of these rely on novel loss functions, such as the lift loss [15], which introduces margins between many training samples, the range loss [27], which encourages data in the same class (different classes) to be close (far away), or the focal loss [10], which conducts online hard negative mining. These methods tend to improve performance on the few-shot end at the cost of many-shot accuracy.

Other methods, e.g. class-balanced experts [18] and knowledge distill [26], try to avoid this problem by manually dividing the training data into subsets, based on the number of examples, and training an expert per subset. However, experts learned from arbitrary data divisions can be sub-optimal, especially for few-shot classes.

Kang et al. [9] tackles the data-imbalance problem by decoupling the training feature embedding and classifier. Zhou et al. [28] also shows the effectiveness by using different training strategies on feature embedding and classifier, and achieves this by cumulative learning. These methods, however, do not discuss the class geometry problem. In face recognition, Liu et al. [12] explores the long-tailed problem by knowledge transfer. The idea is similar to ours. But they achieve this by data synthesis, while we rely on model design and training strategy.

GistNet is closest to the OLTR approach of [13], which uses a visual memory and attention to propagate information between classes. This, however, is insufficient to guarantee the transfer of geometric class structure, as intended by GIST.

Few-shot learning is a well-researched problem. A popular group of approaches is based on meta-learning, using gradient based methods such as MAML and its variants [4, 5], or LEO [17]. These methods take advantage of second derivatives to update the model from few-shot samples. Alternatively, the problem has been addressed with metric based solutions, such as the matching [22], prototypical [19], and relation [20] networks. These approaches learn metric embeddings that are transferable across classes.

There have also been proposals for feature augmentation, aimed to augment the data available for training, e.g. by combining GANs with meta-learning [23], synthesizing features across object views [11] or other forms of data hallucination [7]. All these methods are designed specifically for few-shot classes and often under-perform for many-shot classes.

Learning without forgetting aims to train a model sequentially on new tasks without forgetting the ones already learned. This problem has been recently considered in the few-shot setting [6], where the sequence of tasks includes a mix of many-shot and few-shot classes.

Proposed solutions [6, 16] attempt to deal with this by training on many-shots first, using the many-shot class weights to generate few-shot class weights, and combining them together. These techniques are difficult to generalize to long-tailed recognition, where the transition from many- to few- shot classes is continuous and includes a large number of medium-shot classes.

3. Geometric Structure Transfer

In this section, we introduce the proposed solution of the long-tailed recognition problem by geometric structure transfer and the GistNet architecture.

3.1. Regularization by Geometric Structure Transfer

A popular architecture for classification is the softmax classifier. This consists of an embedding that maps images $\mathbf{x} \in \mathcal{X}$ into feature vectors $f_\phi(\mathbf{x}) \in \mathcal{F}$, implemented by multiple neural network layers, and a softmax layer that estimates class posterior probabilities according to

$$p(y = k|\mathbf{x}; \phi, \mathbf{w}_k) = \frac{\exp[\mathbf{w}_k^T f_\phi(\mathbf{x})]}{\sum_{k'} \exp[\mathbf{w}_{k'}^T f_\phi(\mathbf{x})]} \quad (1)$$

where ϕ denotes the embedding parameters and $\mathbf{w}_k$ is the weight vector of the k^{th} class.

The model is learned with a training set $\mathbb{S} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n^s}$ of n^s examples, by minimizing the cross entropy loss

$$\mathcal{L}_{CE} = \sum_{(\mathbf{x}_i, y_i) \in \mathbb{S}} -\log p(y_i|\mathbf{x}_i). \quad (2)$$

Recognition performance is evaluated on a test set $\mathbb{T} = \{(x_i, y_i)\}_{i=1}^{n^t}$, of n^t examples.

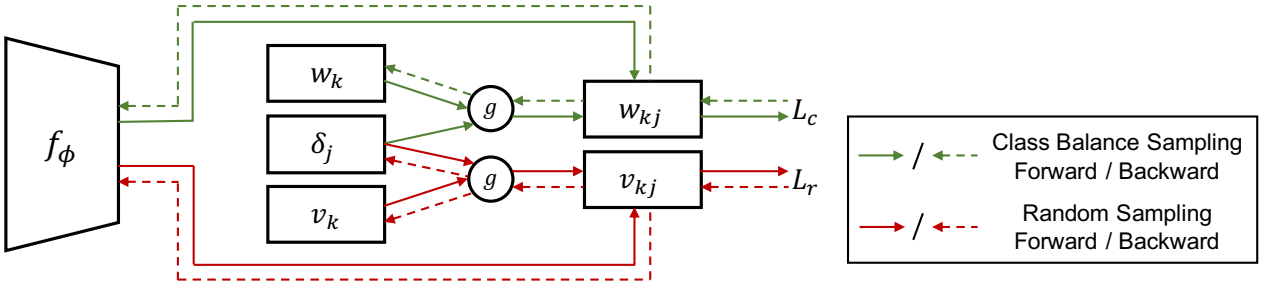


Figure 4. GIST training. Solid arrows represent feed-forward and dashed ones back-propagation. Class-balanced mini-batches are used for the green connections, to guarantee that the parameters $\mathbf{w}_k$ are class-specific. Random sampling mini-batches are used for the red connections, enabling the displacements δ_j to be learned predominantly from many-shot classes. Note that the shape parameters δ_j receive no gradient from the class-balanced loss $\mathcal{L}_c$ and the constellation centers $\mathbf{w}_k$ receive no gradient from the random sampling loss $\mathcal{L}_r$.

Learning with (2) produces a particular data-driven embedding geometry, which we denote the *natural* geometry for the training data. While parameters $\mathbf{w}_k$ of the classifier is class-specific and describes class centers, it is usually impossible to determine this geometry from the learned network parameters.[1]

This is not a problem in regular large-scale recognition. In such a case, each class has enough training data and the natural geometry is successfully learned under cross-entropy loss without further regulations. For long-tailed recognition problems the situation is different. As in few-shot learning, the limited training data of few-shot classes leads to weakly defined class-conditional distributions and embedding geometry. However, this is not the case for classes with many samples, whose natural geometry can be learned from the data. In result, as illustrated in the left of Figure 1, the true class boundaries are usually not well learned for the few-shot classes.

In this work, we seek to leverage geometric regularization to improve the learning of the few-shot classes without sacrificing performance for the populated classes.

One possibility would be to enforce a pre-defined geometry for all classes, e.g. by adopting Mahalanobis distance $d(f_\phi(\mathbf{x}), \mu) = (f_\phi(\mathbf{x}) - \mu)^T \Sigma^{-1} (f_\phi(\mathbf{x}) - \mu)$ associated with Gaussian class conditionals of covariance Σ, or by assuming Gaussian class-conditionals and regularizing the covariance to be close to a pre-defined Σ.

This has several problems. First, it is not clear what the covariance Σ should be. Second, it ignores the natural geometry of the popular classes, which is well learned by the classifier of (1). Third, given the large dimensionality of $f_\phi(\mathbf{x})$, covariance regularization is difficult to implement, even for classes with many examples.

To avoid these problems, we seek a learning-based solution that does not require covariance estimation and leverages the natural geometry of the popular classes to regularize the geometry of the few-shot classes. Rather than forcing geometry through a distance function, which is hard to learn and implement, we pursue an alternative approach to guarantee that all classes have a *shared* geometric structure.

Ideally, this structure should be learned from data, so as to 1) follow the natural geometry of the highly populated classes, and 2) allow the transfer of that geometry to the classes of few examples. It should also be encoded in a relatively small number of parameters, which at most grows linearly with the dimension of $f_\phi(\mathbf{x})$.

To achieve these goals, we continue to rely on the softmax classifier of (1) and the cross-entropy loss of (2), but use an alternative implementation of the softmax layer

$$p_\phi(y = k|\mathbf{x}) = \frac{\exp[\max_j \mathbf{w}_{kj}^T f_\phi(\mathbf{x})]}{\sum_{k'} \exp[\max_j \mathbf{w}_{k'j}^T f_\phi(\mathbf{x})]}, \qquad (3)$$
$$\mathbf{w}_{kj} = g(\mathbf{w}_k, \delta_j),$$

where the canonical parameter vector $\mathbf{w}_k$ is replaced by a *constellation* of parameter vectors $\mathbf{w}_{kj}$, which are a function of $\mathbf{w}_k$ and a set of *structure parameters* δ_j shared by all classes. Under the simplest implementation of this idea, $g(\mathbf{w}_k, \delta_j) = \mathbf{w}_k + \delta_j$ and the structure parameters are a set of displacement vectors, as shown in the right of Figure 1.

Since these displacements are shared by all classes, the constellation is simply replicated around each $\mathbf{w}_k$, which is learned per class. Because, under the loss of (2), the highly populated classes tend to dominate the optimization of the shared parameters, the displacements δ_j tend to follow the natural geometry of these classes, which is thus transferred to the few-shot classes. This regularizes the learning of these classes, enabling the recovery of the true classification boundaries, as shown in the right of Figure 1.

The displacements δ_j are the parameters that contain geometry information. They transfer the geometry from highly populated classes to few-shot classes. With the help of geometry transfer, the model learns a better geometry for few-shot classes.

As shown in Figure 2, (3) is equivalent to replacing the natural geometry by several spherical Gaussians of means w_{kj} and choosing the one closest to the feature. This approximates the non-regulated geometry by a constellation of 5 spherical Gaussians, one per $\mathbf{w}_{kj}$. This geometry is visualized in Figure 3, where features from different classes are

[1] See supplementary material for detail.

regulated by class specific constellations respectively. The constellation can be regarded as an umbrella. The model can learn the shape of the umbrella and where to place the umbrella for each class.

We denote the approach as *GeometrIc Structure Transfer* (GIST), to capture the fact that it transfers the essence, or gist, of the class geometry from popular to few-shot classes.

Note that the classifier in (3) is different from that in (1). There is an additional constraint: that the displacements δ_j are constant across classes. To avoid the model learns w_k to fit one of the constellations and ignore others. We first train the classifier from (1) to get a stable initialization of w_k, and then the whole classifier is trained to get the class-agnostic displacements. In such a case, the model will have to fit all available constellations to get lower loss instead of fitting one of them. Empirical examination in Section 4.3 shows the actual usage of $\{\delta_j\}$ is decent, and supports this assumption.

3.2. Normalization

Recent works [6, 13] have shown that better few-shot or long-tailed classification accuracies are frequently obtained by performing the classification on the unit sphere, i.e. normalizing both embedding and classifier parameters to have unit norm. We follow this practice and adopt the weighted cosine classifier [6], replacing (3) with

$$\begin{aligned} p_\phi(y=k|\mathbf{x}) &= \frac{\exp[\max_j s_\tau(f_\phi(\mathbf{x}), \mathbf{w}_{kj})]}{\sum_{k'} \exp[\max_j s_\tau(f_\phi(\mathbf{x}), \mathbf{w}_{k'j}]}, \\ s_\tau(f_\phi(\mathbf{x}), \mathbf{w}) &= \tau \frac{\mathbf{w}^T f_\phi(\mathbf{x})}{||\mathbf{w}||||f_\phi(\mathbf{x})||} \end{aligned} \tag{4}$$

where τ is a parameter that controls the smoothness of the posterior distribution. This architecture is denoted as GistNet. In our implementation, τ is randomly initialized and learned end-to-end.

3.3. GIST Training

Deep networks are trained by stochastic gradient descent (SGD). This randomly samples mini-batches of b samples, and iterates across the training set. Due to the extreme class imbalance of long-tailed recognition, SGD tends to bias the model towards the classes with more samples.

In the literature, this problem is usually addressed by class-balanced sampling [27]. This first randomly samples b_c classes with equal probability, and draws b_n samples per class, producing a mini-batch of $b = b_c \times b_n$ samples. By iterating through all classes, the model is trained with an overall equal number of examples per class. For the classifier of (1), class-balanced sampling can significantly outperform regular sampling on few-shot classes. This also makes it a good solution for learning the class specific parameters $\{\mathbf{w}_k\}$ of GistNet.

However, the bias of regular sampling towards the highly populated classes is an *advantage* for the learning of the structure parameters $\{\delta_j\}$. After all, the point is exactly to learn these parameters from classes with substantial data and transfer them to the few-shot classes, where they cannot be learned accurately. Since the parameters are shared, both goals are accomplished if the learning procedure emphasizes the highly populated classes, as is the case for regular sampling. This implies that GIST training should include a mix of regular sampling (for shared structure parameters) and class-balanced sampling (for class specific parameters).

We propose to implement this with the hybrid training scheme of Figure 4. In each iteration, two mini-batches $\mathbb{S}_c$, $\mathbb{S}_r$ are first sampled from the training set $\mathbb{S}$ by class-balanced sampling and random sampling, respectively. Two sets of class-specific parameters, $\{\mathbf{w}_k, \boldsymbol{\nu}_k\}$ are then learned, using the combination of (2), (3), and (4). The class-balanced mini-batch $\mathbb{S}_c$ is used with the resulting loss

$$\begin{aligned} \mathcal{L}_c &= \sum_{(\mathbf{x}_i, y_i)\in\mathbb{S}_c} \{-\max_j s(f_\phi(\mathbf{x}_i), \mathbf{w}_{y_i j}) \\ &\quad + \log \sum_k \exp[\max_j s(f_\phi(\mathbf{x}_i), \mathbf{w}_{kj})]\}, \\ \mathbf{w}_{kj} &= g(\mathbf{w}_k, \delta_j) \end{aligned} \tag{5}$$

to learn the parameters $\mathbf{w}_k$. The random mini-batch $\mathbb{S}_r$ is used with the loss

$$\begin{aligned} \mathcal{L}_r &= \sum_{(\mathbf{x}_i, y_i)\in\mathbb{S}_r} \{-\max_j s(f_\phi(\mathbf{x}_i), \boldsymbol{\nu}_{y_i j}) \\ &\quad + \log \sum_k \exp[\max_j s(f_\phi(\mathbf{x}_i), \boldsymbol{\nu}_{kj})]\}, \\ \boldsymbol{\nu}_{kj} &= g(\boldsymbol{\nu}_k, \delta_j) \end{aligned} \tag{6}$$

to learn the parameters $\boldsymbol{\nu}_k$. This results in the overall loss

$$\mathcal{L} = \mathcal{L}_r + \lambda \mathcal{L}_c. \tag{7}$$

The structure parameters δ_j are common to the two losses. However, as shown in Figure 4, during back-propagation only the gradient from $\mathcal{L}_r$ is used to update these parameters. This guarantees that the geometric structure is learned with random sampling. This structure is, however, propagated to the learning of the class specific parameters $\mathbf{w}_k$, which receive the gradient $\mathcal{L}_c$. In this way, the class specific parameters $\mathbf{w}_k$ are learned with class-balanced sampling, but this learning is informed by the structure parameters δ_j learned with random sampling. This leads to parameter constellations $\mathbf{w}_{kj}$ that, while shared across classes, are centered at class-specific locations.

Note that the displacements are forwarded together with $\mathbf{w}_k$ to calculate the class-balanced loss $\mathcal{L}_c$. This makes the two components $\{\mathbf{w}_j\}$ and $\{\delta_j\}$ of the classifier matching each other, although they are learned by different losses.

Table 1. Results on ImageNet-LT and Places-LT. ResNet-10/152 are used for all methods. For many-shot $t > 100$, for medium-shot $t \in (20, 100]$, and for few-shot $t \leq 20$, where t is the number of training samples.

Method	ImageNet-LT				Places-LT			
	Overall	Many-Shot	Medium-Shot	Few-Shot	Overall	Many-Shot	Medium-Shot	Few-Shot
Plain Model	23.5	41.1	14.9	3.6	27.2	**45.9**	22.4	0.36
Lifted Loss [15]	30.8	35.8	30.4	17.9	35.2	41.1	35.4	24.0
Focal Loss [10]	30.5	36.4	29.9	16.0	34.6	41.1	34.8	22.4
Range Loss [27]	30.7	35.8	30.3	17.6	35.1	41.1	35.4	23.2
FSLwF [6]	28.4	40.9	22.1	15.0	34.9	43.9	29.9	29.5
OLTR [13]	35.6	43.2	35.1	18.5	35.9	44.7	37.0	25.3
Decoupling [9]	41.4	51.8	38.8	21.5	37.9	37.8	40.7	31.8
Distill [26]	38.8	47.0	37.9	19.2	36.2	39.3	39.6	24.2
GistNet	**42.2**	**52.8**	**39.8**	**21.7**	**39.6**	42.5	**40.8**	**32.1**

Table 2. Results on the iNaturalist 2018. All methods are implemented with ResNet-50.

Method	Accuracy
CB-Focal [2]	61.1
LDAM+DRW [1]	68.0
Decoupling [9]	69.5
GistNet	**70.8**

The parameters $\boldsymbol{\nu}_k$ are only used at training time, to guarantee that the geometric parameters δ_j follow the natural geometry of the highly populated classes. They are discarded after training.

In GIST training, the class-specific weights $\mathbf{w}_k$ are trained with class-balanced sampling, while the structure parameters δ_j are trained with random sampling. This forces the latter to predominantly represent the structure of the popular classes and is what enables the geometric structure transfer of Figure 1.

4. Experiments

In this section, we discuss an evaluation of the long-tailed recognition performance of the GistNet.

4.1. Experimental set-up

Datasets. We consider three long-tailed recognition datasets, ImageNet-LT [13], Places-LT [13] and iNatrualist18 [21]. ImageNet-LT is a long-tailed version of ImageNet [3] by sampling a subset following the Pareto distribution with power value $\alpha = 6$. It contains 115.8K images from 1000 categories, with class cardinality ranging from 5 to 1280. Places-LT is a long-tailed version of the Places dataset [29]. It contains 184.5K images from 365 categories with class cardinality ranging from 5 to 4980. iNatrualist18 is a long-tailed dataset, which contains 437.5K images from 8141 categories with class cardinality ranging from 2 to 1000.

Baselines. Following [13], we consider three metric-learning baselines, based on the lifted [15], focal [10], and range [27] losses, and one state-of-the-art method, FSLwF [6], for learning without forgetting. We also include state-of-the-art long-tailed recognition methods designed specifically for these two datasets: OLTR [13], Decoupling [9], and Distill [26]. The classifier of (1) with standard random sampling is denoted as the *Plain Model* for comparison.

Training Details. ResNet-10 and ResNet-152 [8] are used on ImageNet-LT and Places-LT respectively, and ResNet-50 is used on iNatrualist18. Unless otherwise noted, we use four vectors δ_j of structure parameters, each with the dimension of $f_\theta(\mathbf{x})$. The class center $\mathbf{w}_k$ completes a constellation of five vectors. The number of structure parameters is ablated in Section 4.3. In all experiments, $\lambda = 0.5$ is used in (7).

The model is first pre-trained without structure parameters, with 60 epochs of SGD, using momentum 0.9, weight decay 0.0005, and a learning rate of 0.1 that decays by 10% every 15 epochs. After this, the full model is subject to GIST training with momentum 0.9, weight decay 0.0005 for 60 epochs, and learning rate 0.1 that decays by 10% every 15 epochs. In this case, each iteration uses class-balanced and random sampling mini-batches of size 128, for an overall batch size of 256. One epoch is defined when the random sampling iterates over the entire training data. Codes are attached in supplementary.

4.2. Results

Table 1 present results on ImageNet-LT and Places-LT. GistNet outperforms all other methods on the two datasets. A further comparison is performed by splitting classes into *many shot* (more than 100 training samples), *medium shot* (between 20 and 100 training samples), and *few shot* (less that 20 training samples). GistNet achieves the best performance on 5 of the 6 partitions and is competitive on the remaining one.

While on Places-LT the largest gains are for the few-shot classes, in ImageNet-LT they hold for the medium- and many-shot classes. This suggests that, in this dataset, the remaining methods overfit to the few-shot classes. The higher robustness of GistNet to this overfitting can be explained

Table 3. Ablation of GistNet components, on the ImageNet-LT *validation set*. For many-shot $t > 100$, for medium-shot $t \in (20, 100]$, and for few-shot $t \leq 20$, where t is the number of training samples.

Method	Overall	Many-Shot	Medium-Shot	Few-Shot
Plain Model	25.1	42.9	16.6	0.43
COS+CB	37.6	49.4	34.8	14.7
COS+CS+CB	39.5	52.6	36.3	14.5
COS+CS+GIST (GistNet)	**43.5**	**54.8**	**41.0**	**21.4**
COS+GIST	40.2	51.4	37.4	19.0
COS+CS+GIST ($\mathbf{w}_k$ and $\boldsymbol{\nu}_k$ combined)	40.9	**58.2**	34.6	14.8
COS+CS+GIST (g rotation)	**43.6**	55.1	40.8	**21.7**
COS+CS+GIST (g MLP)	43.4	54.2	**41.1**	21.5

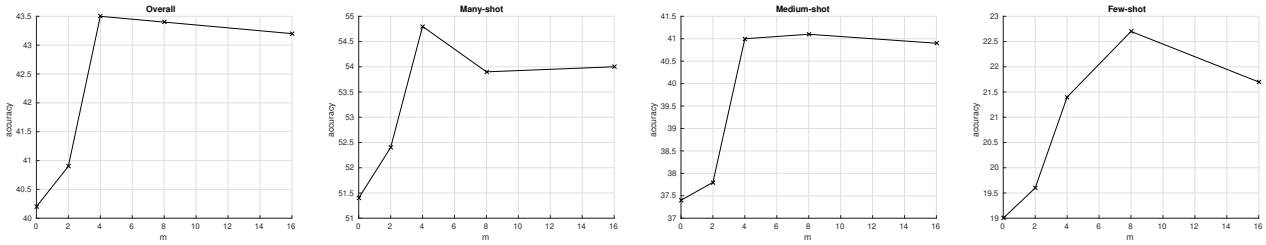


Figure 5. Results on different size of structure parameters in few-shot, medium-shot, many-shot classes, and overall accuracy, searched on validation set.

by the predominance of the many-shot classes in the training of the structure parameters δ_j. Results on iNaturalist18 dataset are shown in Table 2, ours also outperforms all other methods.

4.3. Ablation Study

In this section, we discuss the effectiveness of the various components of GistNet, the choice of constellation function g, the number of structure parameters, and the actual usage of constellations. All models are trained and evaluated on the training and validation set of ImageNet-LT, respectively, using a ResNet-10 backbone.

Component ablation. Starting from the plain model of (1), we incrementally add the cosine classifier (COS) used in (4), class-balanced sampling (CB), class structure parameters (CS), and GIST training (GIST). Table 3 shows that the combination of cosine classifier and class-balanced sampling (COS+CB) improves significantly on the plain model. The simple addition of the class structure parameters (COS+CS+CB) further improves the overall performance.

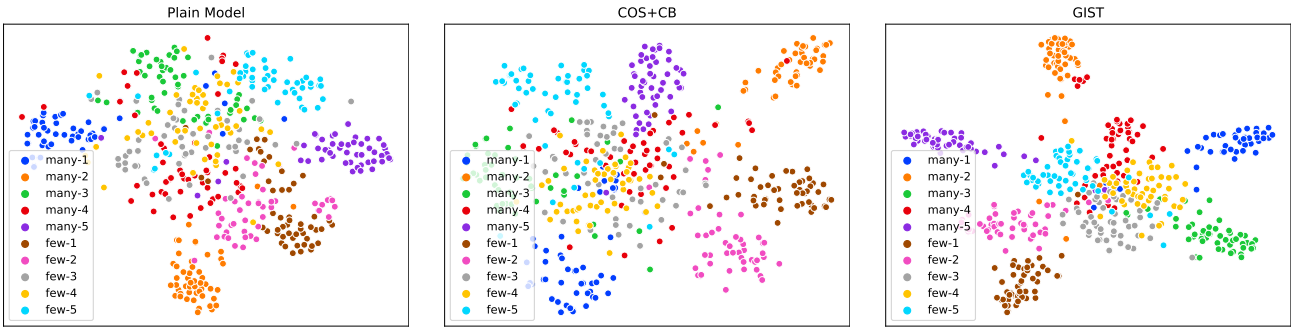
However, there is no significant improvement for few-shot classes. This can be explained by the fact that, with class-balanced sampling, the three class types are equally predominant for the learning of the structure parameters. Hence, there is no transfer of geometric structure from many- to few-shot classes. This is confirmed by the fact that, when GIST training is added (COS+CS+GIST), performance improves significantly for the few-shot classes. When compared to COS+CB, the GistNet model (COS+CS+GIST) has an overall gain of about 6 points and better performance for all class types. Among these, the gains are particularly large (around 6.5 points) for the few-shot classes.

The middle of the table investigates other possible configurations of the GistNet. Applying GIST training without class structure parameters (COS+GIST), i.e. using the combination of class balanced and random sampling only to learn the embedding $f_\phi(\mathbf{x})$, degrades performance for all class partitions. This shows the importance of enforcing a shared class structure among all classes.

Another variant is to remove the additional class centers $\{\boldsymbol{\nu}_k\}$ of Figure 4, using the centers $\{\mathbf{w}_k\}$ for both losses, i.e. replacing $\boldsymbol{\nu}_k$ with $\mathbf{w}_k$ in (6). This variant, denoted COS+CS+GIST ($\mathbf{w}_k$ and $\boldsymbol{\nu}_k$ combined), eliminates all the gains of GistNet for few-shot classes, while increasing the recognition accuracy for those in the many-shot partition. This is because the centers now receive gradient from the random sampling loss and are predominantly trained with many-shot data. The improved performance of GistNet over this variant shows that it is important to maintain the class-specificity of center training, while enforcing transfer of the geometric structure parameters, as done by GIST.

Different choices of g. Beyond these variants, we have also considered different choices for the function g that defines the parameter constellations of (3). In addition to the default addition function implemented by GistNet, we considered two possibilities.

The first was a rotation. After the embedding and classifier parameters are normalized, we evaluate the distance



Plain Model COS+CB GistNet

Figure 6. t-SNE visualizations of the embedding of *test set* images from 5 randomly chosen many- and few-shot ImageNet-LT classes, for three models.

between them on the d-dimensional unit sphere (where d is the dimension of $f_\phi(\mathbf{x})$). The structure parameters are then d-dimensional rotation matrices, which encourage all classes to have the same structure on the unit sphere. This is implement the rotation matrix by a transformation of d-dimensional displacement vector

$$\mathbf{R} = \mathbf{I} - \mathbf{u}\mathbf{u}^T - \mathbf{v}\mathbf{v}^T + [\mathbf{u}, \mathbf{v}]\mathbf{R}_\theta[\mathbf{u}, \mathbf{v}]^T, \quad (8)$$

where $\mathbf{u}$ is a unit vector, $\mathbf{v}$ is the normalized vector of a displacement vector δ_j, and $\mathbf{R}_\theta$ is the 2D rotation matrix between $\mathbf{u}$ and δ. Given a structure parameter vector δ_j, the parameter constellations are implemented as

$$\mathbf{w}_{kj} = g(\mathbf{w}_k, \delta_j) = \mathbf{R}\mathbf{w}_k \quad (9)$$

Details are discussed in supplementary.

The second was a learned function g, implemented by a two-layer MLP, and learned end-to-end.

Table 3 shows that the different implementations of g have little impact on the recognition performance. This suggests that the addition of global geometric constraints is much more important than the specific implementation details of these constraints.

Number of structure parameters. We next investigated the influence of the number m of structure parameters $\{\delta_j\}_{j=1}^m$. As shown in Figure 5, none of the alternatives tried ($m \in \{2, 8, 16\}$) outperformed the four parameters used in GistNet. For overall, many-, and medium-shot classes performance increases until $m = 4$ and then saturates. For few-shot classes, there was a one-point gain in using $m = 8$. This shows that this partition is the one that most benefits from geometry transfer.

Overall, these results confirm that while geometric transfer can produce significant gains, the GistNet architecture is quite robust to variations of detail.

Actual usage of constellations. Cross-entropy minimization encourages the use of more δ_j, since the coverage of the class distributions is better. It would be a waste not to use them all. In the test set of ImageNet-LT, the actual usage was $\{25\%, 23\%, 18\%, 17\%, 17\%\}$. 792 of 1000 classes chose each δ_j for at least 10% of test samples. This results further support that the model does not collapse to traditional classifier by fitting to only one constellation and ignoring others.

4.4. Visualization

Figure 6 shows a t-SNE [14] visualization of the embeddings learned by the Plain Model, the COS+CB baseline, and GistNet. For clarity, we randomly choose five classes from the many- and few-shot splits in ImageNet-LT. The figure shows the t-SNE projection of features of *test* samples from those classes. Compared to the two other models, GistNet produces classes that are better separated and have more consistent geometry. This is especially true for few-shot classes.

5. Conclusion

This work addressed the long-tailed recognition problem. A new architecture, GistNet, and training scheme, GIST, were proposed to enable transfer of geometric structure from highly populated to low-populated classes. This leverages the tendency of SGD training to overfit to the populated classes, rather than simply fighting this tendency.

GistNet was shown to achieve state-of-the-art performance on two popular long-tailed datasets. Ablation studies have shown that, while geometric transfer enables significant recognition gains, the architecture is quite robust to variations of detail. This suggests that the addition of global geometric constraints to long-tailed recognition is more important than the specific implementation of these constraints.

Acknowledgement. Bo Liu and Nuno Vasconcelos were partially supported by NSF awards IIS-1637941, IIS-1924937, and NVIDIA GPU donations. Gang Hua was supported partly by National Key R&D Program of China Grant 2018AAA0101400 and NSFC Grant 61629301.

References

[1] 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 1565–1576, 2019. 6

[2] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9268–9277, 2019. 6

[3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009. 1, 6

[4] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1126–1135. JMLR. org, 2017. 3

[5] Chelsea Finn, Kelvin Xu, and Sergey Levine. Probabilistic model-agnostic meta-learning. In *Advances in Neural Information Processing Systems*, pages 9516–9527, 2018. 3

[6] 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. 3, 5, 6

[7] Bharath Hariharan and Ross Girshick. Low-shot visual recognition by shrinking and hallucinating features. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3018–3027, 2017. 3

[8] 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. 6

[9] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. In *Eighth International Conference on Learning Representations (ICLR)*, 2020. 3, 6

[10] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 1, 3, 6

[11] Bo Liu, Xudong Wang, Mandar Dixit, Roland Kwitt, and Nuno Vasconcelos. Feature space transfer for data augmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 3

[12] 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. 3

[13] 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. 1, 3, 5, 6

[14] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008. 8

[15] 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. 3, 6

[16] Mengye Ren, Renjie Liao, Ethan Fetaya, and Richard Zemel. Incremental few-shot learning with attention attractor networks. In *Advances in Neural Information Processing Systems*, pages 5276–5286, 2019. 3

[17] Andrei A. Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. 3

[18] Saurabh Sharma, Ning Yu, Mario Fritz, and Bernt Schiele. Long-tailed recognition using class-balanced experts. *arXiv preprint arXiv:2004.03706*, 2020. 3

[19] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in neural information processing systems*, pages 4077–4087, 2017. 3

[20] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1199–1208, 2018. 3

[21] 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. 6

[22] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *Advances in neural information processing systems*, pages 3630–3638, 2016. 3

[23] Yu-Xiong Wang, Ross Girshick, Martial Hebert, and Bharath Hariharan. Low-shot learning from imaginary data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7278–7286, 2018. 3

[24] Yu-Xiong Wang and Martial Hebert. Learning to learn: Model regression networks for easy small sample learning. In *European Conference on Computer Vision*, pages 616–634. Springer, 2016. 3

[25] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Learning to model the tail. In *Advances in Neural Information Processing Systems*, pages 7029–7039, 2017. 3

[26] 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. 3, 6

[27] 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. 1, 3, 5, 6

[28] 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. 3

[29] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. In *Advances in neural information processing systems*, pages 487–495, 2014. 6