Locally-Enriched Cross-Reconstruction for Few-Shot Fine-Grained Image Classification

Xiaoxu Li, Qi Song, Jijie Wu, Rui Zhu, Zhanyu Ma, *Senior Member, IEEE*, Jing-Hao Xue, *Senior Member, IEEE*

***Abstract*—Few-shot fine-grained image classification has attracted considerable attention in recent years for its realistic setting to imitate how humans conduct recognition tasks. Metric-based few-shot classifiers have achieved high accuracies. However, their metric function usually requires two arguments of vectors, while transforming or reshaping three-dimensional feature maps to vectors can result in loss of spatial information. Image reconstruction is thus involved to retain more appearance details: the test images are reconstructed by different classes and then classified to the one with the smallest reconstruction error. However, discriminative local information, vital to distinguish sub-categories in fine-grained images with high similarities, is not well elaborated when only the base features from a usual embedding module are adopted for reconstruction. Hence, we propose the novel local content-enriched cross-reconstruction network (LCCRN) for few-shot fine-grained classification. In LCCRN, we design two new modules: the local content-enriched module (LCEM) to learn the discriminative local features, and the cross-reconstruction module (CRM) to fully engage the local features with the appearance details obtained from a separate embedding module. The classification score is calculated based on the weighted sum of reconstruction errors of the cross-reconstruction tasks, with weights learnt from the training process. Extensive experiments on four fine-grained datasets showcase the superior classification performance of LCCRN compared with the state-of-the-art few-shot classification methods. Codes are available at: https://github.com/lutsong/LCCRN.**

***Index Terms*—Few-shot image classification, fine-grained image classification, discriminative local features, ridge regression, image reconstruction**

I. Introduction

With the development of deep learning, the recognition performance of machines has surpassed that of humans in many large-scale image classification tasks. However, when the amount of data that can be learnt from is small, the recognition ability of machine is not satisfactory [1], [2]. Therefore, image classification based on a very small number of labelled samples, often referred to as few-shot classification, has attracted considerable research attention in recent years. Few-shot classification usually involves two types of data with disjoint label spaces, namely the base class data and the novel class data. It aims to use the knowledge learnt from the base class data and a small number of labelled samples from the novel class data to accurately predict the labels of the unseen samples from the novel classes.

In this paper, we aim to investigate the challenging task of few-shot fine-grained image classification [3]–[5]. Fine-grained data have a lot of sub-categories, each with few labelled samples, which are natural choices to evaluate the classification performances of the few-shot algorithms. However, the high similarities between the sub-categories make the classification of fine-grained images difficult.

Metric-based approaches are effective solutions for few-shot classification. They determine the class membership of a test sample via the similarity or dissimilarity between the test sample and the classes, which is evaluated by a metric function, such as the simple pre-defined cosine similarity [6] and Euclidean distance [7], and more advanced nonlinear functions and networks that can be learnt from data [8]–[13]. However, conventional metric-based methods usually do not perform well for fine-grained images, because they do not take the high similarities between the sub-categories into consideration. To precisely classify fine-grained images, extracting the discriminative features that can distinguish between the sub-categories is one key to success. Li et al. propose the bi-similarity network (BSNet) that utilises two similarity measures to learn distinctive characteristics of each class [5]. Huang et al. propose the low-rank pairwise alignment bilinear network (LRPABN) involving the bilinear pooling operation to capture the subtle differences between the support and query images [4]. Huang et al. later propose the target-oriented aligment network (TOAN) to explicitly reduce the intra-class variance by matching the support and query features while enlarge the inter-class variance by extracting discriminative fine-grained features [14].

The feature map reconstruction networks (FRN) designed by Wertheimer et al. [15] can also provide excellent classification performance for fine-grained images. FRN aims to tackle one problem in metric-based approaches: the metric functions usually measure the similarity or dissimilarity between two vectors, while transforming or reshaping the three-dimensional feature maps extracted by convolutional modules to vectors can result in loss of spatial information. In FRN, the feature maps of each class in the support set are pooled to a single two-dimensional matrix of support features, with each column representing the concatenated feature maps of a channel. To classify a query image, every location of its feature map is reconstructed by the weighted sum of the support features of each class via a ridge regression formulation, and the reconstruction error is adopted to calculate the metric score. Images

Corresponding author: Rui Zhu (email: rui.zhu@city.ac.uk)

X. Li, Q. Song, J. Wu are with the School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China.

R. Zhu is with the Faculty of Actuarial Science and Insurance, Bayes Business School, City, University of London EC1Y 8TZ, UK.

Z. Ma are with the Pattern Recognition and Intelligent System Laboratory, School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing 100876, China.

J.-H. Xue is with the Department of Statistical Science, University College London, London WC1E 6BT, U.K.

should be well reconstructed by the feature maps of their corresponding classes because their embeddings are similar, while they are hard to be reconstructed by wrong classes that result in large reconstruction errors. Hence, through the reconstruction process, it is possible to preserve the appearance details to assist to distinguish the sub-categories.

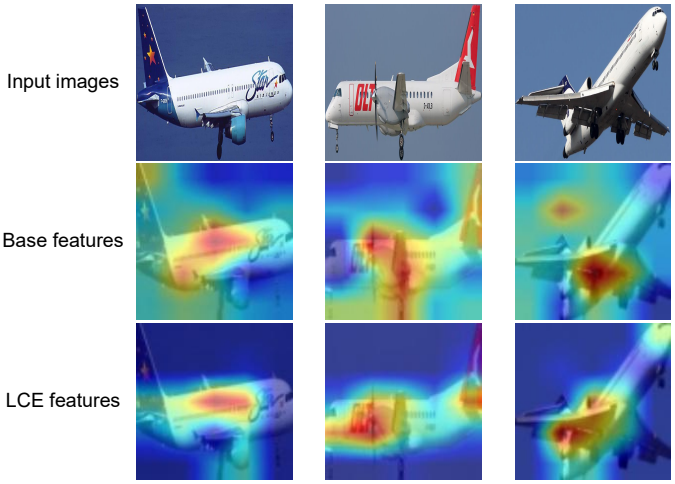


Fig. 1. Illustrations of the base and local content-enriched features of the aircraft images. The base features capture large areas of the aircrafts as well as the annoying background, while the local content-enriched features can focus more on the discriminative local areas of the aircrafts.

In FRN, however, when simply adopting the base features from the usual embedding module for reconstruction, the discriminative local features that are vital for fine-grained image classification are not well considered. For example, in Fig. 1, the base features capture most of the aircrafts, but also involve large areas of the annoying background of sky around the aircrafts. Thus, to further improve the classification performance under the feature reconstruction framework of FRN, we propose a local content-enriched cross-reconstruction network (LCCRN), with a local content extraction module that learns the local information to assist fine-grained few-shot image classification. By utilising the new local content extraction module, the local content-enriched features, as shown in the last row of Fig. 1, can concentrate more on the local areas of the aircrafts and get rid of most of the background.

Besides the local content-enriched features, we generate an additional set of base features through a separate embedding module to preserve the appearance details. To fully exploit the best of both types of features, we propose the new cross-reconstruction module, through which the query features that focus on both the appearance details and local information are reconstructed via ridge regression. The name 'cross-reconstruction' means that the reconstruction tasks are conducted across different feature types, e.g. reconstructing the query image's base features from the corresponding local content-enriched features, and vice-versa. In this way, the relationship between the base features and their neighbourhood is explored, allowing the model to learn more about the local information presented in images. The classification score is derived based on the weighted sum of reconstruction errors of the cross-reconstruction tasks, with the weights adaptively learnt during the training process. The experiments on four fine-grained image datasets demonstrate the superior classification performance of LCCRN over state-of-the-art methods.

To sum up, our contributions are three-fold:

- We propose a novel local content-enriched cross-reconstruction network (LCCRN) for few-shot fine-grained image classification.
- We propose a new local content extraction module (LCEM) to explore the discriminative local information within the feature maps, which can assist semantic understanding of the images.
- We design a novel cross-reconstruction module (CRM) to leverage both the base and local content-enriched feature representations, enabling the learning of both appearance details and local information.

The rest of the paper is organised as follows. In section II, we provide a thorough discussion about the literature closely related to our work. We then present technical details of the proposed LCCRN in section III. Extensive experimental results and ablation study are presented in section IV. Lastly, we draw conclusions in section V and propose future work.

II. Related work

A. Metric-based methods for few-shot image classification

Metric-based methods for few-shot image classification aim to learn a metric function to distinguish between image classes based on the similarity or dissimilarity evaluated in a metric space. For example, the matching networks (MatchNet) [6] utilise the attention mechanism for feature embedding and adopt the cosine similarity. The prototypical networks (ProtoNet) [7] represent each class by a prototype and classify a test sample via its Euclidean distances to the class prototypes. The relation network (RelationNet) [8] improves the prototypical networks by learning a metric module to assess the relation scores between the test sample and the class prototypes. The similarities between images can also be measured by graph neural networks (GNN) [16], [17], which consider the similarities between any two images in the task. Some metric-based approaches can provide promising classification results for fine-grained images, because they aim to extract more discriminative features. For instance, the deep nearest neighbor neural network (DN4) [18] adopts a local descriptor-based image-to-class measure to preserve the local discriminative information. Adopting only one metric function may not well capture the subtle differences between the sub-categories, thus the bi-similarity network (BSNet) [5] involves two distinct similarity measures to learn more discriminative feature maps. The low-rank pairwise alignment bilinear network (LRPABN) [4] can exploit the subtle differences between the support and query images via the bilinear pooling operation.

Different from previous methods, LCCRN involves four reconstruction tasks to exploit the discriminative local information. Thus LCCRN calculates the weighted sum of the reconstruction errors of the reconstruction tasks as the metric for classification.

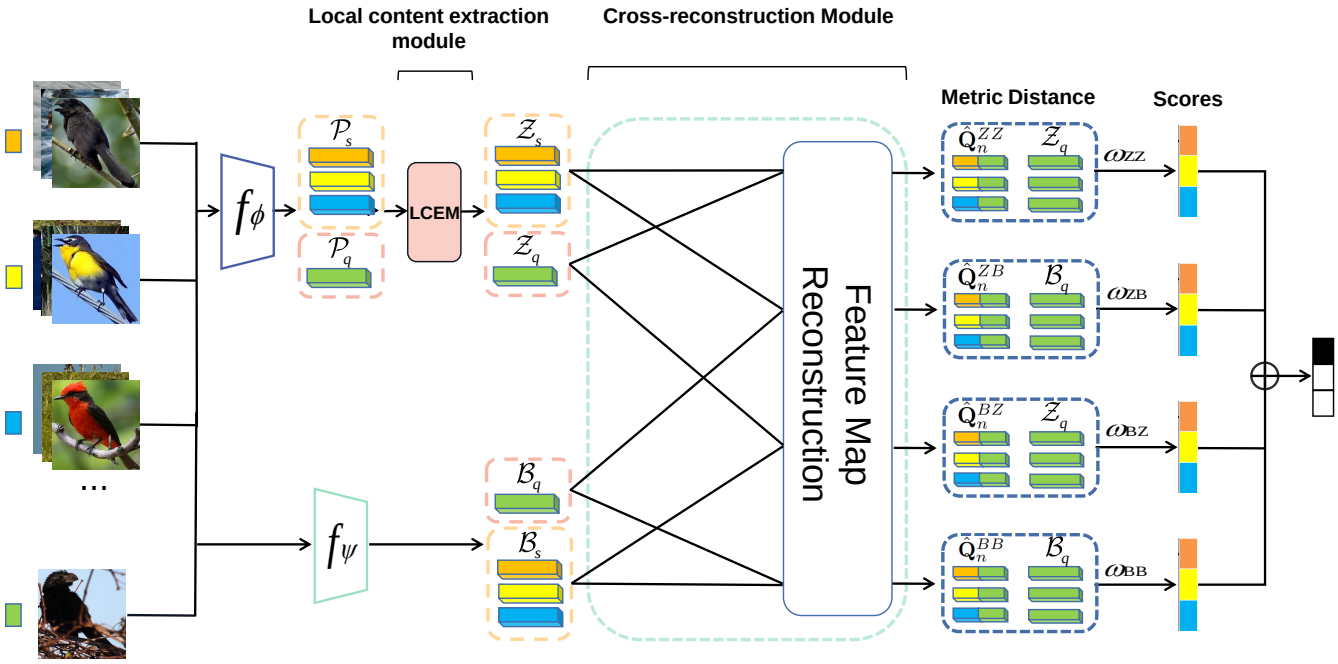


Fig. 2. The framework of LCCRN. We use orange, yellow and blue to represent three sub-categories of support images, and green to represent query images. The original images are fed to the embedding module f_ϕ to obtain the base representations $\boldsymbol{\mathcal{P}}_s$ and $\boldsymbol{\mathcal{P}}_q$, which are later converted into the local content enriched feature representations $\boldsymbol{\mathcal{Z}}_s$ and $\boldsymbol{\mathcal{Z}}_q$ by the local content extraction module (LCEM). A separate set of base representations $\boldsymbol{\mathcal{B}}_s$ and $\boldsymbol{\mathcal{B}}_q$ are obtained by the embedding module f_ψ that does not share parameters with f_ϕ. Then, the cross-reconstruction module cross-reconstructs the two types of query features by the two types of support features, resulting in four reconstruction tasks. Finally, the metric score is calculated based on the weighted reconstruction errors. $\hat{\mathbf{Q}}_n^{ZZ}$, $\hat{\mathbf{Q}}_n^{ZB}$, $\hat{\mathbf{Q}}_n^{BZ}$ and $\hat{\mathbf{Q}}_n^{BB}$ denote the four reconstructed query images based the nth class of the support set. ω_{ZZ}, ω_{ZB}, ω_{BZ} and ω_{BB} denote the corresponding weights to calculate the weighted reconstruction score for the query image.

B. Feature alignment for few-shot image classification

Feature alignment approaches usually aim to align the spatial positions of similar objects to improve the learnt similarities between images. The position-aware relational network (PARN) [19] calculates the similarities between any two positions of features, regardless of their spatial distances, and thus can provide high similarities for images with related objects that are not in the same positions. The semantic alignment metric learning (SAML) [20] adopts the collect-and-select strategy to give higher weights to the semantically-related local regions. DeepEMD [9] uses the Earth Mover's Distance (EMD) to calculate the structural similarity between two images. CrossTransformers (CTX) [21] determine the similarity between two images via the spatially-correlated features via a transformer-based network. The global-local interplay metric learning (GLIML) [22] semantically aligns the global information of image label semantics and the local features, which can decrease the semantics irrelevant to the global information. FRN [15] utilises the ridge regression to reconstruct the feature map of the query image based on the support features, which has a closed-form solution and is computationally efficient. FRN tries to preserve the spatial details, but cannot well reconstruct the local features, and thus the semantic information of the images is not fully considered.

In contrast, in the LCCRN proposed in this paper, we design a novel cross-reconstruction module that can fully integrate both the base and the local content-enriched feature representations to enhance the semantic understanding of the network.

III. Methodology

In this section, we introduce our new local content-enriched cross-reconstruction network (LCCRN), with the technical details of the local content extraction module in section III-C and the cross-reconstruction module in section III-D.

A. Problem definition

In few-shot classification, given a dataset $\mathcal{D}$, we partition it into the base dataset $\mathcal{D}_b$ with its class set $\mathcal{L}_b$, the validation dataset $\mathcal{D}_v$ with its class set $\mathcal{L}_v$ and the novel dataset $\mathcal{D}_n$ with its class set $\mathcal{L}_n$. Note that the three class sets, $\mathcal{L}_b$, $\mathcal{L}_v$ and $\mathcal{L}_n$, are mutually exclusive. The goal of few-shot classification is to train a model based on $\mathcal{D}_b$ and $\mathcal{L}_b$ so that the model can generalize well on tasks sampled from $\mathcal{D}_n$, which is challenging because there are only few labelled samples available for each task. $\mathcal{D}_v$ and $\mathcal{L}_v$ are used to verify the accuracy of the current model during the iterative training process to determine whether it is the best-performing one. We follow the classic N-way K-shot classification setting in this paper. That is, in each task, the few labelled data form the *support set* $\mathcal{S} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N\times K}$, which contains N classes with K support images for each class. The model is then evaluated on a separate *query set* $\mathcal{Q} = \{(\mathbf{x}_j, y_j)\}_{j=1}^{N\times q}$, with the same N classes in $\mathcal{S}$ and q query images for each class. The test performance of the model is calculated as the averaged

accuracy on the query set over multiple tasks sampled from $\mathcal{D}_n$.

B. The framework of LCCRN

In Fig. 2, we depict the framework of LCCRN. The support set $\mathcal{S}$ and the query set $\mathcal{Q}$ are both fed to two distinct embedding modules, f_ϕ and f_ψ, which do not share parameters. Two sets of base features are thus obtained: $\boldsymbol{\mathcal{P}}_s$ and $\boldsymbol{\mathcal{P}}_q$ from f_ϕ; and $\boldsymbol{\mathcal{B}}_s$ and $\boldsymbol{\mathcal{B}}_q$ from f_ψ. We adopt the subscripts s and q to denote features from the support and query sets, respectively. The local content extraction module (LCEM) takes $\boldsymbol{\mathcal{P}}_s$ and $\boldsymbol{\mathcal{P}}_q$ as inputs to generate the local content-enriched (LCE) features $\boldsymbol{\mathcal{Z}}_s$ and $\boldsymbol{\mathcal{Z}}_q$, which then go through the cross-reconstruction module together with $\boldsymbol{\mathcal{B}}_s$ and $\boldsymbol{\mathcal{B}}_q$ to obtain four sets of cross-reconstructed query features. The metric distances between these cross-reconstructed query features and their corresponding ground-truth query features are calculated as the reconstruction errors. The weighted sum of these four reconstruction errors are adopted as the metric score for classification of the query image.

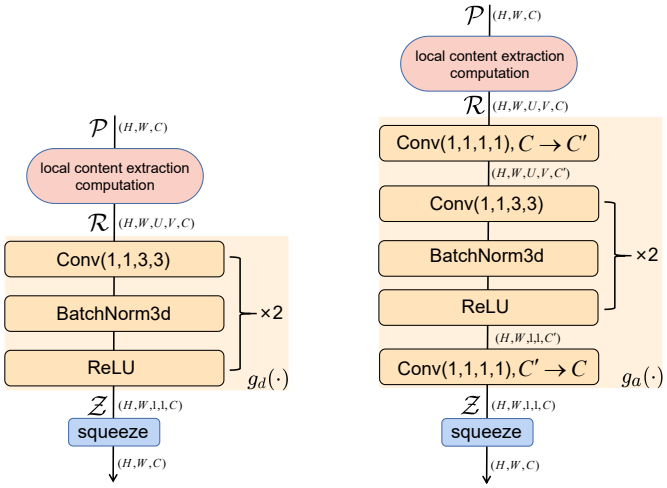


Fig. 3. The local content extraction module. Left: the local information extraction module under the Conv-4 backbone. Right: the local information extraction module under the Resnet-12 structure. The relational patterns in the input local windows $\boldsymbol{\mathcal{R}}$ are captured by convolving the $U \times V$ dimensions.

C. The local content extraction module

The local content extraction module (LCEM) aims to convert the base representations $\boldsymbol{\mathcal{P}}_s$ and $\boldsymbol{\mathcal{P}}_q$ into the ones that can reflect more about the discriminative local information of the support and query images to assist classification.

Suppose the base representation of the i th image in $\mathcal{S}$ is denoted as $\boldsymbol{\mathcal{P}}_{s(i)} \in \mathbb{R}^{H\times W\times C}$, where H and W denote the height and weight of the feature maps, respectively, while C denotes the number of channels. The LCEM is implemented via the following two steps. First, we pool the neighbourhood windows surrounding each position in each channel of $\boldsymbol{\mathcal{P}}_{s(i)}$ to generate the representation $\boldsymbol{\mathcal{R}}_{s(i)} \in \mathbb{R}^{H\times W\times U\times V\times C}$ for further exploration of local structures. To be more specific, for each value of $\boldsymbol{\mathcal{P}}_{s(i)}(h,w,c)$ $(h = 1,2,\ldots,H, \quad w = 1,2,\ldots,W, \quad c = 1,2,\ldots,C)$, we take its neighbourhood window of size $U \times V$ with $\boldsymbol{\mathcal{P}}_{s(i)}(h,w,c)$ in the centre and normalise the window by its Frobenius norm, and aggregate all local windows to form $\boldsymbol{\mathcal{R}}_{s(i)}$. Zero padding is adopted to deal with the positions on the edges of the feature maps.

Then, to learn the local content-enriched (LCE) features, we propose a novel convolutional module to extract the local features from $\boldsymbol{\mathcal{R}}_{s(i)}$ by convolving the $U \times V$ dimensions. Through this module, we also aim to obtain the feature representations that match the dimensions of $\boldsymbol{\mathcal{B}}_{q(i)} \in \mathbb{R}^{H\times W\times C}$ to facilitate the cross-reconstruction step. Fig. 3 (left) presents our designed module for the Conv-4 backbone, which includes two 3×3 convolutional layers for conversion with batch normalisation and ReLU inserted between convolutions. Compared with that for the Conv-4 backbone, the feature map obtained by the ResNet-12 backbone has more channels. In order to improve the computational efficiency, the local content extraction module for the ResNet-12 backbone depicted in Fig. 3 (right) has two additional point convolutional layers for channel size reduction. Note that the strategy of using the two additional point convolutional layers to reduce the channel size for efficient computation can be generalised directly to other networks with a large amount of channels. We denote this convolution block as $g(\cdot) : \mathbb{R}^{H\times W\times U\times V\times C} \to \mathbb{R}^{H\times W\times C}$. Thus, the final LCE features of the ith image in $\mathcal{S}$ can be calculated in the following general formulation:

$$\boldsymbol{\mathcal{Z}}_{s(i)} = g(\boldsymbol{\mathcal{R}}_{s(i)}) = \begin{cases} g_d(\boldsymbol{\mathcal{R}}_{s(i)}) & C \le 64, \\ g_a(\boldsymbol{\mathcal{R}}_{s(i)}) & C > 64, \end{cases} \tag{1}$$

where $\boldsymbol{\mathcal{Z}}_{s(i)} \in \mathbb{R}^{H\times W\times C}$, $g_d(\cdot)$ is the module on the left-hand-side of Fig. 3 when the two point convolutional layers are deactivated while $g_a(\cdot)$ is that on the right-hand-side when the two point convolutional layers are activated for computational efficiency. In $g_a(\cdot)$, we reduce the channel size to 64, i.e. $C' = 64$.

For the jth query image $\boldsymbol{\mathcal{P}}_{q(j)}$, we follow the same approach to obtain $\boldsymbol{\mathcal{R}}_{q(j)}$ and calculate its LCE features as $\boldsymbol{\mathcal{Z}}_{q(j)} = g(\boldsymbol{\mathcal{R}}_{q(j)})$.

D. The feature cross-reconstruction module

To fully engage the local features extracted by the LCEM, we design a novel feature cross-reconstruction module (CRM). The query feature maps $\boldsymbol{\mathcal{B}}_q$ and $\boldsymbol{\mathcal{Z}}_q$ are both cross-reconstructed based on both $\boldsymbol{\mathcal{B}}_s$ and $\boldsymbol{\mathcal{Z}}_s$, which forms four reconstruction tasks: $\boldsymbol{\mathcal{B}}_q$ using $\boldsymbol{\mathcal{B}}_s$, $\boldsymbol{\mathcal{B}}_q$ using $\boldsymbol{\mathcal{Z}}_s$, $\boldsymbol{\mathcal{Z}}_q$ using $\boldsymbol{\mathcal{B}}_s$, and $\boldsymbol{\mathcal{Z}}_q$ using $\boldsymbol{\mathcal{Z}}_s$, as shown in Fig. 2. We follow the strategy of ridge regression in Wertheimer et al. [15] for all reconstructions tasks.

To fit to the setting of ridge regression, the feature representations of the query image and the support images have to be pooled to two matrices, respectively. With abuse of notation to improve the readability of this section, we denote the matrix for the query image by $\mathbf{Q} \in \mathbb{R}^{M\times C}$ and that for the support samples from the nth class by $\mathbf{S}_n \in \mathbb{R}^{KM\times C}$, where $M = H \times W$. Since the reconstruction calculations for images with the base and LCE features are the same, we present one general example of reconstructing $\mathbf{Q}$ by $\mathbf{S}_n$ here.

TABLE I
THE NOTATIONS OF THE CROSS-RECONSTRUCTED QUERY FEATURES BY THE SUPPORT FEATURES OF THE nTH CLASS.

Query \ Support	Base features	LCE features
Base features	$\hat{\mathbf{Q}}_n^{BB}$	$\hat{\mathbf{Q}}_n^{BZ}$
LCE features	$\hat{\mathbf{Q}}_n^{ZB}$	$\hat{\mathbf{Q}}_n^{ZZ}$

Specifically, the query image $\mathbf{Q}$ can be reconstructed by the weighted sum of all support samples in the nth class $\mathbf{S}_n$:

$$\mathbf{Q} = \mathbf{W}\mathbf{S}_n + \mathbf{E}, \tag{2}$$

where $\mathbf{W} \in \mathbb{R}^{M \times KM}$ is the weight matrix and $\mathbf{E} \in \mathbb{R}^{M \times C}$ is the residual. The weights are learnt from ridge regression by solving the following least-squares problem:

$$\mathbf{W}^* = \arg\min_{\mathbf{W}} \|\mathbf{Q} - \mathbf{W}\mathbf{S}_n\|_2^2 + \lambda \|\mathbf{W}\|_2^2, \tag{3}$$

where λ is the constant that controls the bias-variance trade-off. Here we follow the suggestion from Wertheimer et al. [15] to set $\lambda = \frac{KM}{C} e^{\eta}$, where η is learnt from meta training and the normalising factor $\frac{KM}{C}$ is to ensure a stable training process.

The weight matrix $\mathbf{W}$ has a closed-form solution of

$$\mathbf{W}^* = \mathbf{Q}\mathbf{S}_n^T \left(\mathbf{S}_n\mathbf{S}_n^T + \lambda \mathbf{I}\right)^{-1}, \tag{4}$$

where $\mathbf{I} \in \mathbb{R}^{KM \times KM}$ is an identity matrix. Thus the reconstructed query image by the nth class is calculated as

$$\hat{\mathbf{Q}}_n = \gamma \mathbf{W}^* \mathbf{S}_n, \tag{5}$$

where we set $\gamma = e^{\tau}$ with τ learnt from the training process to make γ nonnegative.

For each query image, we have two matrices reshaped from the two types of feature representations: the matrix from the base representation, $\mathbf{Q}^B$; and that from the LCE representation, $\mathbf{Q}^Z$. The reconstruction of $\mathbf{Q}^B$ by the base representation of the nth class of the support set is denoted by $\hat{\mathbf{Q}}_n^{BB}$, while that based on the LCE representation is denoted by $\hat{\mathbf{Q}}_n^{BZ}$. Similarly, the reconstruction of $\mathbf{Q}^Z$ based on the base representation is denoted by $\hat{\mathbf{Q}}_n^{ZB}$, while that based on the LCE representation is denoted by $\hat{\mathbf{Q}}_n^{ZZ}$. We illustrate this notation in Table I to avoid confusion.

The final reconstruction error for the query image based on the nth class of the support set is a weighted sum of the errors of the four reconstruction tasks:

$$\begin{aligned} e_n = &\frac{1}{M}(\omega_{BB}||\mathbf{Q}^B - \hat{\mathbf{Q}}_n^{BB}||_2^2 + \omega_{BZ}||\mathbf{Q}^B - \hat{\mathbf{Q}}_n^{BZ}||_2^2 \\ &+ \omega_{ZB}||\mathbf{Q}^Z - \hat{\mathbf{Q}}_n^{ZB}||_2^2 + \omega_{ZZ}||\mathbf{Q}^Z - \hat{\mathbf{Q}}_n^{ZZ}||_2^2), \end{aligned} \tag{6}$$

where ω_{BB}, ω_{BZ}, ω_{ZB} and ω_{ZZ} are the learnable weights associated with each reconstruction task, respectively.

Based on this reconstruction error, we calculate the predicted probability of the query image belonging to the nth class as

$$P(\hat{y}_j = n \mid \mathbf{x}_j) = \frac{e^{-\delta e_n}}{\sum_{n' \in \mathcal{N}} e^{-\delta e_{n'}}}. \tag{7}$$

For each task, the network is trained by the cross-entropy loss:

$$L = -\sum_{j=1}^{Nq} \log(P(\hat{y}_j = y_j \mid \mathbf{x}_j)). \tag{8}$$

For a test image, we calculate its predicted probabilities in (7) for all classes and classify it to the class with the highest probability.

IV. EXPERIMENTS

Experiments in this section serve five purposes:

1) To compare the proposed LCCRN with state-of-the-art methods for few-shot fine-grained image classification;
2) To investigate the impact of the new LCEM and CRM on the classification performance of LCCRN;
3) To assess the effect of the numbers of ways, shots and epochs;
4) To visualise and compare the base features and LCE features obtained by the LCEM;
5) To visualise the reconstructed images generated from the CRM.

A. Datasets

In the experiments, we test four fine-grained datasets: CUB [30], Aircraft [31], Flowers [32] and Cars [33].

The CUB dataset has 200 classes and 11,788 images of birds, and we randomly divide it into a training set with 100 classes, a validation set with 50 classes and a test set with 50 classes. In addition, we crop each image to a human annotated bounding box following the pre-processing method of [9], [34].

The Aircraft dataset contains 100 aircraft classes with 10,000 images, and we randomly select 50 classes for the training set, 25 classes for the validation set and 25 classes for the test set. Images are pre-cropped to the provided bounding box.

The Flowers dataset consists of 102 categories of common flowers, and each category consists of 40 to 256 images with large-scale, pose and light variations. We randomly divide this dataset into a training set with 51 classes, a validation set with 26 classes and a test set with 25 classes.

The Cars dataset contains 16,185 images of 196 classes of cars. We randomly select 98 classes to form the training set, 48 classes for the validation set and 48 classes for the test set.

We also test the classification performance of LCCRN on two coarse-grained datasets: mini-ImageNet [6] and few-shot CIFAR (FC100) [35].

The mini-ImageNet dataset has 100 classes of 60,000 images, with each class containing 600 images. We randomly select 64 classes for the training set, 16 classes for the validation set and 20 classes for the test set.

The FC100 dataset contains 20 high-level classes with 100 object categories, and each object category has 600 images. The training set is consisting of 12 high-level classes with 60 object categories, the validation set is consisting of 4 high-level classes with 20 object categories and the test set is also consisting of 4 high-level classes with 20 object categories.

TABLE II
THE 5-WAY FEW-SHOT CLASSIFICATION ACCURACIES ON THE CUB, AIRCRAFT,FLOWERS AND CARS DATASETS FOR THE CONV-4 BACKBONE.

Model	CUB		Aircraft		Flowers		Cars	
	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
MatchingNet [6]	60.06±0.88	74.57±0.73	58.23±0.89	74.90±0.66	71.89±0.90	85.46±0.59	44.73±0.77	64.74±0.72
ProtoNet [7]	61.82±0.23	83.37±0.15	50.90±0.22	80.65±0.15	64.23±0.23	84.94±0.16	48.42±0.22	71.38±0.18
Relation [8]	63.94±0.92	77.87±0.64	61.73±0.98	75.96±0.72	69.50±0.96	83.91±0.63	46.04±0.91	68.52±0.78
Baseline++ [23]	62.36±0.84	79.08±0.61	58.38±0.83	77.62±0.60	70.54±0.84	86.63±0.58	46.82±0.76	68.20±0.72
DSN [24]	71.57±0.92	83.51±0.60	66.30±0.87	79.00±0.61	67.71±0.92	84.58±0.70	48.16±0.86	60.77±0.75
DN4 [18]	57.45±0.89	84.41±0.58	68.41±0.91	87.48±0.49	70.44±0.95	89.45±0.52	34.12±0.68	87.47±0.47
BSNet(D&C) [25]	62.84±0.95	85.39±0.56	56.51±1.09	70.80±0.81	66.60±1.04	80.42±0.75	40.89±0.77	86.88±0.50
MixFSL [26]	53.61±0.88	73.24±0.75	44.89±0.75	62.81±0.73	68.01±0.90	85.10±0.62	44.56±0.80	59.63±0.79
FRN [15]	73.46±0.21	88.13±0.13	69.29±0.22	83.94±0.13	73.60±0.22	88.69±0.14	64.03±0.22	84.02±0.13
TDM [27]	74.39±0.21	88.89±0.13	69.90±0.23	83.34±0.15	70.66±0.24	85.14±0.17	65.89±0.22	82.45±0.15
Ours	**76.22±0.21**	**89.39±0.13**	**76.81±0.21**	**88.21±0.11**	**75.57±0.22**	**90.15±0.12**	**71.62±0.21**	**86.41±0.12**

TABLE III
THE 5-WAY FEW-SHOT CLASSIFICATION ACCURACIES ON THE CUB, AIRCRAFT , FLOWERS AND CARS DATASETS FOR THE RESNET-12 BACKBONE.

Model	CUB		Aircraft		Flowers		Cars	
	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
MatchingNet [6]	73.02±0.88	85.17±0.60	82.20±0.80	88.99±0.50	75.70±0.88	87.61±0.55	73.32±0.93	87.61± 0.55
ProtoNet [7]	79.64± 0.20	91.15±0.11	86.57±0.18	93.51±0.09	75.41±0.22	89.46±0.14	82.29±0.20	93.11±0.10
Relation [8]	63.94±0.92	77.87±0.64	74.20±1.04	86.62±0.55	69.51±1.01	86.84±0.56	69.67±1.01	84.29±0.68
Baseline++ [23]	64.62±0.98	81.15±0.61	74.51±0.90	88.06±0.44	69.03±0.92	85.72±0.63	67.92±0.92	84.17±0.58
DeepEMD [9]	71.11±0.31	86.30±0.19	69.86±0.30	85.17±0.28	70.00±0.35	83.63±0.26	73.30±0.29	88.37±0.17
VFD [28]	79.12±0.83	91.48±0.39	76.88±0.85	88.77±0.46	76.20±0.92	89.90±0.53	77.52±0.85	90.76±0.46
RENet [29]	79.49±0.44	91.11±0.24	82.04±0.41	90.50±0.24	79.91±0.42	92.33±0.22	79.66±0.44	91.95±0.22
MixFSL [26]	67.87±0.94	82.18±0.66	60.55±0.86	77.57±0.69	72.60±0.91	86.52±0.65	58.15±0.87	80.54±0.63
FRN [15]	**83.11±0.19**	92.49±0.11	87.53±0.18	93.98±0.09	81.07±0.20	92.52±0.11	86.48±0.18	94.78±0.08
TDM [27]	82.41±0.19	92.37±0.10	88.35±0.17	94.36±0.08	82.85±0.19	93.60±0.10	86.91±0.17	96.11±0.07
Ours	82.97±0.19	**93.63±0.10**	**88.48±0.17**	**94.61±0.08**	**86.95±0.19**	**96.13±0.08**	**87.04±0.17**	**96.19±0.07**

TABLE IV
THE ABLATION STUDY OF REMOVING BOTH THE LCEM AND CRM IN SETTING (A), ONLY THE LCEM IN SETTING (B) AND THE CRM IN SETTING (C), ON THE CUB, FLOWERS, AIRCRAFT AND CARS DATASETS FOR THE 5-WAY FEW-SHOT SCHEME.

	LCEM	CRM	CUB		Flowers		Aircraft		Cars	
			1-shot	5-shot	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
(a)	✗	✗	74.02	88.74	73.63	88.59	67.90	84.03	63.86	84.37
(b)	✗	✓	74.16	88.81	74.77	89.37	68.54	84.57	65.34	85.36
(c)	✓	✗	75.98	88.82	73.58	88.70	76.02	88.04	70.50	85.77
LCCRN	✓	✓	**76.22**	**89.39**	**75.57**	**90.15**	**76.81**	**88.21**	**71.62**	**86.41**

B. Implementation details

We conduct experiments under two widely-used backbone architectures for few-shot image classification: Conv-4 [36], [37] and ResNet-12 [37], [38]. We train the models under the Conv-4 and ResNet-12 backbones for 800 and 1,200 epochs, respectively.

The initial learning rate is set to 0.1 and the weight decay is set to $5e^{-4}$. After every 400 epochs, the learning rate decreases by a factor of 10. We train the network by the 20-way 5-shot setting for the Conv-4 backbone and the 10-way 5-shot setting for the ResNet-12 backbone. LCCRN based on Conv-4 backbone has 0.3M parameters with a computational cost of 0.2G FLOPs, while that based on ResNet-12 backbone has 25M parameters with a computational cost of 7G FLOPs.

In addition, we select the best-performing model based on the validation set and validate it every 20 epochs. For all experiments, we report the average classification accuracies with 95% confidence intervals of 10,000 randomly generated tasks on the test datasets under the standard 5-way 1-shot, and 5-shot settings.

C. Comparisons with the state-of-the-arts

In order to verify the effectiveness of our method in fine-grained few-shot classification, we reproduce the results of the classic few-shot image classification methods (MatchingNet [6], ProtoNet [7], Relation [8] and Baseline++ [23]) and those

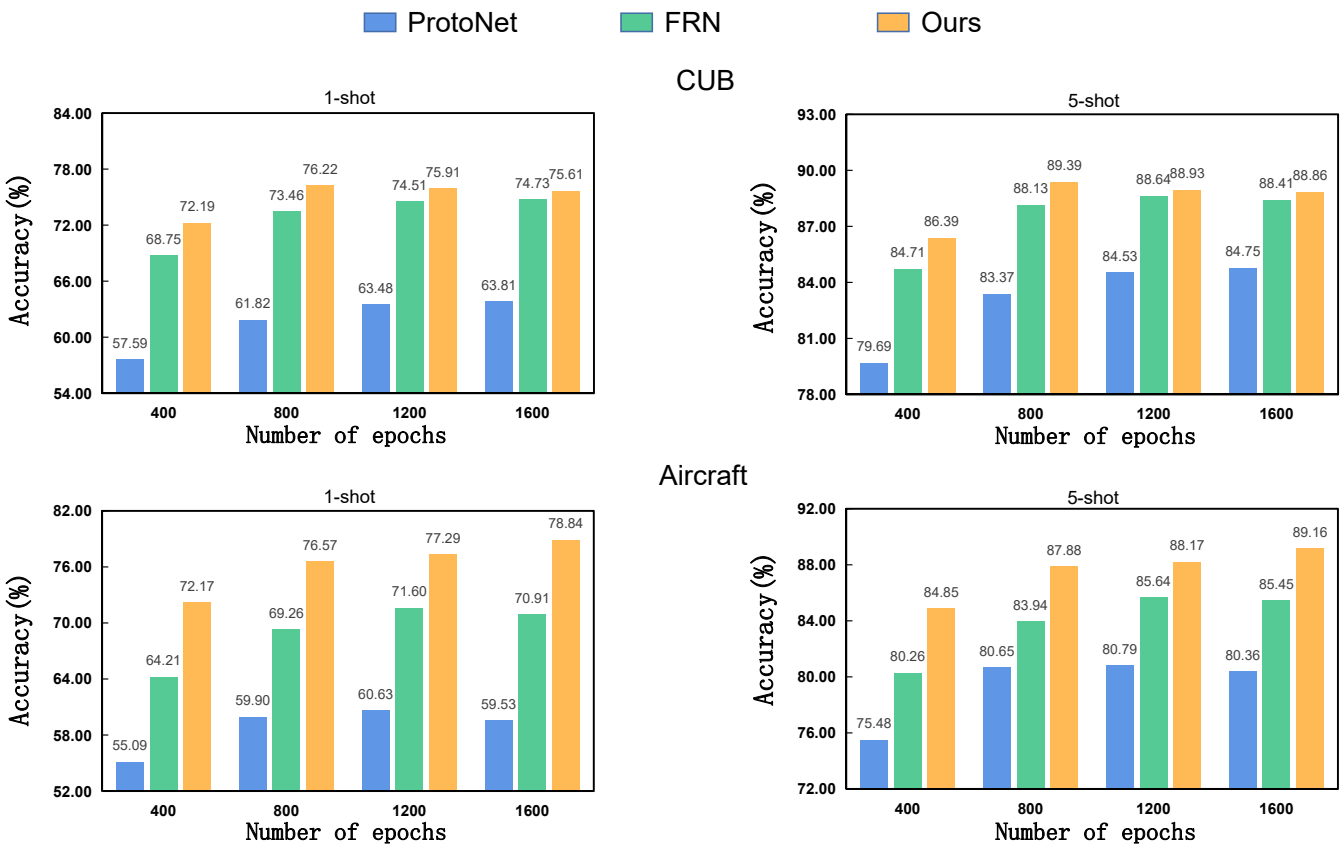


Fig. 4. The barplots of the classification accuracy against the number of epochs on the CUB and Aircraft datasets for the 5-way 1-shot and 5-way 5-shot schemes of our proposed LCCRN (orange), FRN (green) and ProtoNet (blue), respectively.

TABLE V
THE CLASSIFICATION ACCURACIES OF DIFFERENT NUMBER OF SHOTS UNDER THE CONV-4 BACKBONE FOR THE CUB AND AIRCRAFT DATASETS.

Model	Backbone	CUB					Aircraft				
		1-shot	3-shot	5-shot	7-shot	9-shot	1-shot	3-shot	5-shot	7-shot	9-shot
ProtoNet [26]	Conv-4	61.82	79.13	83.37	85.33	86.40	50.90	76.90	80.65	81.98	83.08
FRN [15]	Conv-4	73.46	85.98	88.13	89.95	90.63	69.29	81.03	83.94	85.05	85.70
Ours	Conv-4	**76.22**	**86.98**	**89.39**	**90.21**	**90.67**	**76.81**	**85.96**	**88.21**	**88.53**	**89.05**

of the latest few-shot image classification methods (DSN [24], DN4 [18], BSNet [25], DeepEMD [9], VFD [28], RENet [29], MixFSL [26], FRN [15] and TDM [27]). The datasets used in these methods are also used in this paper.

The classification accuracies of 5-way few-shot classification accuracies on the fine-grained datasets are summarised in Tables II and III for the Conv-4 and ResNet-12 backbones, respectively. It is obvious that our proposed LCCRN has the best mean classification accuracies for all settings, except for the 5-way 1-shot classification of the CUB dataset with the ResNet-12 backbone. Moreover, the widths of the confidence intervals of our results are also noticeably narrower than most of the state-of-the-art methods, which indicates that LCCRN can provide more precise predictions.

D. Ablation study

In this section, we first investigate the impact of the new LCEM and CRM on the classification performance of LCCRN on the four fine-grained datasets under the Conv-4 backbone. We then study the impact of the number of ways, shots and epochs on the classification performance of LCCRN on the CUB and Aircraft datasets under the Conv-4 backbone. Lastly, we provide some visual comparisons of the base and LCE features to show the advance of the LCE features in capturing semantic structures.

1) The impact of the LCEM and CRM: In Table IV, we summarise the classification accuracies of removing both the LCEM and CRM in setting (a) and either of them in settings (b) and (c). In setting (b), when the LCEM is removed, we still keep f_ϕ as a usual embedding module and conduct cross-reconstruction tasks based on the based features extracted from f_ϕ and f_ψ. The experiments are conducted under the Conv-4 backbone for all four fine-grained datasets with the same parameter settings in section IV-B and the results of the 5-way 1-shot and 5-shot experiments are reported.

It is apparent that removing both modules (setting (a))

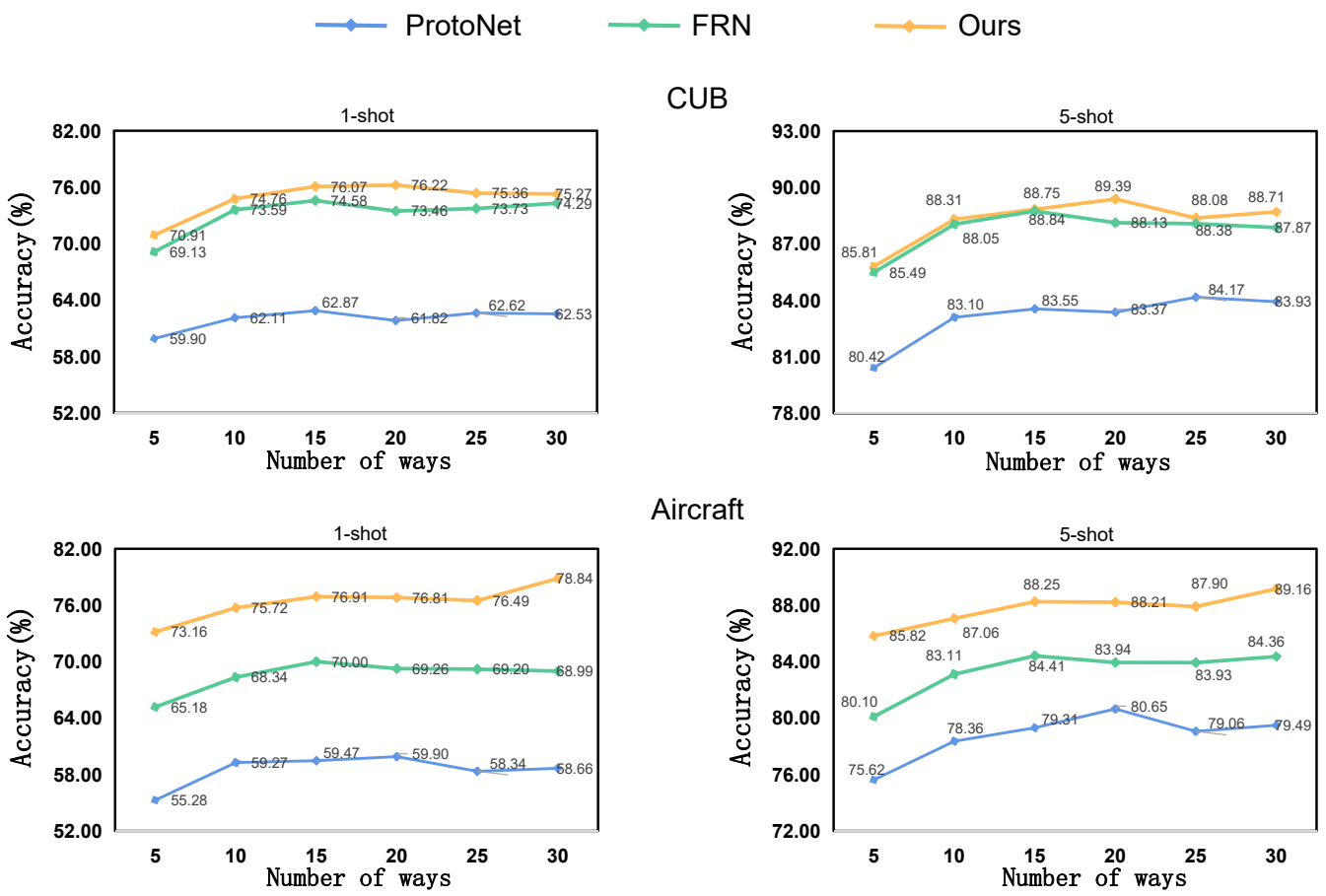


Fig. 5. The curves of the classification accuracy against the number of ways on the CUB and Aircraft datasets for the 1-shot and 5-shot schemes of our proposed LCCRN (orange), FRN (green) and ProtoNet (blue), respectively.

performs the worst in almost all cases. The classification accuracies of LCCRN are also remarkably higher than those without the LCEM in setting (b), with the largest difference of 8.27% for the 5-way 1-shot classification on the Aircraft dataset. Even the two separate usual embedding modules in setting (b) cannot provide competitive classification performance compared with explicitly involving the LCEM, which demonstrates the importance of the local features learnt by the LCEM.

Moreover, in setting (c), we modify the CRM in LCCRN to a parallel-reconstruction version by removing the "cross" tasks. That is, the base query features are reconstructed by their corresponding base support set features while the LCE query features are reconstructed by their corresponding LCE support set features, and e_n in (6) only involves the two terms with ω_{BB} and ω_{ZZ}. Clearly, cross-exploiting the differences and commonalities of the two types of features can further boost the classification performance. However, we note that the accuracy of the Flowers data under the 1-shot setting is slightly below the base accuracy in setting a). This is because the colour, shape and texture of the petals of some flower images from different classes are extremely similar, which makes the local information less valuable for such classification task. Thus simply considering the additional local content information in the parallel-reconstruction fashion without properly fusing it with the global base features could potentially lower the classification performance.

To further demonstrate the roles of the four reconstruction tasks, we present the four weights in equation (6) for the CUB and Aircraft data in Table VI. It is clear that the weights associated with the two cross tasks, i.e. ω_{ZB} and ω_{BZ}, have higher values, which suggests that the cross tasks weigh more to determine the classification results for the two datasets.

TABLE VI
THE WEIGHTS OF THE FOUR RECONSTRUCTION TASKS IN EQUATION (6) FOR THE CUB AND AIRCRAFT DATA.

	ω_{ZZ}	ω_{ZB}	ω_{BZ}	ω_{BB}
CUB	0.578	7.001	2.152	0.830
Aircraft	0.241	5.558	3.979	1.298

2) The impact of the number of ways, epochs and shots: In this section, we compare the classification performances of LCCRN with the classic metric-based method ProtoNet [7] and the latest and most relevant FRN [15] with different numbers of ways, shots and epochs in the training process. We set the initial learning rate to 0.1 and scale down the learning rate by a factor of 10 every 400 epochs.

In Fig. 4, we plot the classification accuracy against the number of epochs on the CUB and Aircraft datasets for the

TABLE VII
THE CLASSIFICATION ACCURACIES OF FRN AND LCCRN ON TWO COARSE-GRAINED DATASETS, THE MINI-IMAGENET AND THE FC100.

	mini-ImageNet				FC100			
	Conv-4		ResNet-12		Conv-4		ResNet-12	
	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
FRN	53.15±0.20	**71.22±0.66**	**62.57±0.21**	77.24±0.15	**37.33±0.17**	**51.18±0.17**	**38.80±0.17**	**52.44±0.18**
Ours	**53.93±0.20**	70.41±0.16	62.24±0.20	**79.19±0.14**	35.38±0.16	48.76±0.17	38.71±0.17	52.24±0.17

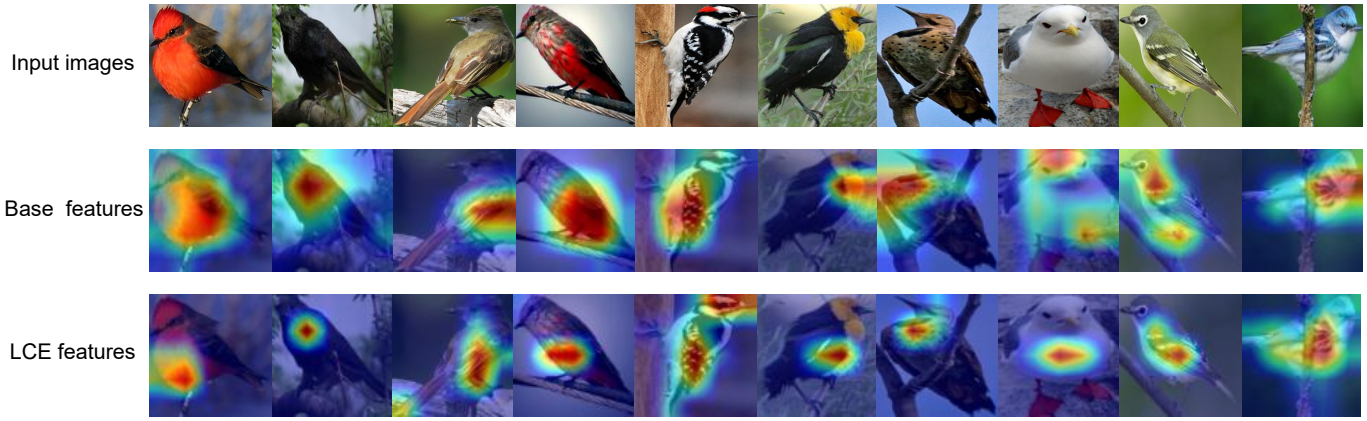


Fig. 6. The visual comparisons of the base and LCE features. The LCEM extracts more semantic information compared with the base embedding module, allowing the LCE features to focus more on the birds themselves and reducing the effect of the background.

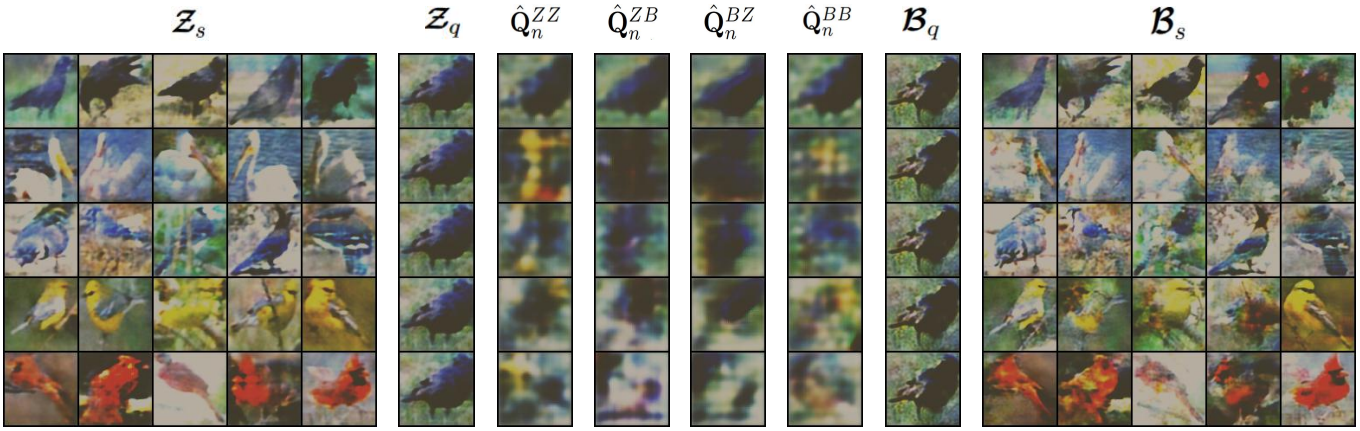


Fig. 7. The visualization of the reconstructed images in the CRM via the ResNet decoder on the CUB dataset, with $\mathcal{Z}$ and $\mathcal{B}$ as inputs. The two large blocks on the two ends are $\mathcal{Z}_s$ and $\mathcal{B}_s$, respectively, showing five classes of birds in each row. The middle four columns are the reconstructed images for the query with features $\mathcal{Z}_q$ and $\mathcal{B}_q$. The top row is for the same class of the query and the rest four rows are for the classes different from the query.

5-way 1-shot and 5-way 5-shot schemes. We can observe that for different datasets, LCCRN achieves the highest accuracy with different number of epochs: for the CUB dataset, 800 is the best number of epochs while for the Aircraft dataset, 1,600 is the best. It is also worth noting that the classification accuracies of LCCRN are better than those of ProtoNet and FRN for all numbers of epochs considered, which demonstrates the superiority classification performance of LCCRN.

The impact of the number of ways is depicted in Fig. 5. Clearly, the number of ways do affect the classification performance: in general, the classification accuracy is higher when the number of ways is large. Similarly to the pattern of the number of epochs, the number of ways required for the best classification accuracy varies for different datasets and methods. For example, for the CUB dataset, when the number of ways is larger than 20, there is a slight decrease in the classification accuracy of LCCRN. Nevertheless, Fig. 5 shows that our proposed LCCRN is better than ProtoNet and FRN for all numbers of ways considered.

Finally, the results of the number of shots are reported in Table V. We adopt the same training setting as in section IV-B, but change the number of shots for test. It is clear that our LCCRN has the best classification accuracies for all number of shots on the CUB and Aircraft datasets. We also note that the

gaps between the classification accuracies of the three methods become larger as the number of shots decreases, which also demonstrates the superior performance of LCCRN on the more challenging tasks with smaller numbers of shots.

E. The classification performance of LCCRN on coarse-grained images

Here we compare the classification accuracies of LCCRN with FRN on two coarse-grained datasets, mini-ImageNet [6] and few-shot CIFAR (FC100) [35]. We adopt the same training strategy as for fine-grained datasets. The results in Table VII indicate that LCCRN cannot beat FRN for classification of coarse-grained data in most cases. One potential reason is that coarse-grained data usually have larger inter-class differences compared with fine-grained data. Thus classifiers focus more on the global features of the entire image, such as FRN, can achieve better results for coarse-grained data. In contrast, LCCRN involving extensive amount of local discriminative details may bring confusing information to classification, and hence has a lower classification performance.

F. Visual comparisons of the base and LCE features

To compare the base and LCE features, we visualise few examples of the bird class of the CUB dataset in Fig. 6. Here we train LCCRN with 20 ways, 5 shots and 800 epochs under the Conv-4 backbone. The results confirm that the LCEM pays more attention to the local information by learning the features related to the local neighborhood, so that the effect of the background part is reduced. Moreover, the LCE features can also focus on the more discriminative areas to distinguish different species of birds.

G. Visualization of the reconstructed images

To visualize the reconstructed images and illustrate the effectiveness of the CRM, we train the reverse ResNet as a decoder to visualize the base features, LCE features and reconstruction features. To train the decoder, we adopt the Adam optimizer with an initial learning rate of 0.01, set the batch size to 100 and train 1000 epochs to measure the prediction error with L1 loss. The decoded image features can then be obtained by feeding the base, LCE and reconstructed features generated from the trained LCCRN to the decoder. The recovered images are of size $3 \times 84 \times 84$.

In Fig. 7, we visualise the two types of embeddings, $\boldsymbol{\mathcal{Z}}$ and $\boldsymbol{\mathcal{B}}$, and the four reconstructed images of the query image by 5-shot reconstruction for images in the CUB dataset. Clearly, the four reconstructed images based on the same category are visually more similar to the query image, but with differences in morphology and colour. This suggests that the four reconstruction tasks can provide complementary information about the query image. On the contrary, the reconstructed images based on the different categories are visually different from the query image, which result in large reconstruction errors.

V. Conclusion

In this work, we propose the new LCCRN for few-shot fine-grained image classification, which extracts LCE features and cross-reconstructs the base and LCE features for better semantic content understanding. The experiments on four fine-grained datasets demonstrate the superior classification performance of LCCRN to the state-of-the-art methods.

References

[1] X. Li, Z. Sun, J.-H. Xue, and Z. Ma, "A concise review of recent few-shot meta-learning methods," *Neurocomputing*, vol. 456, pp. 463–468, 2021.

[2] J. Shu, Z. Xu, and D. Meng, "Small sample learning in big data era," *ArXiv*, vol. abs/1808.04572, 2018.

[3] X. Shu, J. Tang, G.-J. Qi, Z. Li, Y.-G. Jiang, and S. Yan, "Image classification with tailored fine-grained dictionaries," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 2, pp. 454–467, 2018.

[4] H. Huang, J. Zhang, J. Zhang, J. Xu, and Q. Wu, "Low-rank pairwise alignment bilinear network for few-shot fine-grained image classification," *IEEE Transactions on Multimedia*, vol. 23, pp. 1666–1680, 2020.

[5] X. Li, J. Wu, Z. Sun, Z. Ma, J. Cao, and J.-H. Xue, "BSNet: Bi-similarity network for few-shot fine-grained image classification," *IEEE Transactions on Image Processing*, vol. 30, pp. 1318–1331, 2020.

[6] O. Vinyals, C. Blundell, T. P. Lillicrap, K. Kavukcuoglu, and D. Wierstra, "Matching networks for one shot learning," in *Advances in Neural Information Processing Systems (NIPS)*, 2016.

[7] J. Snell, K. Swersky, and R. S. Zemel, "Prototypical networks for few-shot learning," *ArXiv*, vol. abs/1703.05175, 2017.

[8] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. S. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1199–1208, 2018.

[9] C. Zhang, Y. Cai, G. Lin, and C. Shen, "DeepEMD: Few-shot image classification with differentiable earth mover's distance and structured classifiers," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12 200–12 210, 2020.

[10] W. Li, L. Wang, J. Huo, Y. Shi, Y. Gao, and J. Luo, "Asymmetric distribution measure for few-shot learning," in *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, 2021, pp. 2957–2963.

[11] W. Jiang, K. Huang, J. Geng, and X. Deng, "Multi-scale metric learning for few-shot learning," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, pp. 1091–1102, 2021.

[12] F. Zhou, L. Zhang, and W. Wei, "Meta-generating deep attentive metric for few-shot classification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, pp. 6863–6873, 2022.

[13] P. Bateni, R. Goyal, V. Masrani, F. D. Wood, and L. Sigal, "Improved few-shot visual classification," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14 481–14 490, 2020.

[14] H. Huang, J. Zhang, L. Yu, J. Zhang, Q. Wu, and C. Xu, "TOAN: Target-oriented alignment network for fine-grained image categorization with few labeled samples," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 2, pp. 853–866, 2022.

[15] D. Wertheimer, L. Tang, and B. Hariharan, "Few-shot classification with feature map reconstruction networks," *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8008–8017, 2021.

[16] V. G. Satorras and J. Bruna, "Few-shot learning with graph neural networks," *ArXiv*, vol. abs/1711.04043, 2017.

[17] C. Chen, K. Li, W. Wei, J. T. Zhou, and Z. Zeng, "Hierarchical graph neural networks for few-shot learning," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 1, pp. 240–252, 2022.

[18] W. Li, L. Wang, J. Xu, J. Huo, Y. Gao, and J. Luo, "Revisiting local descriptor based image-to-class measure for few-shot learning," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7253–7260, 2019.

[19] Z. Wu, Y. Li, L. Guo, and K. Jia, "PARN: Position-aware relation networks for few-shot learning," *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6658–6666, 2019.

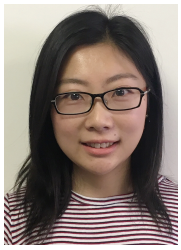
[20] F. Hao, F. He, J. Cheng, L. Wang, J. Cao, and D. Tao, "Collect and select: Semantic alignment metric learning for few-shot learning," *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8459–8468, 2019.

[21] C. Doersch, A. Gupta, and A. Zisserman, "CrossTransformers: spatially-aware few-shot transfer," *ArXiv*, vol. abs/2007.11498, 2020.
[22] F. Hao, F. He, J. Cheng, and D. Tao, "Global-local interplay in semantic alignment for few-shot learning," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 7, pp. 4351 – 4363, 2022.
[23] W.-Y. Chen, Y.-C. Liu, Z. Kira, Y. Wang, and J.-B. Huang, "A closer look at few-shot classification," *ArXiv*, vol. abs/1904.04232, 2019.
[24] C. Simon, P. Koniusz, R. Nock, and M. T. Harandi, "Adaptive subspaces for few-shot learning," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4135–4144, 2020.
[25] X. Li, J. Wu, Z. Sun, Z. Ma, J. Cao, and J.-H. Xue, "BSNet: Bi-similarity network for few-shot fine-grained image classification," *IEEE Transactions on Image Processing*, vol. 30, pp. 1318–1331, 2021.
[26] A. Afrasiyabi, J.-F. Lalonde, and C. Gagn'e, "Mixture-based feature space learning for few-shot image classification," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9021–9031, 2021.
[27] S. Lee, W. Moon, and J.-P. Heo, "Task discrepancy maximization for fine-grained few-shot classification," *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5321–5330, 2022.
[28] J. Xu, H. M. Le, M. Huang, S. Athar, and D. Samaras, "Variational feature disentangling for fine-grained few-shot classification," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8792–8801, 2021.
[29] D. Kang, H. Kwon, J. Min, and M. Cho, "Relational embedding for few-shot classification," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8802–8813, 2021.
[30] C. Wah, S. Branson, P. Welinder, P. Perona, and S. J. Belongie, "The caltech-UCSD birds-200-2011 dataset," 2011.
[31] S. Maji, E. Rahtu, J. Kannala, M. B. Blaschko, and A. Vedaldi, "Fine-grained visual classification of aircraft," *ArXiv*, vol. abs/1306.5151, 2013.
[32] M.-E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*. IEEE, 2008, pp. 722–729.
[33] J. Krause, M. Stark, J. Deng, and L. Fei-Fei, "3d object representations for fine-grained categorization," *2013 IEEE International Conference on Computer Vision Workshops*, pp. 554–561, 2013.
[34] H.-J. Ye, H. Hu, D.-C. Zhan, and F. Sha, "Few-shot learning via embedding adaptation with set-to-set functions," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8805–8814, 2020.
[35] B. Oreshkin, P. Rodríguez López, and A. Lacoste, "TADAM: Task dependent adaptive metric for improved few-shot learning," in *Advances in Neural Information Processing Systems*, 2018.
[36] M. Tong, S. Wang, B. Xu, Y. Cao, M. Liu, L. Hou, and J.-Z. Li, "Learning from miscellaneous other-class words for few-shot named entity recognition," *arXiv preprint arXiv:2106.15167*, 2021.
[37] K. Lee, S. Maji, A. Ravichandran, and S. Soatto, "Meta-learning with differentiable convex optimization," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10 649–10 657, 2019.
[38] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

Xiaoxu Li received the Ph.D. degree from Beijing University of Posts and Telecommunications in 2012. She is currently an Associate Professor with the School of Computer and Communication, Lanzhou University of Technology. Her research interests include machine learning fundamentals with a focus on applications in image and video understanding. She is also a member of the China Computer Federation.

Song Qi received the B.E. degree in computer science and technology from Gansu Agricultural University in 2019. He is a postgraduate student in Lanzhou University of Technology. His research interests include computer vision and few-shot learning.

Jijie Wu received the M.E. degree in Software Engineering from Lanzhou University of Technology, China, in 2019. He is currently working toward the PhD degree with Lanzhou University of Technology. His research interests include computer vision and few-shot learning.



Rui Zhu received the Ph.D. degree in statistics from University College London in 2017. She is a Senior Lecturer in the Faculty of Actuarial Science and Insurance, City, University of London. Her research interests include machine learning and its applications in image quality assessment, hyperspectral image analysis and actuarial science.

Zhanyu Ma is currently a Professor at Beijing University of Posts and Telecommunications, Beijing, China, since 2019. He received the Ph.D. degree in electrical engineering from KTH Royal Institute of Technology, Sweden, in 2011. From 2012 to 2013, he was a Postdoctoral Research Fellow with the School of Electrical Engineering, KTH. He has been an Associate Professor with the Beijing University of Posts and Telecommunications, Beijing, China, from 2014 to 2019. His research interests include pattern recognition and machine learning fundamentals with a focus on applications in computer vision, multimedia signal processing. He is a Senior Member of IEEE.

Jing-Hao Xue received the Dr.Eng. degree in signal and information processing from Tsinghua University in 1998 and the Ph.D. degree in statistics from the University of Glasgow in 2008. He is a Professor in the Department of Statistical Science at University College London. His research interests include statistical pattern recognition, machine learning and computer vision. He is an Associate Editor of the IEEE Transactions on Circuits and Systems for Video Technology, the IEEE Transactions on Cybernetics, and the IEEE Transactions on Neural Networks and Learning Systems.