Applied Intelligence (2025) 55:96
https://doi.org/10.1007/s10489-024-05926-9

Manifold and patch-based unsupervised deep metric learning for fine-grained image retrieval

Shi-hao Yuan[1] · Yong Feng[1] · A-Gen Qiu[2] · Guo-fan Duan[3] · Ming-liang Zhou[1] · Bao-hua Qiang[4] · Yong-heng Wang[5]

Accepted: 16 October 2024 / Published online: 7 December 2024

Abstract

Accurately and swiftly retrieving from fine-grained images is a critical and challenging task. As the key technology for fine-grained image retrieval, deep metric learning aims to learn a mapping space, where samples exhibit two properties: positive concentration and negative separation, facilitating the measurement of similarities between samples. Unsupervised deep metric learning, which obviates the need for labels during training, has garnered widespread attention compared to its supervised counterparts due to its convenience. Current methods in unsupervised deep metric learning face issues such as imbalance in sample construction, difficulty in sample differentiation, and neglect of intrinsic image features. To address these challenges, we propose Manifold and Patch-based Unsupervised Deep Metric Learning (MPUDML) for Fine-Grained Image Retrieval. Specifically, we adopt a manifold similarity-based balanced sampling strategy for constructing more balanced mini-batch samples. Moreover, we leverage soft supervision information obtained from the manifold and cosine similarities between unlabeled images for sample differentiation, effectively reducing the impact of noisy samples. Additionally, we utilize the rich feature information between internal image patches through image patch-level clustering and localization tasks to guide the acquisition of a more comprehensive feature embedding representation, thereby enhancing retrieval performance. Our method, MPUDML, was evaluated against various state-of-the-art unsupervised deep metric learning approaches in fine-grained image retrieval and clustering tasks. Experimental findings indicate that our MPUDML method exceeds other advanced methods in recall ($R@K$) and Normalized Mutual Information (NMI).

Keywords Unsupervised metric learning · Fine-grained image retrieval · Manifold similarity · Patch-level features

1 Introduction

Image data contains a wealth of visual and semantic content, and image retrieval is widely applied in reality. For instance, in e-commerce platforms, users can search for similar clothes or other products by taking photos [1]. Unlike general image retrieval, fine-grained image retrieval [2] focuses more on images with only subtle differences between categories (for example, images of different species of birds). The larger intra-class variance and smaller inter-class variance make this task extremely challenging. Measuring the similarity between images is a crucial step in image retrieval tasks. Metric learning, aimed at establishing a measure of similarity between samples, has been introduced as a research method applied to various computer vision tasks such as face recognition [3], image classification [4], and pedestrian re-identification [5]. Metric learning seeks to learn a mapping space where sample data exhibits two characteristics: positive concentration and negative separation, meaning that samples of the same category are closer together while samples of different categories are further apart, thereby accomplishing the measurement of similarity between samples.

With the rise of deep learning techniques, deep Metric Learning (DML) has emerged, which integrates the concepts of metric learning into deep neural networks. DML automatically extracts and learns salient feature representations of data by training deep models, enabling these features to maintain the semantic similarity among samples in the embedding space. This capability endows DML with significant advantages in processing complex, high-dimensional data. However, in practical applications, acquiring labeled data is often costly and time-consuming, which limits the widespread application of supervised DML methods. To overcome this challenge, Unsupervised Deep Metric Learning (UDML)

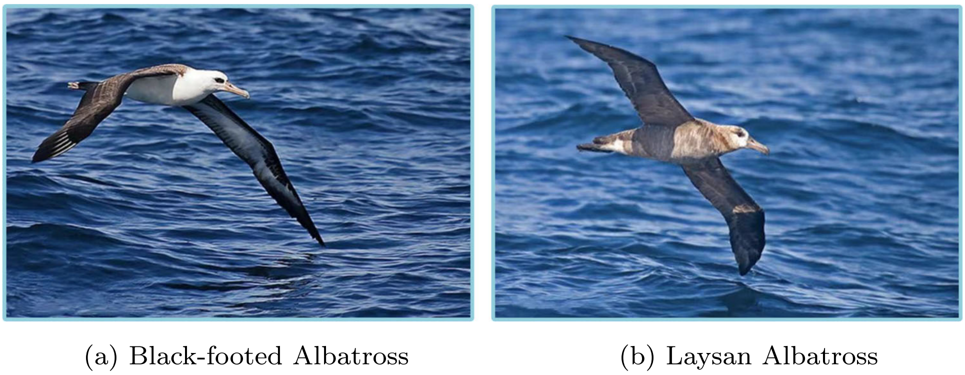
Extended author information available on the last page of the article arises. UDML aims to automatically discover and utilize potential semantic structures and patterns from unlabeled data to learn effective feature representations. This characteristic makes UDML particularly crucial when dealing with large-scale, unlabeled datasets, especially in real-world scenarios where labeled data is scarce. Our work precisely focuses on UDML. Despite the satisfactory performance achieved by existing unsupervised deep metric learning methods [6–8], there are still inherent limitations. Firstly, the selection of mini-batch samples is crucial for improving model performance and convergence [9]. Traditional mini-batch selection methods are often based on simple random sampling or distance-based sampling, but these methods may not effectively capture the complex relationships and correlations between data. We introduce manifold similarity to guide the selection of mini-batch samples. In this way, the model can receive more comprehensive information during each iteration, making it easier to learn the distinguishing features between different categories. Secondly, sample differentiation poses a key challenge in unsupervised learning. Most existing methods rely on pseudo labels generated by clustering for positive and negative sample mining, yet clustering algorithms are sensitive to initialization and the preset number of clusters. Therefore, the effectiveness can be heavily influenced by the quality of pseudo labels, which in turn compromises model accuracy and performance. Lastly, images contain a wealth of semantic information [10, 11]. However, current methods primarily focus on the global feature embedding relationships between image samples, neglecting the rich intra-image feature interactions, which can lead to the loss of discriminative feature information.

Among the nearest neighbor images of anchor samples, i.e., images that are close in the feature embedding space, some are noisy data that lack similarity to the anchor image, which may affect retrieval performance. As illustrated in Fig. 1, the two images are mostly similar except for the main target, which only has subtle differences. The main target in Fig. 1(a) belongs to the species Black-footed Albatross, while the main target in Fig. 1(b) belongs to the species Laysan Albatross. This indicates that the two images are different at a higher semantic level, yet their feature embedding distances are close. Consequently, Fig. 1(a) could be one of the K-nearest neighbors of Fig. 1(b). Based on clustering by feature embedding distance, Fig. 1(a) would be mistakenly identified as the same category as Fig. 1(b), thereby misleading the model and degrading retrieval performance.

Based on the issue we mentioned, utilizing the intrinsic manifold structure can effectively reduce the impact of noisy data points. As illustrated in Fig. 2, the red dots represent the query sample points, and the yellow data points are its K-nearest neighbors defined by feature distance. The points are separated into two categories based on the manifold structure, where the yellow triangles, representing noise data points, are on a different manifold from the red dots. This indicates that it is possible to distinguish noisy data points from K-nearest neighbor data points by relying on the manifold structure.

Based on the above analysis, we propose the Manifold and Patch-based Unsupervised Deep Metric Learning method, which leverages manifold similarity and image patch-level features to mitigate the impact of noisy sample points on model training. This approach enables more comprehensive feature embedding representations, achieving positive concentration and negative separation of unlabeled image samples in the embedding space. The main contributions of our study are summarized as follows:

- We formulate a balanced sampling strategy based on manifold similarity to guide the selection of mini-batch samples, addressing the issue of sample imbalance.
- We obtain soft supervision information based on the cosine similarity and manifold similarity between images, distinguishing between positive and negative samples to reduce the impact of noisy samples.
- We employ image patch-level clustering tasks and patch-level localization tasks, utilizing the rich information among internal image patches to guide the acquisition of more comprehensive feature embedding representations.
- We evaluated our method against various advanced unsupervised deep metric learning approaches through



(a) Black-footed Albatross

(b) Laysan Albatross

Fig. 1 The two schematic images of different categories

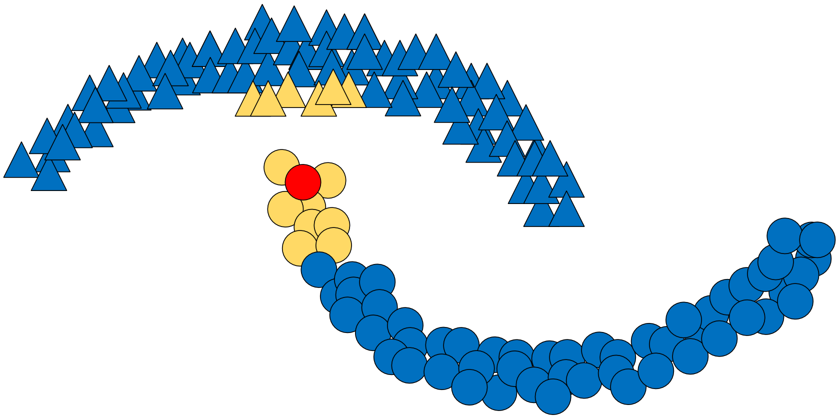


Fig. 2 The diagram of the manifold of samples

comparative experiments, ablation studies, and sensitivity analysis, validating its superiority and effectiveness.

The rest of the paper is organized as follows: Section 2 introduces related work, Section 3 provides a detailed description of the proposed MPUDML method. Subsequently, Section 4 presents and discusses our experiments on standard benchmarks. Finally, Section 5 concludes the paper.

2 Related works

2.1 Mini-batch selection

To enhance the convergence of optimization methods for classification tasks, research has been conducted on strategies for selecting mini-batches [12–15]. However, the validity of these studies regarding the creation of mini-batches remains limited. Random sampling is a simple and commonly used batch selection method, but it ignores the intrinsic relationships between data and may not efficiently utilize the information in the training data. Hard negative mining [15] is widely used to select samples that are difficult for the model to classify correctly as part of the training data. However, hard negative mining may lead to sampling bias, where the model focuses excessively on specific hard-to-classify samples while neglecting other equally important samples. Many studies [16, 17] utilize the distance distribution between samples to guide sampling, such as selecting a certain number of the nearest samples as a mini-batch. This method considers the spatial relationships between data to some extent, but it may not perform well in the initial stages due to lack of sufficient context information. Inspired by [9, 18], we further improve our method by introducing manifold similarity in an unsupervised manner, elegantly formulating a balanced sampling strategy to guide the selection of mini-batch samples. Compared to distance-based sampling guidance, manifold similarity-based mini-batch construction is more representative and information-rich. Specifically, we first randomly sample some samples and then obtain several samples with the highest manifold similarity to each sample, thereby forming a mini-batch. This approach not only takes into account the category diversity within the mini-batch but also ensures a uniform distribution of categories in the mini-batch data. This allows the model to receive more comprehensive information during each iteration, making it easier to learn distinguishing features between different categories and enhancing the model's generalization ability.

2.2 Deep metric learning

In recent years, deep metric learning has achieved significant breakthroughs in the field of computer vision. Initial attempts in DML algorithms were made with the Contrastive loss [19], which captures the similarity between pairs of data points. By taking pairs of samples as input, it strives to reduce the distance within samples from the same category while increasing the distance between samples from different categories. Building upon Contrastive loss, Triplet loss not only strives to reduce the intra-class embedding distances and increase the inter-class embedding distances but also takes into account their relative positioning. For each anchor point, Triplet loss [20] selects a positive and a negative sample, ensuring that the distance between the anchor and the negative sample is greater than the distance between the anchor and the positive sample by a predetermined margin. N-pair Loss [21] and Lifted Struct Loss [22] extend this concept by allowing joint comparisons with N-1 samples and all negative samples respectively, rather than selecting a single negative sample for each class, to generalize the triad of losses. To prevent the discard of potentially useful samples, Wang et al. introduced the Ranked List Loss [23], which segregates samples into positive and negative sets and utilizes the information from all samples to construct a similarity structure. Differing from the aforementioned distance-based metrics, Angular Loss [24] learns similarity measurements by constraining the angle within the negative sample of a triplet. Moreover, approaches such as Hierarchical Triplet Loss [25], Attention-based Ensemble [26], Adaptively Hardness-aware Deep Metric Learning [27], and Multi-Similarity Loss [28] are all designed to optimize the distances or similarities between global feature embeddings of samples, aiming to consolidate positive consolidation and negative separation.

Existing DML methods [25–28] are often sensitive to noise and outliers, as they directly rely on distance or cosine similarity calculations between samples. These noisy samples can mislead model learning, leading to performance degradation. They focus on optimizing global feature embeddings but neglect finer-grained information within images, such as local features or patch-level features. This limits the model's generalization ability in complex scenarios. Manifold similarity can capture the similarity of data on the manifold structure, thereby reducing dependence on noisy samples. Our proposed method combines cosine similarity between images and manifold similarity to obtain more comprehensive soft supervision information. Our method not only considers global features but also guides the learning of feature embeddings through rich patch-level information within images, thereby enhancing the model's generalization ability. We adopt a combined loss function that adds manifold similarity metric loss to patch-level localization loss and patch-level clustering loss. This combination can simultaneously optimize global and local feature embeddings, improving model performance.

2.3 Unsupervised metric learning

Current unsupervised metric learning methods typically "create" supervisory information through self-supervised representation learning and clustering approaches. Based on the characteristic of transformation invariance, which posits that sample data remains semantically consistent even after transformations, several studies [29–33] consider transformed data as positive samples to obtain discriminative embedding representations for supervisory information. Inspired by human visual experience, Li et al. proposed an unsupervised metric learning approach based on transformation attention consistency and contrastive clustering loss, employing an attention mechanism to learn the Siamese Deep Neural Network. This method acquires inter-class discriminative features based on the consistency of transformed attentions [30]. Beyond category supervision, Ye et al. suggested utilizing instance-level supervisory information by treating instances subjected to random data transformations as positive samples, and different instances as negative samples [31]. Instance-based supervision[34–36] plays a pivotal role in unsupervised learning.

Clustering methods are widely applied in unsupervised metric learning. Dutta et al. introduced Orthogonality based Probabilistic Unsupervised Metric Learning, which uses graph-based clustering to acquire pseudo labels, forming triplets to guide the metric learning process and compensating for the absence of category labels [37]. Additionally, Dutta et al. presented Synthetic Unsupervised pseudo Metric Learning, which forms triplets through random sampling of unlabeled data, considering all possible semantic combinations and generating synthetic constraints [37]; Mining On Manifold leverages manifold space for hard training sample mining [38]. Although the aforementioned methods achieve robust performance, their effectiveness is easily impacted by the quality of pseudo labels and tends to overlook the rich feature information within images.

3 Proposed method

3.1 Overview

Our objective is to learn a feature embedding space for unlabeled data samples, wherein the samples exhibit two properties: positive concentration and negative separation. Addressing the limitations present in current unsupervised deep metric learning approaches, we introduce the Manifold and Patch-based Unsupervised Deep Metric Learning method .

As illustrated in Fig. 3, the framework of our proposed unsupervised deep metric learning method, MPUDML, can be segmented into four modules: the sampling module, global manifold similarity computation module, distance matrix computation module, and patch-level feature learning module, with the entire process being trained in an end-to-end manner.

In the sampling module, the training dataset is leveraged to construct mini-batches through a balanced sampling strategy, as illustrated in the purple box I in Fig. 3. The global manifold similarity computation module employs a pre-trained network to extract global features from the training data, facilitating the computation of manifold similarities to differentiate between positive and negative samples. It also incorporates sample mining and weighting strategies for metric measurements of sample pairs, as depicted in the purple box II in Fig. 3. The distance matrix computation module (depicted in the purple box III in Fig. 3) utilizes the pre-trained network to extract feature embeddings from the mini-batch data, which are then used to compute the distance matrix. By combining the preceding similarity matrix and distance matrix, a manifold similarity metric loss is derived. The patch-level feature learning module (illustrated in the purple box IV in Fig. 3) involves segmenting each image in the training batch into patches and randomly arranging

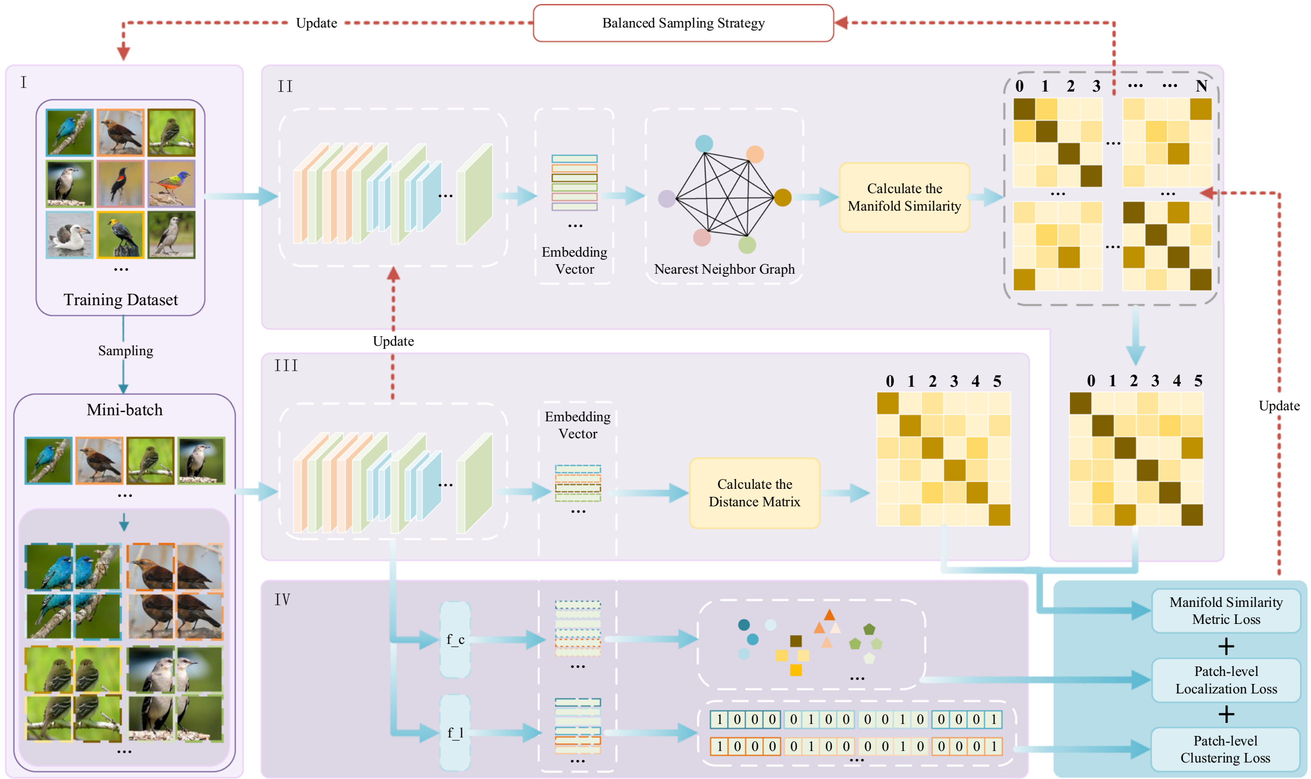


Fig. 3 The overall frame diagram of the proposed method

them. The feature embeddings obtained through the feature extraction module are utilized to predict patch-level clustering and localization tasks. Clustering across different patches from multiple images aids the model in learning cross-patch image-level features; the localization task distinguishes the original positions of individual patches within the image, enabling the model to capture intra-image information. Ultimately, our loss function comprises three primary components: patch-level clustering loss, patch-level localization loss, and a deep metric learning loss based on manifold similarity. Among them, minimizing the patch-level clustering loss and patch-level localization loss can help the model learn rich feature information and positional information within the image. Minimizing the deep metric learning loss based on manifold similarity can improve the model's ability to distinguish different features by capturing subtle differences in the data within the manifold space. The specific content of the loss function will be elaborated in Section 3.5.

3.2 Balanced sampling strategy

Traditional random sampling strategies involve randomly selecting samples from a dataset for training. This approach may overlook some vital information within the dataset, especially when it encounters class imbalance issues. Balanced sampling strategies, by adjusting the number of samples per category [9], enhance the model's ability to recognize all categories, thereby improving the overall performance of the model. However, unsupervised learning, lacking sample labels, can only rely on distance metrics to search for unsupervised balanced sample data, consisting of equivalent data points closest to the cluster centroids in each cluster [6, 18]. Yet, this distance-based method fails to effectively represent the similarity between data points. Therefore, we adopt a balanced sampling strategy based on manifold similarity. Specifically, we first randomly sample a samples, then calculate the manifold similarity between these a samples and all samples in the dataset, ranking them from highest to lowest, and selecting the top b samples. This process yields a mini-batch of ab samples. To ensure the reliability of the balanced sampling strategy, the mini-batch samples for each epoch are rebuilt according to the latest manifold similarity recalculated by the training's best model. The calculation method for manifold similarity is introduced in Section 3.3.

3.3 The acquisition of manifold similarity

For each image sample x_i, we extract its feature vector f_i through the feature extraction network. Subsequently, based on the cosine similarity, we select its K nearest neighbors to form the sample set nn_k^c. The cosine similarity is defined as follows:

$$S_c\left(x_i, x_j\right) = \frac{f_i^T f_j}{|f_i|\left|f_j\right|} \tag{1}$$

Inspired by the work presented in [39], we utilize random walks on nearest neighbor graph to quantify manifold similarities among image sample data points. Nearest neighbor graph is constructed as an undirected weighted graph with image samples from the dataset $\mathcal{X}$ (which contains N training samples in total) serving as nodes. We describe this graph with a sparse symmetric adjacency matrix $G \in \mathbb{R}^{N \times N}$, as described by the following equation:

$$\mathrm{G}_{ij} = \begin{cases} S_c\left(x_i, x_j\right), & x_i \in nn_k^c\left(x_j\right) \cap x_j \in nn_k^c\left(x_i\right) \\ 0, & otherwise \end{cases} \tag{2}$$

In the matrix G, G_{ij} denotes the element located at the i^{th} row and j^{th} column of G. The notation $nn_k^c\left(x_j\right)$ represents the set of K-nearest neighbor samples of the sample x_j. For an undirected weighted graph, if sample x_i and x_j are mutual K-nearest neighbors, the edge weight between them is assigned based on their cosine similarity; otherwise, the weight is set to zero. Additionally, all values on the diagonal of G are set to zero.

For the nearest neighbor graph G, for each node x_i, the following iterative process is adhered to:

$$r_i^t = \alpha \check{G} r_i^{t-1} + (1-\alpha) h_i \tag{3}$$

where $\alpha \in [0, 1]$, $\check{G} = D^{-1/2} G D^{-1/2}$. The modified graph, denoted as $\check{G}$, is defined as $D^{-1/2} G D^{-1/2}$, where D represents a diagonal matrix with D_{ii}, D_{ii} equal to the sum of the elements in the i^{th} row of matrix G. The initial vector $r_i^0 \in R^N$ is chosen arbitrarily, and h_i is a one-hot vector where the i^{th} element is set to 1, with all other elements being 0.

Paper [40] introduces a straightforward algorithm for obtaining smooth solutions. At the heart of this method lies the iterative process of propagating label information from each point to its neighbors until a global stable state is reached. The convergence of this process is demonstrated, enabling the direct calculation of outcomes without the need for repetitive iterations. Consequently, the results $r_{i\ t=0}^{t*}$ converge to r_i^*, that is:

$$r_i^* = \alpha \check{G} r_i^* + (1-\alpha) h_i \tag{4}$$

$$r_i^* = (1-\alpha)\left(I - \alpha \check{G}\right)^{-1} h_i \tag{5}$$

Where I denotes the identity matrix. Consequently, the j^{th} element of r_i^*, denoted as r_{ij}^*, represents the manifold similarity between image x_i and image x_j.

3.4 Sample discrimination based on manifold similarity

For each image x_i, we rank other data points in descending order based on manifold similarity, obtaining a set of the top O data points, denoted as $nn_o^m\left(x_i\right)$. For each image x_i, we have obtained two sets: the set of the top K data points ranked by cosine similarity of image feature embeddings, denoted as $nn_k^c\left(x_i\right)$, and the set of the top O data points ranked by manifold similarity, denoted as $nn_o^m\left(x_i\right)$. If a sample point is in $nn_k^c\left(x_i\right)$ but not in $nn_o^m\left(x_i\right)$, it is considered to be a potential noise point for image x_i, that is, an image not belonging to the same category as x_i. Hence, based on this concept, we reconstruct the data points in $nn_k^c\left(x_i\right)$, treating those that are both in $nn_k^c\left(x_i\right)$ and $nn_o^m\left(x_i\right)$ as positive samples similar to the image; those not in either $nn_k^c\left(x_i\right)$ or $nn_o^m\left(x_i\right)$ as negative samples dissimilar to the image, with the rest considered as ambiguous samples. Consequently, we represent the similarity matrix $\hat{S}$ as:

$$\hat{S}_{ij} = \begin{cases} 1, & x_j \in nn_k^c\left(x_i\right) \cap x_j \in nn_o^m\left(x_i\right) \\ 0, & x_j \notin nn_k^c\left(x_i\right) \cap x_j \notin nn_o^m\left(x_i\right) \\ -1, & otherwise \end{cases} \tag{6}$$

Here, $\hat{S}_{ij}$ represents the element in the i^{th} row and j^{th} column of matrix $\hat{S}$. When $\hat{S}_{ij} = 1$, it indicates that image x_i and image x_j belong to the same category; when $\hat{S}_{ij} = 0$, it indicates that image x_i and image x_j belong to different categories; when $\hat{S}_{ij} = -1$, it suggests that the categorical relationship between image x_i and image x_j is ambiguous.

To ensure the symmetry of the similarity matrix, we adhere to the following rules for further updates: If $\hat{S}_{ij} = 1$ or $\hat{S}_{ji} = 1$, then $\hat{S}_{ij} = \hat{S}_{ji} = 1$; if $\hat{S}_{ij} = 0$ and $\hat{S}_{ji} = 0$, then $\hat{S}_{ij} = \hat{S}_{ji} = 0$; otherwise, $\hat{S}_{ij} = \hat{S}_{ji} = -1$.

Furthermore, the features extracted by pre-trained CNNs contain rich semantic information. This implies that it is possible to unearth semantic similarities from the features of images. Consequently, we integrate cosine similarity and manifold similarity for sample differentiation. For pairs of images with ambiguous similarity, we further define their similarity using the cosine similarity of image feature embeddings, resulting in the final manifold similarity matrix S, as demonstrated in the following formula:

$$\mathrm{S}_{ij} = \begin{cases} \hat{S}_{ij}, & \hat{S}_{ij} \neq -1 \\ S_c\left(x_i, x_j\right), & otherwise \end{cases} \tag{7}$$

In summary, the similarity between samples fall into three cases: positive samples that are completely similar with $S_{ij} = 1$, negative samples that are completely dissimilar with $S_{ij} = 0$, and ambiguous samples that are partially similar with $S_{ij} \in (0, 1)$.

3.5 Objective function

We distinguish between positive, negative, and ambiguous samples based on the final manifold similarity matrix S. Diverging from conventional metric learning that uses discrete labels for supervision, we employ manifold similarity as soft supervision information for metric learning. Thus, we utilize the Relaxed Contrastive Loss [41], as demonstrated in the following formula:

$$\mathcal{L}_{met}=\frac{1}{n}\sum_{i=1}^{n}\sum_{j\neq i}^{n}w_{ij}dis_{ij}+\frac{1}{n}\sum_{i=1}^{n}\sum_{j\neq i}^{n}\left(1-w_{ij}\right)\left[\delta-dis_{ij}\right]_{+} \tag{8}$$

Herein, w_{ij} represents the weight factor, corresponding to the final manifold similarity matrix S. dis_{ij} denotes the squared Euclidean distance between samples x_i and x_j in the embedding space. When $w_{ij} = 1$, indicating positive samples, the loss is represented by the first part, aimed at reducing the distance between positive samples. When $w_{ij} = 0$, indicating negative samples, the loss is represented by the second part, aimed at increasing the distance between negative samples that violate the constraint boundary δ. The more a negative sample violates the constraint, the greater its contribution to the loss function. In this paper, δ is set to 1.0.

For a batch of n images $X = \{x_1, x_2, x_3, \cdots, x_n\}$, each image is divided into $m \times m$ patches, amounting to a total of $n \times m \times m$ patches. Due to the significant variance among different regions of some images, learning becomes challenging without any overlapping areas. Therefore, as illustrated in Fig. 4, we enhance learning by extending intersections, rather than dividing the image into disjoint patches, enabling the network to learn features more effectively. In all our experiments, m is set to 2. Because a larger m would exponentially increase complexity, rendering the network incapable of learning effectively [42].

Through feature extraction, the patches are processed to obtain $n \times m \times m$ features. The patch clustering task aims to cluster the $n \times m \times m$ patches into n clusters, constituting a supervised clustering task, as the $m \times m$ patches belong to the same category. The objective of supervised clustering is to bring patches of the same class closer together and push patches of different classes further apart. Cosine similarity is utilized to measure the distance between patches. Therefore, for each pair of patches z_i, z_j within the same cluster, the loss function is as follows:

$$l_{ij} = -\log \frac{\exp(S_c(z_i, z_j)/\tau)}{\sum_{k=1,k\neq i}^{nmm} \exp(S_c(z_i, z_k)/\tau)} \tag{9}$$

Here, τ represents the temperature parameter. The final loss function for the clustering task is the aggregation of all patch pairs from the same cluster, as follows:

$$\mathcal{L}_{clu} = \frac{1}{nmm} \sum_{i} \left(\frac{1}{mm-1} \sum_{j\in C_i} l_{ij} \right) \tag{10}$$

Where C_i denotes the set of patches that belong to the same cluster as patch z_i.

The patch localization task aims to predict the position of the aforementioned patches within the original image and is considered a classification task. The cross-entropy loss is employed, with the loss function for a single batch formulated as follows:

$$\mathcal{L}_{loc} = CrossEntropy(L,\ L_{gt}) \tag{11}$$

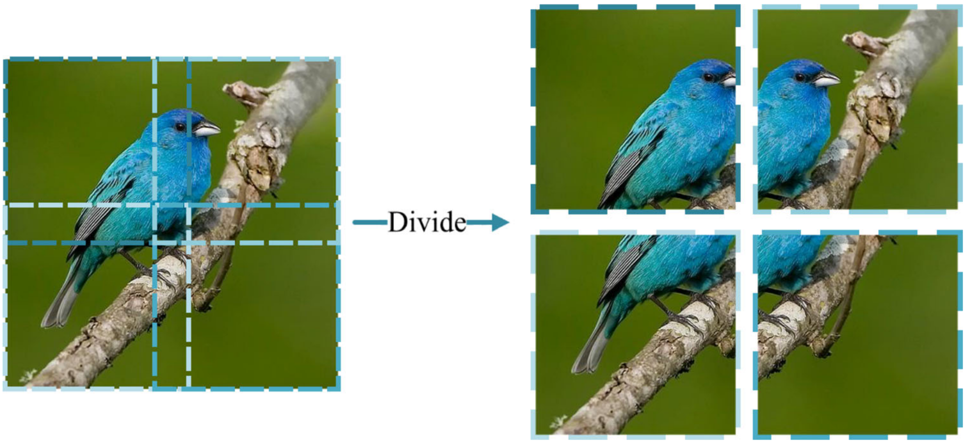


Fig. 4 The diagram of an image divided into patches

Here, L_{gt} represents the ground truth location of the patches, while L denotes the classification labels predicted by the proposed method.

In summary, our final objective function is as follows:

$$\mathcal{L} = \mathcal{L}_{met} + \alpha\mathcal{L}_{clu} + \beta\mathcal{L}_{loc} \tag{12}$$

Here, $\mathcal{L}_{met}$ signifies the metric learning loss based on manifold similarity, $\mathcal{L}_{clu}$ denotes the clustering loss for the patch-level feature learning module and $\mathcal{L}_{loc}$ represents the localization loss for the patch-level feature learning module. α and β are hyperparameters used to weigh the importance of the losses, which are both set to 1.0 in this experiment.

The entire training process of the Manifold and Patch-based Unsupervised Deep Metric Learning algorithm is depicted in Algorithm 1.

Algorithm 1 Training Steps of Our MPUDML.

```
Require: Training dataset $\mathcal{X}$, batch size $n$, epochs, Pretrained GoogLeNet parameters $\theta_g$.
Ensure: Finalized $\theta_g$.
for number of epochs do
    Calculate the cosine similarity matrix and the manifold similarity matrix of the dataset using (1), (5).
    Construct mini-batches $X$ based on the balanced sampling strategy.
    for all mini-batch in $X$ do
        Get the manifold similarity matrix of the mini-batch samples.
        Perform sample classification using (7).
        Calculate $\mathcal{L}_{met}$ using (8).
        Calculate $\mathcal{L}_{clu}$ using (10).
        Calculate $\mathcal{L}_{loc}$ using (11).
        Minimize the sum of $\mathcal{L}_{met}$, $\mathcal{L}_{clu}$ and $\mathcal{L}_{loc}$ through (12).
        Update the model parameters $\theta_g$.
    end for
end for
return Finalized $\theta_g$.
```

4 Experiments

In this section, we conduct extensive experiments on two widely used datasets in fine-grained image retrieval, CUB-200-2011 [43] and Cars196 [44], to evaluate the proposed method and compare it with state-of-the-art approaches.

4.1 Experimental settings

4.1.1 Datasets

We follow the standard data partitioning technique used in previous DML research to separate the training and test datasets, as detailed below:

- **CUB-200-2011** [43] comprises 11,788 images distributed among 200 bird species categories. We allocate the first 100 categories (5,864 images) for training and the latter 100 categories (5,924 images) for testing.
- **Cars196** [44] includes 16,185 images spanning 196 car categories. The initial 98 categories (8,054 images) form our training set, while the subsequent 98 categories (8,131 images) are used for testing.

4.1.2 Evaluation

To effectively assess the effectiveness of the proposed method, we conduct studies on two tasks: fine-grained image retrieval and image clustering, using commonly employed metrics in metric learning tasks, recall ($R@K$) and Normalized Mutual Information (NMI).

- **Recall** is a commonly used evaluation metric in image retrieval tasks, indicating the model's retrieval accuracy calculated based on the retrieval recall results for each image query sample. A higher value indicates higher accuracy. Specifically, for each input image query sample x, when the K nearest samples in the metric space are returned, if they share the same category label with sample x, the score is 1; otherwise, it is 0. $R@K$ represents the average score of the K recalled samples. The detailed calculation formula is as follows:

$$R@K = \frac{1}{K}\sum_{k=1}^{K} score(l_x = l_{x_k}) \tag{13}$$

Where $score()$ represents a conditional function that returns 1 if the label of the retrieved sample matches the label of the query sample x, and 0 otherwise.

- **Normalized Mutual Information** is a commonly used evaluation metric in image clustering tasks, which is employed to measure the similarity between the model's results and the ground truth. NMI is calculated based on the theoretical foundations of information entropy and mutual information.

The formula for calculating information entropy is expressed as follows:

$$H(U) = -\sum_{i=1}^{n} p_i log p_i \tag{14}$$

Where n represents the number of clusters after clustering, and p_i denotes the probability that a sample is assigned to the i-th cluster after clustering.

As a useful information measure in information theory, Mutual Information is obtained based on the joint distribution probability and marginal distribution probabilities of the model's results and the ground truth.

$$I(U,V)=\sum_{i=1}^{|U|}\sum_{j=1}^{|V|}p_{i,j}log\frac{p_{i,j}}{p_i\times p_j} \tag{15}$$

Where $|U|$ represents the number of clusters after clustering by the model, $|V|$ represents the number of true categories in the dataset, and $p_{i,j}$ denotes the joint probability, which is the probability that a sample belongs to both cluster i and category j.
Finally, the NMI formula is obtained through normalization as follows. Its value ranges from [0,1]. If NMI tends to 1, it indicates that the clustering results are closer to the original true category labels, which also indicates better model performance.

$$NMI(U,V)=2\bullet\frac{I(U,V)}{H(U)+H(V)} \tag{16}$$

4.1.3 Implementation details

Our experimental approach utilizes PyTorch, adhering to conventional experimental protocols from previous studies for benchmarking. Due to the success of GoogLeNet in various image recognition applications, we establish GoogLeNet [45] as the feature extraction network and build a fully connected layer. The GoogLeNet used here has been thoroughly pre-trained on the ImageNet dataset, enabling its weights to contain rich image feature information. In this way, we can get feature embeddings. The default size of feature embeddings is set to 1024, with a batch size of 100, training for 50 epochs. The learning rate for the backbone network is set to 2×10^{-5}, and the temperature parameter τ is set to 0.07. To ensure fairness in our experiments, we employ common image preprocessing operations: resizing images in the training set to 256 × 256 to ensure consistent image dimensions, followed by random cropping and horizontal flipping to 224 × 224, which enhances the dataset and mitigates overfitting. On the other hand, images in the test set undergo center cropping only. A stochastic gradient descent optimizer is used in our experiments. According to [23], we set the learning rate for the CUB-200-2011 dataset at 5×10^{-4} because it has fewer images and some of its images overlap with those in ImageNet. For the Cars196 dataset, we set a higher learning rate of 1×10^{-3}. In the balanced sampling strategy, we set $a=20$ and $b=5$. Details on the setting of more hyperparameters will be elaborated in Section 4.4.

4.2 Comparative experiment

To quantitatively assess the effectiveness of the proposed method, we conducted comparative experiments to benchmark MPUDML against nine current advanced unsupervised deep metric learning methods, including: Exemplar [29], NCE [34], DeepCluster [46], MOM [38], AND [35], Instance [31], UMM [47], aISIF [48], and MIGCN [8]. Among them, Exemplar, NCE, AND, Instance, and aISIF are instance-based methods; DeepCluster leverages k-means clustering to acquire pseudo labels for identifying positive and negative samples; MOM engages in hard sample mining through manifold space; UMM addresses outlier issues through uncertain modeling; MIGCN introduces a mutual information graph convolutional network-based method.

In this section, we will present the comparative performance of MPUDML against the aforementioned methods on two public benchmark datasets, CUB-200-2011 and Cars196, for tasks of image retrieval and image clustering, as shown in Tables 1 and 2, with the best results highlighted in bold font. To ensure the fairness of the experiments, all methods were implemented using their optimal experimental settings as reported in their original publications. Missing values in the tables were not found in the original publications.

Table 1 Comparison of the proposed method with other advanced unsupervised methods on the CUB-200-2011 [43], expressed in percentage (%) results

Methods	Backbone	CUB				
		R@1	R@2	R@4	R@8	NMI
Examplar [29]	GoogLeNet	38.2	50.3	62.8	75.0	45.0
NCE [34]	GoogLeNet	39.2	51.4	63.7	75.8	45.1
DeepCluster [46]	GoogLeNet	42.9	54.1	65.6	76.2	53.0
MOM [38]	GoogLeNet	45.3	57.8	68.6	78.4	55.0
AND [35]	GoogLeNet	47.3	59.4	71.0	81.0	–
Instance [31]	GoogLeNet	46.2	59.0	70.1	80.2	55.4
UMM [47]	GoogLeNet	47.3	59.5	70.6	–	56.0
aISIF [48]	GoogLeNet	47.7	59.9	71.2	81.4	–
MIGCN [8]	ResNet18	48.5	60.5	71.9	82.0	56.6
Ours	GoogLeNet	**52.8**	**65.4**	**76.1**	**85.0**	**59.3**

Table 2 Comparison of the proposed method with other advanced unsupervised methods on the Cars196 [44], expressed in percentage (%) results

Methods	Backbone	Cars R@1	R@2	R@4	R@8	NMI
Examplar [29]	GoogLeNet	36.5	48.1	59.2	71.0	35.4
NCE [34]	GoogLeNet	37.5	48.7	59.8	71.5	35.6
DeepCluster [46]	GoogLeNet	32.6	43.8	57.0	69.5	38.5
MOM [38]	GoogLeNet	35.5	48.2	60.6	72.4	**38.6**
AND [35]	GoogLeNet	38.4	49.6	60.2	72.9	–
Instance [31]	GoogLeNet	41.3	52.3	63.6	74.9	35.8
UMM [47]	GoogLeNet	40.5	52.4	63.3	–	37.3
aISIF [48]	GoogLeNet	41.2	52.6	63.8	75.1	–
Ours	GoogLeNet	**41.8**	**53.0**	**64.7**	**75.2**	36.1

In terms of experimental performance improvements, compared to the method aISIF, which also uses GoogLeNet [45] as the backbone, our method shows increases of 5.1%, 5.5%, 4.9%, and 3.6% on the CUB-200-2011 dataset for R@1, R@2, R@4, and R@8, respectively. Even when compared with the latest method MIGCN, which uses ResNet18 [49] as the backbone, our method demonstrates enhancements of 4.3%, 4.9%, 4.2%, 3.0%, and 2.7% for R@1, R@2, R@4, R@8, and NMI, respectively.

On the Cars196 dataset, compared to the aISIF method, which also uses GoogLeNet as the backbone, our method improved by 0.6%, 0.4%, 0.9%, and 0.1% for R@1, R@2, R@4, and R@8, respectively. The experimental results validate the effectiveness of MPUDML in acquiring soft supervisory information through manifold similarity for sample discrimination. Our method performs better on the CUB-200-2011 dataset compared to the Cars196 dataset, which may be attributed to the superior capability of GoogLeNet, pretrained on ImageNet [50], in extracting features of birds over features of cars.

Holistically, our MPUDML method achieves the optimal results on both the CUB-200-2011 and Cars196 datasets. As can be seen in Table 3, our method even surpasses some supervised metric learning methods.

Table 3 Comparison of our unsupervised method with certain supervised methods on the CUB-200-2011 [43], expressed in percentage (%) results

Methods	R@1	R@2	R@4	R@8	NMI
Lifted [22]	43.6	56.6	68.6	79.6	56.5
Clustering [51]	48.2	61.4	71.8	81.9	59.2
Triplet+ [16]	45.9	57.7	69.6	79.8	58.1
Smart+ [16]	49.8	62.3	74.1	83.3	59.9
N-pair [21]	45.4	58.4	69.5	79.5	–
ABIER [52]	57.5	68.7	78.3	86.2	–
Ours	52.8	65.4	76.1	85.0	59.3

4.3 Ablation study

In this section, we verify the impact of each component in the proposed MPUDML method. As previously mentioned, our method principally executes unsupervised metric learning through three branches: a balanced sampling strategy, a task based on manifold similarity metric learning, patch-level clustering tasks, and patch-level localization tasks. Accordingly, we developed the following variants of MPUDML:

- **MPUDML-S** indicates this sub-model adopts a conventional sampling strategy to validate the impact of the balanced sampling strategy on model performance.
- **MPUDML-M** denotes the sub-model that has been stripped of the manifold similarity metric learning task branch from the original method, to ascertain its effect.
- **MPUDML-C** represents the sub-model that has been stripped of the patch-level clustering task branch from the original method.
- **MPUDML-L** signifies the sub-model that has been stripped of the patch-level localization task branch from the original method.

We conducted experiments with the proposed method and the four sub-models on the CUB-200-2011 dataset, respectively, and transformed the experimental data into bar charts for visual representation, as shown in Fig. 5.

As depicted in Fig. 5, each component contributes differently to the model's performance, leading to the following conclusions:

- By comparing the performance of the proposed method, MPUDML, and its sub-model MPUDML-S on the dataset as shown in Fig. 5, it is observed that MPUDML-S exhibits a slight decline in all metrics except for R@1. This validates the effectiveness of the balanced sampling strategy to a certain extent.

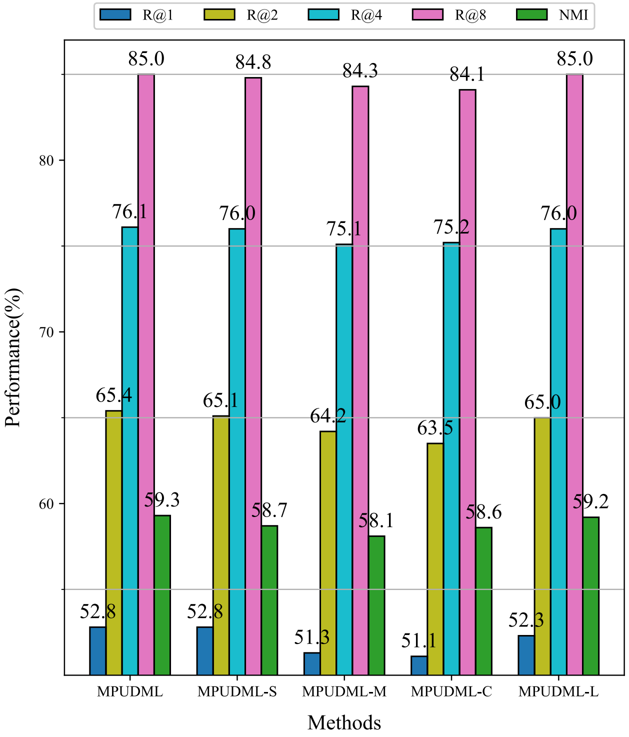


Fig. 5 Bar chart of experimental results for the proposed method and its four sub-models on the CUB-200-2011 dataset

- The comparison between MPUDML and its sub-model MPUDML-M in Fig. 5 reveals a significant decrease in the R@1 and NMI metrics for MPUDML-M. This confirms the efficacy of the manifold similarity-based metric learning task branch, which distinguishes positive and negative samples by leveraging manifold similarity as soft supervisory information between unlabeled samples.
- By examining the impact of the MPUDML-C and MPUDML-L sub-models on the overall performance of MPUDML as illustrated in Fig. 5, it can be inferred that patch-level clustering and localization tasks slightly enhance the model's feature extraction capability, particularly noticeable in the R@1 metric. The patch-level clustering branch facilitates the learning of instance-level and image-level information, while the patch-level localization branch supports the clustering branch by providing more detailed positional information.

In summary, the validity of the balanced sampling strategy, manifold similarity-based metric learning, patch-level clustering, and patch-level localization branches is confirmed. These components collectively enhance the overall performance of the model.

4.4 Sensitive parameter analysis

4.4.1 Impact of embedding dimension

In order to explore the impact of embedding dimensions on experimental outcomes, we conducted experiments on the CUB-200-2011 dataset using multiple different embedding dimensions. As illustrated in Fig. 6, there is a gradual enhancement in the model's retrieval performance on this dataset with the increase in embedding dimensions. However, this trend appears to reach a "critical point," beyond which further increases in dimensions yield only marginal performance improvements. Therefore, the selection of an appropriate dimension should consider not only performance enhancement but also the efficiency and constraints of resource utilization. Based on the experimental outcomes, setting the embedding dimension to 1024 emerges as a prudent choice.

4.4.2 Impact of parameters K and O

In our proposed method, distinguishing samples based on the manifold similarity matrix is crucial, and its effectiveness is related to the selection of K and O values, i.e., K-nearest neighbors based on cosine similarity and O-nearest neighbors based on manifold similarity. Thus, this section evaluates the effect of our method on the CUB-200-2011 dataset with different K and O values.

To study the impact of setting K and O to the same values, we fixed $O = K$ and varied K within the set $\{0.001N, 0.01N, 0.05N, 0.1N, 0.5N\}$, where N is the number of training data samples. We assessed the impact on fine-grained image retrieval and clustering tasks using metrics such as R@1, R@2, R@4, R@8, and NMI, with experimental results shown in Fig. 7. It can be observed that as K varies, the overall performance generally first increases and then decreases. When the value of K is too small, it may lead to some positive samples not being identified; when K is large, some noise samples are mistakenly treated as positive samples, both of which can impact performance. However, by utilizing manifold similarity values as soft supervision and adjusting the degree of attraction and repulsion based on similarity for ambiguous samples, the performance does not fluctuate significantly. In our experiments, K was set to $0.05N$ for optimal performance.

To explore the impact of different ratios of K and O values, we set $K = 0.05N$ and varied the ratio of O to K within the set {0, 0.2, 1.0, 5.0, 10.0}, with results shown in Fig. 8. It can be seen that as the ratio of O to K increases, the overall performance generally exhibits a minor trend of first increasing and then decreasing, with the optimal

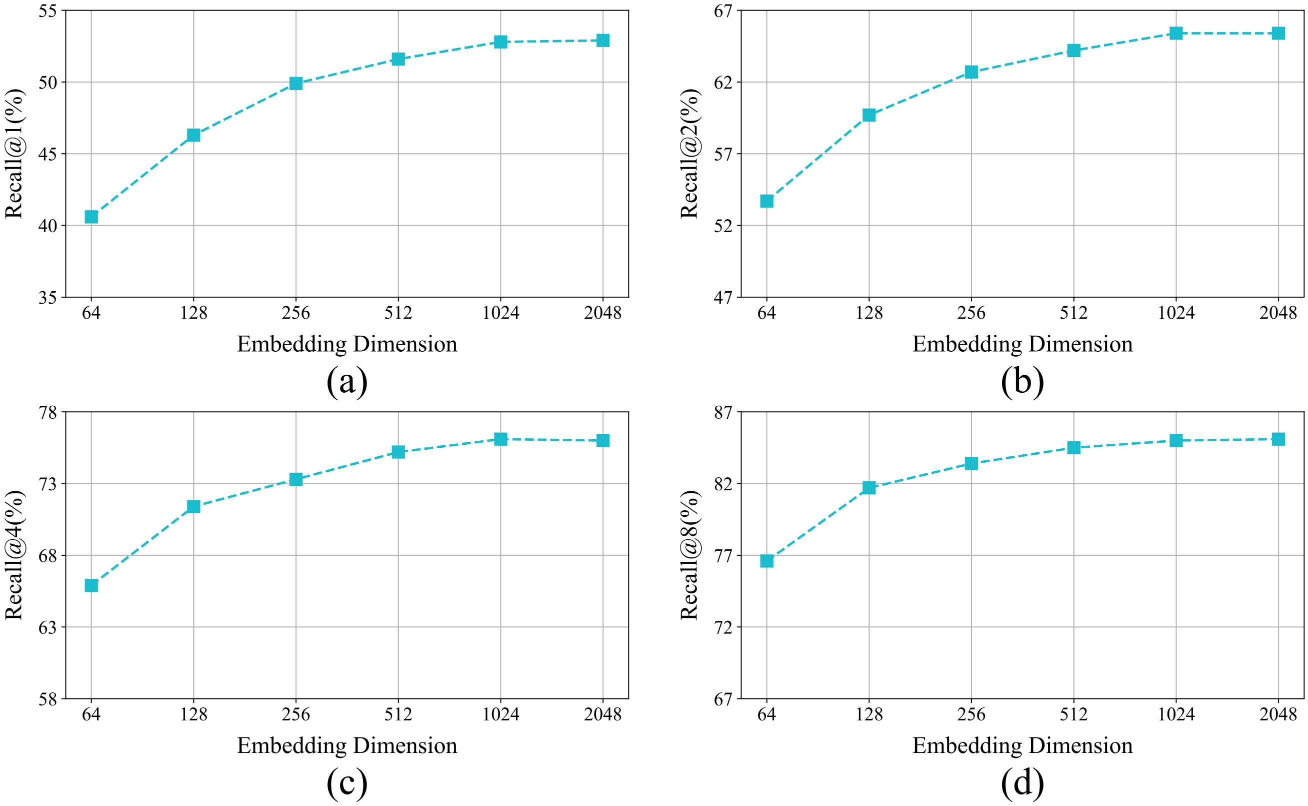


Fig. 6 Comparative results of different embedding dimensions on the CUB-200-2011 dataset

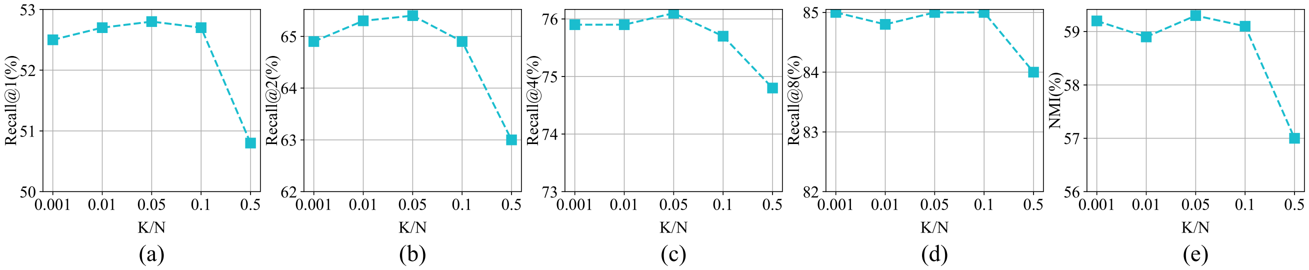


Fig. 7 Comparative results of different K/N values on the CUB-200-2011 dataset

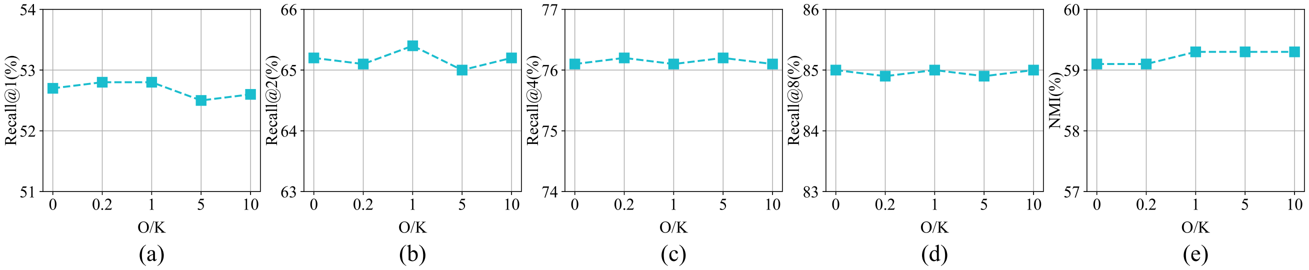


Fig. 8 Comparative results of different O/K values on the CUB-200-2011 dataset

experimental performance occurring when $O = K$. When the ratio of O to K is low, meaning fewer O-nearest neighbors based on manifold similarity than K-nearest neighbors based on cosine similarity, it may fail to correctly differentiate some positive samples, losing some discriminative feature information; when the ratio of O to K is high, meaning more O-nearest neighbors based on manifold similarity than needed, it may introduce incorrect samples, reducing performance. In conclusion, we use a parameter setting of $O = K = 0.05N$.

4.5 Result visualization

4.5.1 Retrieval visualization

To further understand the proposed MPUDML, we randomly selected three images from the CUB-200-2011 and Cars-196 datasets as query samples to demonstrate the fine-grained image retrieval results. As shown in Fig. 9, it is evident that during the early training phase, there are some incorrectly retrieved images. After 50 epochs, the retrieval results improved significantly, with the incorrectly retrieved images highly resembling the query samples. For instance, in the later stages of training, query sample Q1 still exhibited one incorrect retrieval result. The main reason is that the image closely mimicked the appearance and posture of the query sample. This suggests that the proposed MPUDML effectively generates discriminative embeddings that accurately respond to the similarities between images.

Additionally, complex image backgrounds play a crucial role in fine-grained image retrieval tasks. From query sample Q2, we observed that it contained a complex background of branches, which initially caused significant interference in the retrieval results. However, by the later stages of training, more correct images were retrieved, effectively minimizing background distractions. This improvement indicates that the embeddings were trained to focus more on the subject's features rather than the background.

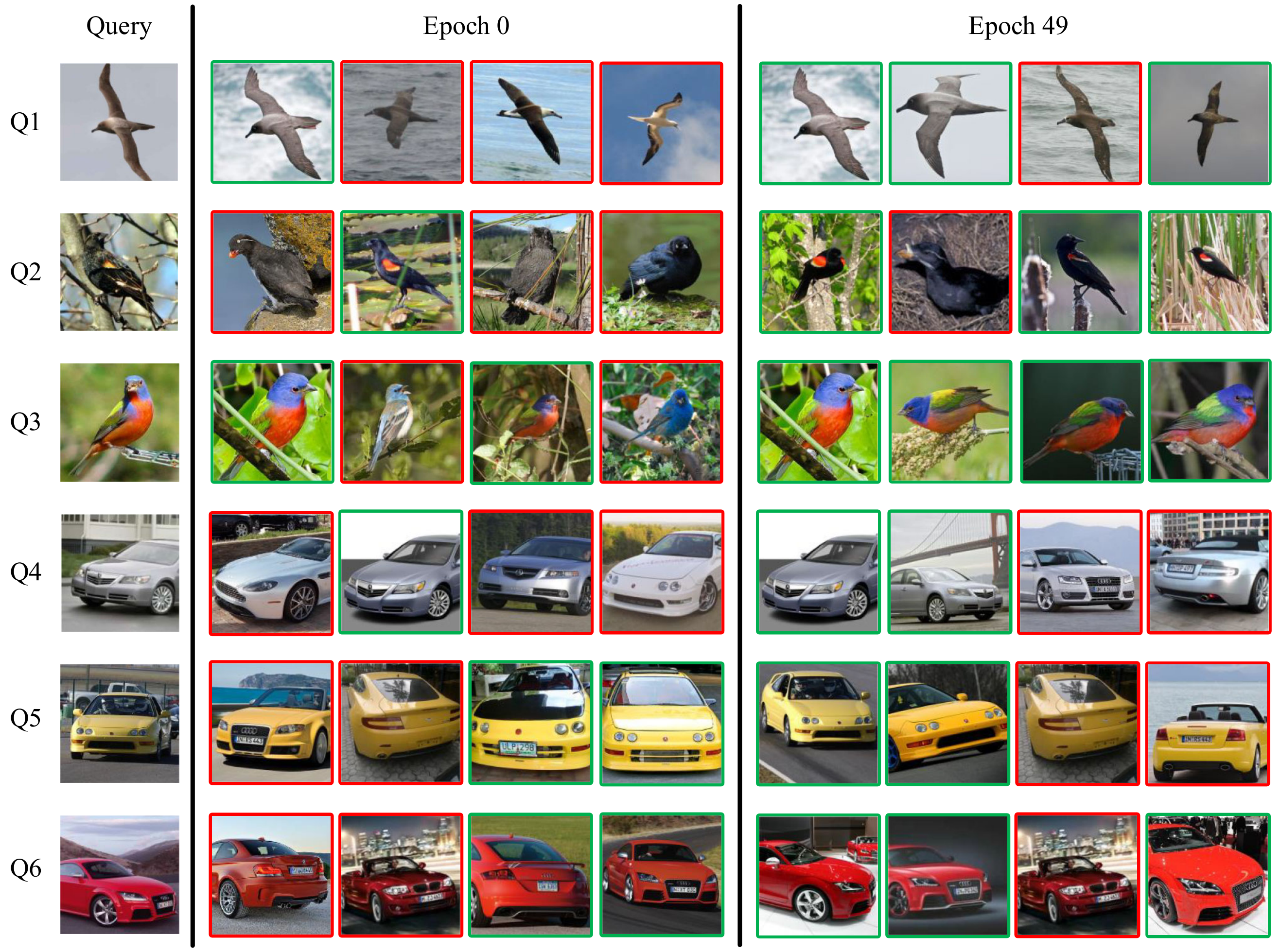


Fig. 9 Using our method with four nearest neighbors set on the CUB-200-2011 and Cars-196 datasets, we generated fine-grained image retrieval results. The leftmost column represents the query samples. Images retrieved within green frames denote correct results, while those in red frames indicate incorrect results

Moreover, after training for 50 epochs, the ranking of erroneously retrieved images also moved further down the list. For example, in the top four images returned for query sample Q5, there were two incorrectly retrieved images both before and after training. However, pre-training these images ranked first and second, whereas post-training, they moved to third and fourth positions, illustrating the effectiveness of the learning process.

Overall, Fig. 9 highlights the significant improvements in retrieval performance in the later stages of training compared to the early stages, thoroughly validating the effectiveness of the learning process.

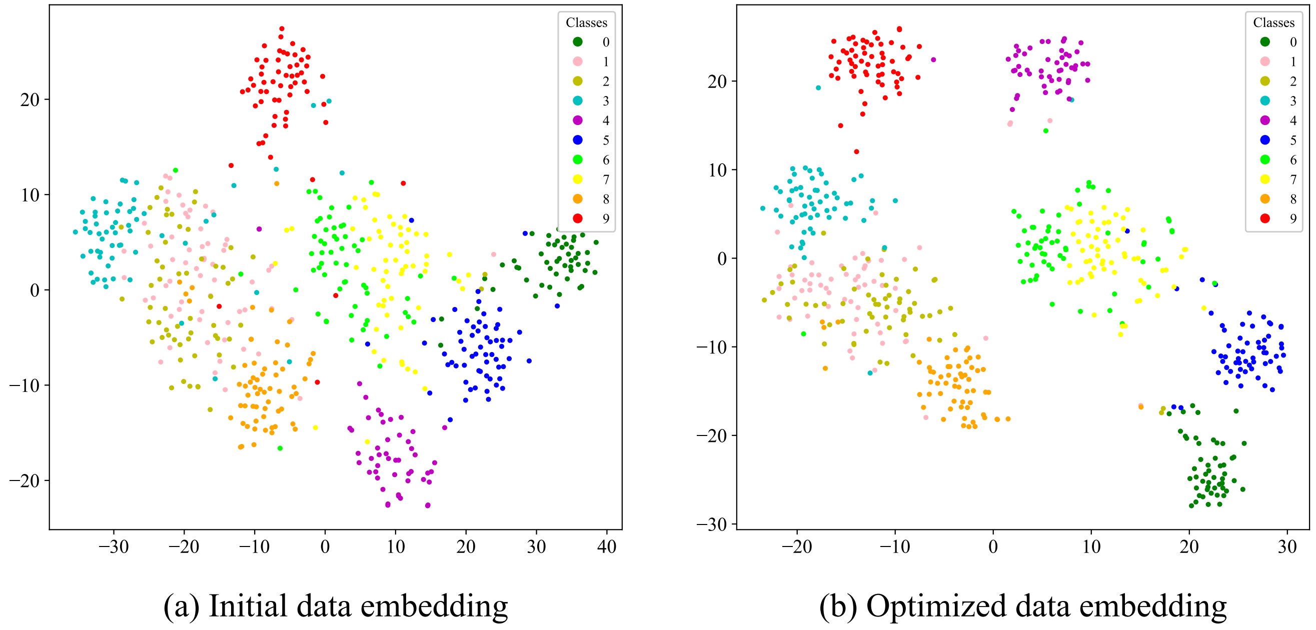
4.5.2 Clustering visualization

To further validate the advanced nature of the proposed method MPUDML, this section presents a visual display of the data embedding distribution. We randomly selected images from ten categories from the CUB-200-2011 and Cars196 datasets. We performed dimensionality reduction on both the initial feature embeddings and the image feature embeddings optimized by the proposed method, and visually displayed them in Figs. 10 and 11, where different colors represent different category samples.

As can be seen from Figs. 10 and 11, the data feature embeddings optimized by our proposed MPUDML method demonstrate characteristics of positive concentration and negative separation. After optimization, the distances within the same category in the embedding space are reduced, while the distances between different categories are increased. For instance, the data samples represented by numbers 4, 5, and 6 in Figure 10, and numbers 2, 7, and 9 in Figure 11, show that the optimized feature embeddings are more compact intra-class and more separated inter-class, reflecting the effectiveness of the proposed MPUDML method.

5 Conclusion and future work

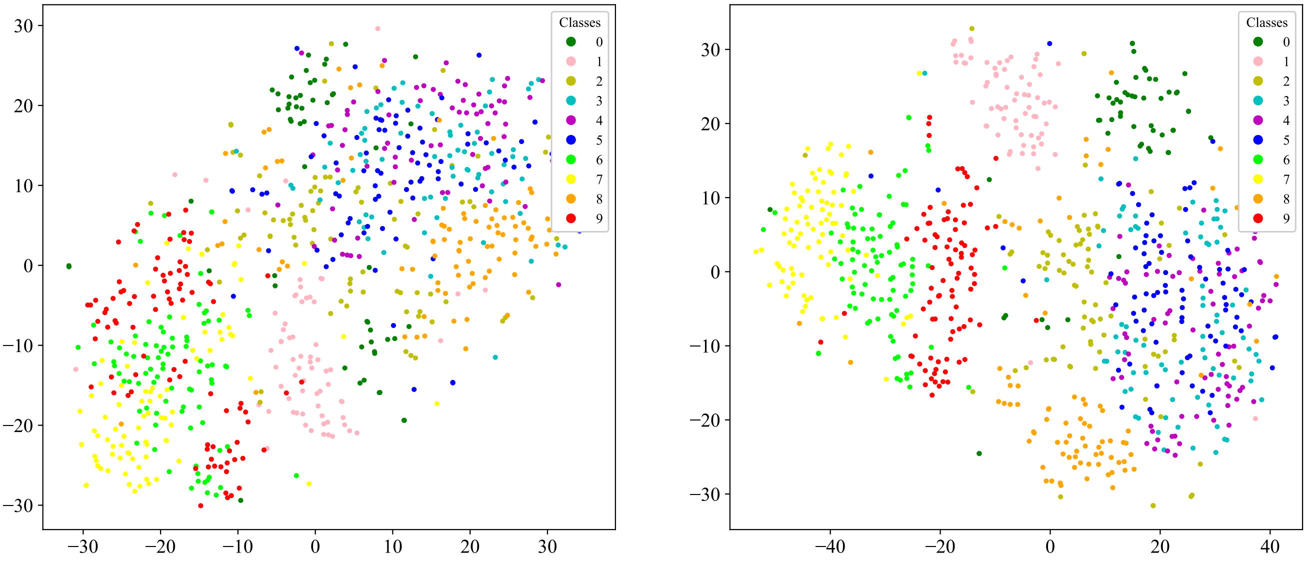
In this paper, we introduce the MPUDML method, which employs a manifold-similarity-based balanced sampling strategy to construct more balanced mini-batches. Additionally, soft supervision derived from manifold similarity and cosine similarity between unlabeled images is used for distinguishing samples, effectively reducing the impact of noisy samples. Simultaneously, image patch-level clustering and localization tasks utilize the rich feature information between patches within an image to guide the acquisition of more comprehensive feature embeddings, thus enhancing retrieval performance. Extensive image retrieval and clustering experiments demonstrate that our method achieves state-of-the-art performance on two challenging datasets. In future work, we will consider experimenting with other feature extraction networks as backbones and attempt to introduce hierarchical similarity that is more suitable for representing sample relationships for measurement to enhance model performance. In real-life scenarios, while manually annotating a large amount of data is expensive and inefficient, annotating a small amount of data is feasible. Leveraging limited label information to guide the model in learning useful features



(a) Initial data embedding

(b) Optimized data embedding

Fig. 10 Distribution graph of data embedding representations for 10 random categories on the CUB-200-2011 dataset



(a) Initial data embedding

(b) Optimized data embedding

Fig. 11 Distribution graph of data embedding representations for 10 random categories on the Cars196 dataset

from a large amount of unlabeled data brings significant performance improvements. From a practical perspective, we will also consider further modifying our model and utilizing limited data annotation for semi-supervised metric learning tasks. Additionally, we will consider using the Stanford Online Product dataset, which contains a large number of product images, for experimentation to further enhance the applicability of our model.

Acknowledgements This work is supported by the National Natural Science Foundation of China under Grant No. 62262006, the Guangxi Science and Technology Key Research and Development Program under Grant No. AB24010112, the Key Project of Science and Technology Research Program of Chongqing Education Commission of China under Grant No. KJZD-K202402501, the State Key Laboratory of Geo-Information Engineering and the Key Laboratory of Surveying and Mapping Science and Geospatial Information Technology of MNR, CASM under Grant No. 2023-04-03, and Heavy Rainfall Research Foundation of China under Grant No. BYKJ2024Z12.

Author Contributions Shi-hao Yuan designed the research plan and conducted key experimental validations. Yong Feng (co-corresponding author) provided critical academic guidance and participated in the optimization of the methodology. A-Gen Qiu (co-corresponding author) was responsible for the experimental components and made the final modifications to the methods. Guo-fan Duan and Ming-liang Zhou conducted the data analysis and wrote the initial draft. Bao-hua Qiang and Yong-heng Wang reviewed and revised the entire manuscript. All authors participated in the writing and revision of the manuscript and approved the final version of the content.

Data availability and access The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

Declarations

Competing interests The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Ethical and informed consent for data used Not applicable.

References

1. Lu J, Hu J, Jie Z (2017) Deep metric learning for visual understanding: an overview of recent advances. IEEE Signal Process Mag 34(6):76–84
2. Qayyum A, Anwar SM, Awais M, Majid M (2017) Medical image retrieval using deep convolutional neural network. Neurocomputing 266:8–20
3. De Divitiis L, Becattini F, Baecchi C, Del Bimbo A (2023) Disentangling features for fashion recommendation. ACM Trans Multimed Comput Commun Appl 19(1s):1–21
4. Ji Z, Yao W, Pi H, Wei L, He J, Wang H (2017) A survey of personalised image retrieval and recommendation. In: Theoretical computer science: 35th national conference, NCTCS 2017, Wuhan, China, October 14-15, 2017, Proceedings, Springer, pp 233–247
5. Karnila S, Irianto S, Kurniawan R (2019) Face recognition using content based image retrieval for intelligent security. Int J Advan Eng Res Sci 6(1):91–98
6. Kim S, Kim D, Cho M, Kwak S (2022) Self-taught metric learning without labels. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 7431–7441
7. Yan J, Luo L, Deng C, Huang H (2021) Unsupervised hyperbolic metric learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 12465–12474

8. Zhang L, Zhang M, Song R, Zhao Z, Li X (2023) Unsupervised embedding learning with mutual-information graph convolutional networks. IEEE Trans Multimedia 25:5916–5926
9. Roth K, Milbich T, Sinha S, Gupta P, Ommer B, Cohen JP (2020) Revisiting training strategies and generalization performance in deep metric learning. In: International conference on machine learning, PMLR, pp 8242–8252
10. Liu Y, Guo Y, Zhu Y, Ming Y (2022) Mining semantic information from intra-image and cross-image for few-shot segmentation. Multimed Tool Appl 81(13):18305–18326
11. Zhang R, Isola P, Efros AA (2016) Colorful image colorization. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14, Springer, pp 649–666
12. Mirzasoleiman B, Bilmes J, Leskovec J (2019) Coresets for accelerating incremental gradient methods
13. Johnson TB, Guestrin C (2018) Training deep models faster with robust, approximate importance sampling. Advan Neural Inform Process Syst 31
14. Sinha S, Zhang H, Goyal A, Bengio Y, Larochelle H, Odena A (2020) Small-gan: speeding up gan training using core-sets. In: International conference on machine learning, PMLR, pp 9005–9015
15. Bucher M, Herbin S, Jurie F (2016) Hard negative mining for metric learning based zero-shot classification. In: Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III 14, Springer, pp 524–531
16. Harwood B, Vijay KBG, Carneiro G, Reid I, Drummond T (2017) Smart mining for deep metric learning. In: Proceedings of the IEEE international conference on computer vision, pp 2821–2829
17. Chao-Yuan W, Manmatha R, Smola AJ, Krahenbuhl P (2017) Sampling matters in deep embedding learning. In: Proceedings of the IEEE international conference on computer vision, pp 2840–2848
18. Zhang C, Wan Y, Qiang H (2024) Deep noise mitigation and semantic reconstruction hashing for unsupervised cross-modal retrieval. Neural Comput Appl:1–15
19. Hadsell R, Chopra S, LeCun Y (2006) Dimensionality reduction by learning an invariant mapping. In: 2006 IEEE computer society conference on Computer Vision and Pattern Recognition (CVPR'06), IEEE, vol 2, pp 1735–1742
20. Schroff F, Kalenichenko D, Philbin J (2015) Facenet: a unified embedding for face recognition and clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 815–823
21. Sohn K (2016) Improved deep metric learning with multi-class n-pair loss objective. Advan Neural Inform Process Syst 29
22. Song HO, Xiang Y, Jegelka S, Savarese S (2016) Deep metric learning via lifted structured feature embedding. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4004–4012
23. Wang X, Hua Y, Kodirov E, Guosheng H, Garnier R, Robertson NM (2019) Ranked list loss for deep metric learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 5207–5216
24. Wang J, Zhou F, Wen S, Liu X, Lin Y (2017) Deep metric learning with angular loss. In: Proceedings of the IEEE international conference on computer vision, pp 2593–2601
25. Ge W (2018) Deep metric learning with hierarchical triplet loss. In: Proceedings of the European Conference on Computer Vision (ECCV), pp 269–285
26. Kim W, Goyal B, Chawla K, Lee J, Kwon K (2018) Attention-based ensemble for deep metric learning. In: Proceedings of the European Conference on Computer Vision (ECCV), pp 736–751
27. Zheng W, Chen Z, Jiwen L, Zhou J (2019) Hardness-aware deep metric learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 72–81
28. Wang X, Han X, Huang W, Dong D, Scott MR (2019) Multi-similarity loss with general pair weighting for deep metric learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 5022–5030
29. Alexey D, Fischer P, Tobias J, Springenberg MR, Brox T (2016) Discriminative unsupervised feature learning with exemplar convolutional neural networks. IEEE TPAMI 38(9):1734–1747
30. Li Y, Kan S, He Z (2020) Unsupervised deep metric learning with transformed attention consistency and contrastive clustering loss. In: European conference on computer vision, Springer, pp 141–157
31. Mang YX, Zhang PC, Yuen, Shih-Fu C, (2019) Unsupervised embedding learning via invariant and spreading instance feature. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 6210–6219
32. Cao X, Chen B-C, Lim S-N (2019) Unsupervised deep metric learning via auxiliary rotation loss. arXiv:1911.07072
33. Zhang L, Qi G-J, Wang L, Luo J (2019) Aet vs. aed: unsupervised representation learning by auto-encoding transformations rather than data. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 2547–2555
34. Zhirong W, Xiong Y, Yu SX, Lin D (2018) Unsupervised feature learning via non-parametric instance discrimination. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3733–3742
35. Huang Jiabo, Dong Qi, Gong Shaogang, Zhu Xiatian (2019) Unsupervised deep learning by neighbourhood discovery. In: International conference on machine learning, PMLR, pp 2849–2858
36. Ye M, Jianbing S, Zhang X, Yuen PC, Shih-Fu C (2020) Augmentation invariant and instance spreading feature for softmax embedding. IEEE Trans on Pattern Anal Mach Intell 44(2):924–939
37. Dutta UK, Harandi M, Sekhar CC (2020) Unsupervised deep metric learning via orthogonality based probabilistic loss. IEEE Trans Artif Intell 1(1):74–84
38. Iscen A, Tolias G, Avrithis Y, Chum O (2018) Mining on manifolds: metric learning without labels. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 7642–7651
39. Zhou D, Weston J, Gretton A, Bousquet O, Schölkopf B (2003) Ranking on data manifolds. Advan Neural Inform Process Syst 16
40. Zhou D, Bousquet O, Lal T, Weston J, Schölkopf B (2003) Learning with local and global consistency. Advan Neural Inform Process Syst 16
41. Kim S, Kim D, Cho M, Kwak S (2021) Embedding transfer with label relaxation for improved metric learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 3967–3976
42. Chen P, Liu S, Jia J (2021) Jigsaw clustering for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 11526–11535
43. Wah C, Branson S, Welinder P, Perona P, Belongie S (2011) The caltech-ucsd birds-200-2011 dataset
44. Krause J, Stark M, Deng J, Fei-Fei L (2013) 3d object representations for fine-grained categorization. In: Proceedings of the IEEE international conference on computer vision workshops, pp 554–561
45. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1–9

46. Caron M, Bojanowski P, Joulin A, Douze M (2018) Deep clustering for unsupervised learning of visual features. In: Proceedings of the European Conference on Computer Vision (ECCV), pp 132–149
47. Zhou J, Tang Y, Bing S, Ying W (2021) Unsupervised embedding learning from uncertainty momentum modeling. arXiv:2107.08892
48. Ye M, Jianbing S, Zhang X, Yuen PC, Shih-Fu C (2022) Augmentation invariant and instance spreading feature for softmax embedding. IEEE Trans Pattern Anal Mach Intell 44(2):924–939
49. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
50. Deng J, Dong W, Socher R Li L-J, Li K, Fei-Fei L (2009) Imagenet: a large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition, IEEE, pp 248–255
51. Song HO, Jegelka S, Rathod V, Murphy K (2017) Deep metric learning via facility location. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5382–5390
52. Michael O, Georg W, Horst P, Horst B (2018) Deep metric learning with bier: boosting independent embeddings robustly. IEEE TPAMI 42(2):276–290

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

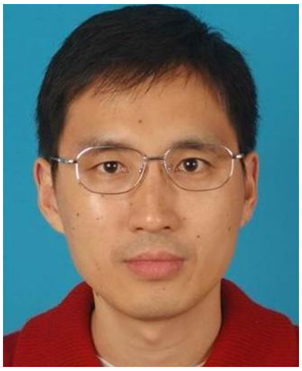
Shi-hao Yuan received his B.Eng. degree in Internet of Things Engineering from the School of Marine Electrical Engineering at Dalian Maritime University, Dalian, Liaoning, China, in 2022. Currently he is studying for a master's degree in Computer Technology at Chongqing University, Chongqing, China. His current research interests include fine-grained image retrieval and unsupervised deep metric learning.

Yong Feng received his Ph.D. degree in computer science and technology from College of Computer Science at Chongqing University, Chongqing, China, in 2006. Currently he is a Professor at the College of Computer Science, Chongqing University. His research interest covers Big Data Analysis and Data Mining, Artificial Intelligence and Big Data Processing, Deep Learning and Big Data Retrieval. One of the corresponding authors of this paper.

A-Gen Qiu received his Ph.D. degree in cartography and geographic information engineering from Wuhan University, Wuhan, China, in 2017. He has been engaged in spatially assisted decision making, spatial-temporal big data analysis and mining. He is responsible for the research and development of Geowindows spatial-temporal big data platform and natural resources big data supercomputing cloud platform. One of the corresponding authors of this paper.

Guo-fan Duan received his Master degree in Civil Engineering from College of Civil Engineering at Chongqing University, Chongqing, China, in 2014. Currently he is a associate professor at Chongqing Metropolitan College of Science and Technology. His research interest covers Big Data Analysis, Image processing and Artificial Intelligence.

Ming-liang Zhou received his Ph.D. degree in computer science from Beihang University, Beijing, China, in 2017. He was a Postdoctoral Researcher with the Department of Computer Science, City University of Hong Kong, Hong Kong, China, from September 2017 to September 2019. He was a Postdoctoral Fellow with the State Key Lab of Internet of Things for Smart City, University of Macau, Macau, China, from October 2019 to October 2021. He is currently an Associate Professor with the School of Computer Science, Chongqing University, Chongqing, China. His research interests include image and video coding, perceptual image processing, multimedia signal processing, rate control, multimedia communication, machine learning, and optimization.



Bao-hua Qiang received his Ph.D. degree in Department of Computer Science from Chongqing University, Chongqing, China, in 2005. He is now a Professor at the Guangxi Cooperative Innovation Center of cloud computing and Big Data, Guilin University of Electronic Technology. His research interest is Big Data Processing and Information Retrieval.

Yong-heng Wang received his Ph.D. degree in computer science and technology from National University of Defense Technology, Changsha, China, in 2006. Currently he is a research specialist at the research center of Big data intelligence, Zhejiang Lab. His research interest covers Big data analysis, machine learning, computer simulation and intelligent decision making.

Authors and Affiliations

Shi-hao Yuan[1] · **Yong Feng**[1] · **A-Gen Qiu**[2] · **Guo-fan Duan**[3] · **Ming-liang Zhou**[1] · **Bao-hua Qiang**[4] · **Yong-heng Wang**[5]

✉ Yong Feng
fengyong@cqu.edu.cn

✉ A-Gen Qiu
qiuag@casm.ac.cn

Shi-hao Yuan
shyuancqu@163.com

Guo-fan Duan
wlm_bkesz@foxmail.com

Ming-liang Zhou
zml-0913yy@163.com

Bao-hua Qiang
qiangbh@guet.edu.cn

Yong-heng Wang
wangyh@zhejianglab.com

[1] College of Computer Science, Chongqing University, Chongqing 401331, China, and Heavy Rainfall Research Center of China, No.3, Donghu East Road, Hongshan District, Wuhan 401311, China

[2] State Key Laboratory of Geo-Information Engineering and Key Laboratory of Surveying and Mapping Science and Geospatial Information Technology of MNR, CASM, Beijing 100036, China

[3] Chongqing Metropolitan College of Science and Technology, Chongqing 402167, China

[4] Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology, Guilin 541004, China

[5] 8# of Zhejiang Lab,, Yuhang district Hangzhou 311121, China