Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI-19)

Zero-shot Metric Learning

Xinyi Xu , **Huanhuan Cao** , **Yanhua Yang** , **Erkun Yang** and **Cheng Deng***
School of Electronic Engineering, Xidian University, Xian 710071, China
xyxu.xd@gmail.com, hhcao@stu.xidian.edu.cn, yanhyang@xidian.edu.cn,
{erkunyang, chdeng.xd}@gmail.com

Abstract

In this work, we tackle the zero-shot metric learning problem and propose a novel method abbreviated as ZSML, with the purpose to learn a distance metric that measures the similarity of unseen categories (even unseen datasets). ZSML achieves strong transferability by capturing multi-nonlinear yet continuous relation among data. It is motivated by two facts: 1) relations can be essentially described from various perspectives; and 2) traditional binary supervision is insufficient to represent continuous visual similarity. Specifically, we first reformulate a collection of specific-shaped convolutional kernels to combine data pairs and generate multiple relation vectors. Furthermore, we design a new cross-update regression loss to discover continuous similarity. Extensive experiments including intra-dataset transfer and inter-dataset transfer on four benchmark datasets demonstrate that ZSML can achieve state-of-the-art performance.

1 Introduction

Metric learning aims to find appropriate similarity measurements of data points, whose core intuition is to preserve the distance between data points in embedding space. This topic is of important practice due to its wide applications in many related areas, such as face recognition [Guillaumin *et al.*, 2009], clustering [Davis *et al.*, 2007; Xing *et al.*, 2003], and retrieval [Zhou *et al.*, 2004].

Euclidean distance is one of the most common similarity metrics since it does not require priori information and training process. However, unsatisfactory results may be yielded as it treats all feature dimensions equally and independently, thus fails to capture the idiosyncrasies of data. In contrast, parametric Mahalanobis distance that can model the different dimension importance, has been adopted in many works. Some representative Mahalanobis approaches [Hoi *et al.*, 2006; Xing *et al.*, 2003] project data linearly and minimize Euclidean distance between positive pairs, while maximize it between negative pairs. Alternatively, one may also directly optimize the Mahalanobis metric for nearest neighbor classification, among which representative works include, but are not limited to, Neighborhood Component Analysis (NCA) [Roweis *et al.*, 2004], Large Margin Nearest Neighbor (LMNN) [Weinberger and Saul, 2009], and Nearest Class Mean (NCM) [Mensink *et al.*, 2013]. Priori information plays a pivotal role in the success of these metric learning schemes. Therefore, unsatisfactory results can be produced when the priori is not available.

In this paper, we are committed to a more challenging task: zero-shot metric learning, whose ambition is to learn an effective metric for unseen categories and datasets. It claims that the learned metric must measure the similarity without access to the target data. Powerful transferability can be obtained by capturing the multi-nonlinear and continuous relations, which is consistent with the innate character of data. Particularly, we first reformulates a set of specific-shaped convolutional kernels to discover various kinds of relations. It is well known that convolutional neural network (CNN) has great power in feature embedding[Lecun *et al.*, 1998; Donahue *et al.*, 2013; Toshev and Szegedy, 2014], while in this paper it is employed to reveal the correlation among data. Then, we design a cross-update regression loss, which relax the binary supervision employed on the positive pairs (PPs) and negative pairs (NPs) to extend generalization capability. Specifically, we initialize a coarse continuous label as a weak supervision of the predicted similarity, and update the coarse label and the predicted similarity alternately till convergence. By doing so, we can learn the similarity order and improve transferability. To better demonstrate the superiority of ZSML, we present multi-level transfer tasks, which covers transferring to unseen category within one dataset (intra-dataset ZSML) and unseen datasets (inter-dataset ZSML). In a nutshell, the main contributions of our work can be summarized as follows:

- Departing from the traditional single and linear relation representation, we reformulate a family of specific-shaped convolutional kernels which can capture the multi-nonlinear relations among data points.
- We devise a cross-update regression loss for learning continuous similarity to improve generalization capability, which is verified in our empirical study.
- Extensive transfer experiments demonstrate that our model can better measure the similarity of unseen cate-

*Corresponding author.

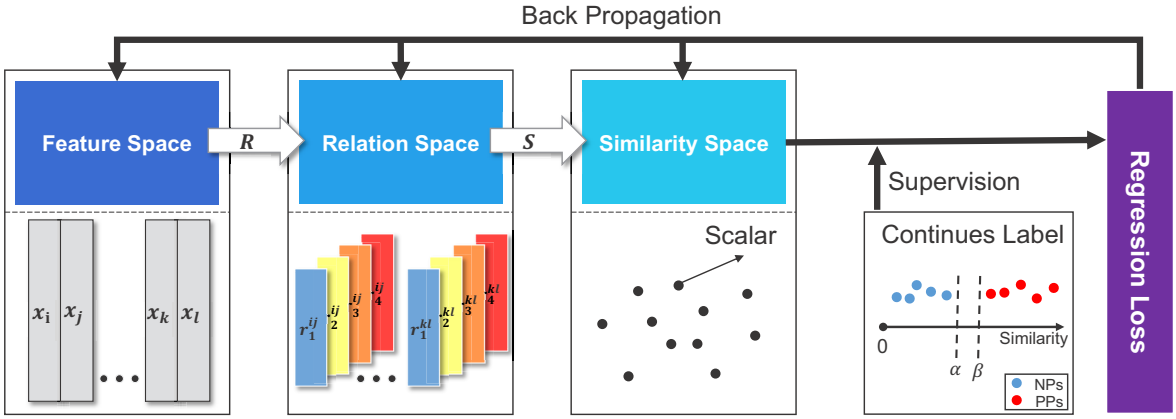

Figure 1: The multi-nonlinear regression metric learning framework of our proposed method. ZSML employs a relation function $\mathbf{R}$ and a similarity function $\mathbf{S}$ to project data from the feature space into a scalar similarity space, where the similarity extent of two examples are measured. Finally, a regression from scalars in the similarity space to the continuous labels is adopted to guide the training process. The continuous labels are ranged in $(0, \alpha)$ for NPs and $(\beta, 1)$ for PPs.

gories and unseen dataset compared with the peer methods.

2 Related Work

The Mahalanobis metric is among the most commonly used linear metric learning methods, and a majority of methods have been developed based on the Mahalanobis metric. For instance, Davis proposed an information theoretical metric learning (ITML) method [Davis *et al.*, 2007], which essentially minimized the differential relative entropy between two multi-variate Gaussians. LMNN [Weinberger and Saul, 2009] enforced a large margin of a Support Vector Machine (SVM) within the triplet data and depicted the relative relations among three individual examples. Recently, GMML [Zadeh *et al.*, 2016] revisited the task of learning an Euclidean metric from weakly supervised data, where pairs of similar and dissimilar points building on geometric intuition are given. Furthermore, Ye [Ye *et al.*, 2016] proposed a unified multi-metric learning approach (UM^{2}L) to combine both spatial connections and rich semantic factors. Xiong [Xiong *et al.*, 2012] proposed a single adaptive metric termed position dependence structure, which additionally incorporated the feature mean vector to encode the distance besides the feature difference vector. Thereon, Huang [Huang *et al.*, 2016] proposed to encode the two linear structures of data pairs and then map the feature pair to a similarity space.

Zero-shot learning aims for the learning of a task without training samples [Huang *et al.*, 2015; Lampert *et al.*, 2014]. Usually, this involves transferring the knowledge either by the model parameters or by shared features. Numerous models have been proposed to focus on descriptive attributes to represent object classes [Lampert *et al.*, 2014; Farhadi *et al.*, 2010]. Some other models exploit the hierarchical semantics of data [Griffin and Perona, 2008; Marszalek and Schmid, 2007]. The sample space is imposed by a general-to-specific order either based on an existing hierarchy [Marszalek and Schmid, 2007] or learned from visual features [Griffin and Perona, 2008]. Scalability is achieved by associating classifiers with each hierarchy node. In this paper, we focus on an analogous yet different issue: a zero-shot metric. The main purpose of our zero-shot metric learning is to measure the similarity between instances which are never seen before.

3 Proposed Approach

Figure 1 shows the framework of our proposed ZSML. The relation mining function $\mathbf{R}$ is first employed to project data pairs from feature space to relation space, in which each kind of relation is encoded as an vector. We then employ a similarity function $\mathbf{S}$ to map the relation vector into a scalar similarity space, where each scalar implies how similar two data points is. Finally a regression loss guides the whole optimization procedure under the supervision of a continuous label.

3.1 Preliminaries

Let $\chi = \{\mathbf{x}_i | i = 1, 2, \cdots, n\}$ be the training set, where $\mathbf{x}_i$ is an m-dimensional vector. $\mathbf{P}$ is a data pair set, which contains N pairs randomly built up within χ. Given a random data pair $(\mathbf{x}_i, \mathbf{x}_j)$, $(\mathbf{r}_1^{ij}, \mathbf{r}_2^{ij}, \cdots, \mathbf{r}_k^{ij})$ indicate the corresponding k relation vectors produced by a relation function $\mathbf{R}$, and $s_{ij} \in \mathbf{s}$ is the predicted similarity generated by a similarity function $\mathbf{S}$. To achieve continuous supervision, we encode available binary label information $\boldsymbol{y}^b \in R^N$ into continuous form $\boldsymbol{y}^c \in R^N$. For PPs (NPs), the binary label $y^b = 1$ ($y^b = 0$) while the continuous label $y^c \in (\beta, 1)$ ($y^c \in (0, \alpha)$), where α and β are two boundaries for NPs and PPs respectively.

3.2 Multi-Nonlinear Relations Mining

Traditional Mahalanobis metric learning algorithms usually employ a linear projection $\mathbf{A}$ to map the original data points as $\mathbf{Ax}$, and compute a simple Euclidean distance $\|\mathbf{Ax}_i - \mathbf{Ax}_j\|^2$ to imply the similarity extent of data pair. However, it fails to describe inherently complex relations among data, and we tackle this problem by adopting a family of specific-shaped convolutional kernels to project data pair from feature

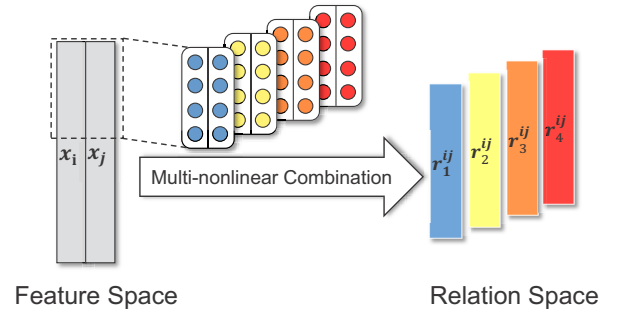

Figure 2: The multi-nonlinear relations mined by specific-shaped convolutional kernels, whose width are all 2. The purpose is to combine data pair and generate multiple relation vectors. This figure shows an example when $k = 4$.

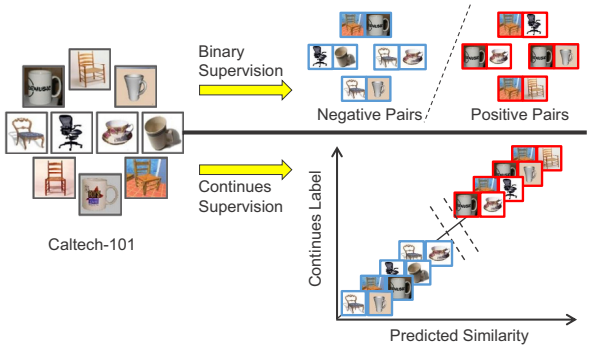

Figure 3: Comparison between binary supervision and continuous supervision. The blue edged pairs are negative pairs (NPs) while the red-edged ones are positive pairs (PPs). Binary supervision only separates NPs and PPs which ignores the order of similarity, while continuous supervision shows great priority as it can reveal continuous visual similarity.

space into relation space. As illustrated in Figure 2, four different convolutional kernels, whose width are all 2, slide on the $m \times 2$ feature matrix vertically, and thus produce four various relation vectors (distinguished by color). By doing this, we can unearth multiple relations among data. Taking a data pair $(\mathbf{x}_j, \mathbf{x}_j)$ as input, k relation vectors are computed by

$$(\mathbf{r}_1^{ij}, \mathbf{r}_2^{ij}, \cdots, \mathbf{r}_k^{ij}) = \mathbf{R}(\mathbf{x}_i,\ \mathbf{x}_j), \quad \mathbf{R} \rightarrow \{\mathbf{W}_1^C,\ \mathbf{b}_1^C\}, \quad (1)$$

where $(\mathbf{r}_1^{ij}, \mathbf{r}_2^{ij}, \cdots, \mathbf{r}_k^{ij})$ are k relation vectors, $\mathbf{R}$ is a relation function implemented by a convolutional layer (Conv) and a rectified linear unit layer (ReLU), and $\left(\mathbf{W}_1^C, \mathbf{b}_1^C\right)$ are the parameters of $\mathbf{R}$.

We employ a similarity function $\mathbf{S}$ to project the above relation vectors into a one-dimensional similarity space, in which the larger value indicates the more similar is. $\mathbf{S}$ is implemented by three fully connected (FC) layers, which the last layer contains one neuron. Therein, the predicted similarity s_{ij} of data pair $(\mathbf{x}_i,\ \mathbf{x}_j)$ is computed by

$$\begin{aligned} s_{ij} &= \mathbf{S}(\mathbf{r}_1^{ij}, \mathbf{r}_2^{ij}, \cdots, \mathbf{r}_k^{ij}), \\ \mathbf{S} &\rightarrow \{\mathbf{W}_1^I, \mathbf{b}_1^I; \mathbf{W}_2^I, \mathbf{b}_2^I; \mathbf{W}_3^I, \mathbf{b}_3^I\}, \end{aligned} \quad (2)$$

where $\{\mathbf{W}_1^I, \mathbf{b}_1^I; \mathbf{W}_2^I, \mathbf{b}_2^I; \mathbf{W}_3^I, \mathbf{b}_3^I\}$ are the parameters of the three fully connected layers. All projection parameters are $\mathbf{V} = \{\mathbf{W}^C, \mathbf{b}^C, \mathbf{W}^I, \mathbf{b}^I\}$. Notably, the neural network serves as a tool for correlation uncovering, which is quite different from the traditional feature extracting.

3.3 Regression Loss

Conventional metric learning algorithms usually adopt a binary label as supervision, which is prone to be over-fitting for trying to individually push the similarity in terms of two single points [Huang *et al.*, 2016]. As depicted in Figure 3, binary labels consider data points in the same class with the same similarity extent, and neglect the intra-class data manifold. To better preserve the original data similarities, we propose a regression loss to learn continuous similarity, which enables data points from the same category to reside on a manifold, and also maintain a distance between data points from different categories. The regression loss is implemented in two steps: 1) generate an N-dimensional vector according to binary label $\boldsymbol{y}^b$, which serves as an initialization of continuous label $\boldsymbol{y}^c$; 2) a cross-update strategy is employed to alternately optimize predicted similarity $\boldsymbol{s}$ and $\boldsymbol{y}^c$ by forcing the consistency of them.

The binary essence of $\boldsymbol{y}^b$ is a obstacle towards guiding continuous similarity learning. Therein, we encode $\boldsymbol{y}^b$ into a continuous form $\boldsymbol{y}^c$, which contributes in a certain range. As shown in Figure 4, the Euclidean distances are mapped into $(\beta, 1)$ for PPs and $(0, \alpha)$ for NPs. Concretely, we adopt the following mapping functions:

$$\begin{aligned} y_{ij}^c &= -\alpha d_{ij}^2 + \alpha, \ \ if \ \ y_{ij}^b = 0, \\ y_{ij}^c &= (\beta - 1) d_{ij}^2 + 1, \ \ if \ \ y_{ij}^b = 1, \end{aligned} \quad (3)$$

where d_{ij} is the normalized Euclidean distance, α and β are the boundaries for the continuous labels.

Another problem is that the initialized continuous label is coarse and needs to be finely tuned. ZSML achieves this goal by adopting a cross-update strategy to optimize the predicted similarity $\mathbf{s}$ and continuous labels $\boldsymbol{y}^c$ alternatively. We design our objective function to consist of two parts, and the intuition is to make $\mathbf{s}$ and $\boldsymbol{y}^c$ close:

- The loss of the similarity $\mathcal{L}_{\mathbf{s}}$ is

$$\mathcal{L}_{\mathbf{s}}(\mathbf{s}, \mathbf{V}) = \frac{1}{N} \sum_{(\mathbf{x}_i,\ \mathbf{x}_j) \in \mathbf{P}} \left(s_{ij} - y_{ij}^c\right)^2, \quad (4)$$
$$s.t.\ \beta < s_{ij} < 1, \ \ if \ \ y_{ij}^b = 1; \ \ 0 < s_{ij} < \alpha, \ \ if \ \ y_{ij}^b = 0.$$

As similarity $\mathbf{s}$ is obtained by our model parameterized by $\mathbf{V}$, the target optimization variables of Equation (4) are $\mathbf{s}$ and $\mathbf{V}$ when $\boldsymbol{y}^c$ is fixed.

- The loss of the continuous labels $\mathcal{L}_{\boldsymbol{y}^c}$ is

$$\mathcal{L}_{\boldsymbol{y}^c}(\boldsymbol{y}^c) = \frac{1}{N} \sum_{(\mathbf{x}_i,\ \mathbf{x}_j) \in \mathbf{P}} \left(y_{ij}^c - s_{ij}\right)^2, \quad (5)$$
$$s.t.\ \beta < y_{ij}^c < 1, \ \ if \ \ y_{ij}^b = 1; \ \ 0 < y_{ij}^c < \alpha, \ \ if \ \ y_{ij}^b = 0,$$

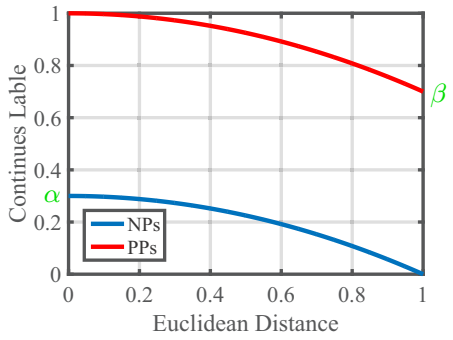

Figure 4: Continuous label generation. We initialize a continuous label $\boldsymbol{y}^c$ into a certain distribution: $\boldsymbol{y}^c \in (0, \alpha)$ when $\boldsymbol{y}^b = 0$ (the blue curve); $\boldsymbol{y}^c \in (\beta, 1)$ when $\boldsymbol{y}^b = 1$ (the red curve).

Combining $\mathcal{L}_{\mathbf{s}}$ and $\mathcal{L}_{\boldsymbol{y}^c}$, we can get the overall objective function as:

$$\mathcal{L} = \mathcal{L}_{\mathbf{s}} + \lambda \mathcal{L}_{\boldsymbol{y}^c} = \frac{1}{N} \sum_{(\mathbf{x}_i,\, \mathbf{x}_j) \in \mathbf{P}} (1+\lambda)(s_{ij} - y_{ij}^c)^2,$$
$$s.t. \beta < y_{ij}^c,\ s_{ij} < 1,\ if\ y_{ij}^b = 1;\ \ 0 < y_{ij}^c,\ s_{ij} < \alpha,\ if\ y_{ij}^b = 0, \tag{6}$$

where λ is the hyper-parameter to balance $\mathcal{L}_{\mathbf{s}}$ and $\mathcal{L}_{\boldsymbol{y}^c}$. $\boldsymbol{y}^c$ is the initialized continuous label according to the binary label, which meets the two constraints: the values range in $(\beta, 1)$ for positive pairs and $(0, \alpha)$ for negative pairs. We only finely tune $\boldsymbol{y}^c$ and thus λ is set to a small value (0.1 for our experiments). The continuous supervised regression loss can relax the binary supervision employed on PPs and NPs and then extend generalization capability efficiently.

4 Optimization

To optimize this model, we first apply a hinge-loss function to transform the constrained optimization problem to an unconstrained one [Hadsell *et al.*, 2006]. For the similarity $\mathbf{s}$ in Equation (4), we have

$$\min_{\mathbf{s},\mathbf{V}} \frac{1}{N} \Bigg(\sum_{(\mathbf{x}_i,\, \mathbf{x}_j) \in \mathbf{P}} (s_{ij} - y_{ij}^c)^2 + \sum_{(\mathbf{x}_i,\, \mathbf{x}_j) \in \mathbf{P}, y_{ij}^b = 1} \Big([s_{ij} - 1]_+^2 + [\beta - s_{ij}]_+^2 \Big) + \sum_{(\mathbf{x}_i,\, \mathbf{x}_j) \in \mathbf{P}, y_{ij}^b = 0} \Big([s_{ij} - \alpha]_+^2 + [-s_{ij}]_+^2 \Big) \Bigg), \tag{7}$$

where the operator $[\cdot]_+$ indicates the hinge function $\max(0, \cdot)$. For the continuous label $\boldsymbol{y}^c$ in Equation (5), we have

$$\min_{\boldsymbol{y}^c} \frac{1}{N} \Bigg(\sum_{(\mathbf{x}_i,\, \mathbf{x}_j) \in \mathbf{P}} (y_{ij}^c - s_{ij})^2 + \sum_{(\mathbf{x}_i,\, \mathbf{x}_j) \in \mathbf{P}, y_{ij}^b = 1} \Big([y_{ij}^c - 1]_+^2 + [\beta - y_{ij}^c]_+^2 \Big) + \sum_{(\mathbf{x}_i,\, \mathbf{x}_j) \in \mathbf{P}, y_{ij}^b = 0} \Big([y_{ij}^c - \alpha]_+^2 + [-y_{ij}^c]_+^2 \Big) \Bigg). \tag{8}$$

Algorithm 1 Learning of the proposed ZSML model.

Input: Training set $\{\mathbf{x}_i\}$. Initialized parameters $\{\mathbf{W}^C, \mathbf{b}^C\}$ of the convolution layer, $\{\mathbf{W}^I, \mathbf{b}^I\}$ of the fully connected layers, and $\boldsymbol{y}^c = \{y_{ij}^c | (\mathbf{x}_i,\ \mathbf{x}_j) \in \mathbf{P}\}$ of the loss layer. hyper-parameters λ, α, β and learning rate μ^t. The number of iterations $t \leftarrow 0$.
Output: The parameters $\mathbf{V} = \{\mathbf{W}^C, \mathbf{b}^C, \mathbf{W}^I, \mathbf{b}^I\}$.
1: **while** not converge **do**
2: $t \leftarrow t + 1$.
3: Compute the joint loss by $\mathcal{L}^t = \mathcal{L}_{\mathbf{s}}^t + \lambda \mathcal{L}_{\boldsymbol{y}^c}^t$.
4: Fix $\mathbf{s}$. Update $\boldsymbol{y}^c$ by $\boldsymbol{y}^{t+1} = \boldsymbol{y}^t - \mu^t \cdot \lambda \cdot \frac{\partial \mathcal{L}_{\boldsymbol{y}}^t}{\partial \boldsymbol{y}^t}$. Record $\boldsymbol{y}^c$ as $\boldsymbol{y}$ for the conflict of superscripts c and t.
5: Fix $\boldsymbol{y}$. Compute backpropagation error $\frac{\partial \mathcal{L}_{\mathbf{s}}^t}{\partial \mathbf{s}^t}$ for all the data points.
6: Update $\mathbf{V}$ by $\mathbf{V}^{t+1} = \mathbf{V}^t - \mu^t \cdot \frac{\partial \mathcal{L}_{\mathbf{s}}^t}{\partial \mathbf{s}^t} \cdot \frac{\partial \mathbf{s}^t}{\partial \mathbf{V}^t}$.
7: **end while**

We employ a cross-optimization strategy, fixing $\mathbf{s}$ when optimize $\boldsymbol{y}^c$ and vice versa. The gradients of the loss function with regard to $\mathbf{s}$, $\mathbf{V}$, and $\boldsymbol{y}^c$ are computed by Equation (9) and (10) as

$$\frac{\partial \mathcal{L}_{\mathbf{s}}}{\partial \mathbf{s}} = \frac{1}{2N} \Bigg(\sum_{(\mathbf{x}_i,\, \mathbf{x}_j) \in \mathbf{P}} (s_{ij} - y_{ij}^c) + \sum_{(\mathbf{x}_i,\, \mathbf{x}_j) \in \mathbf{P}, y_{ij}^b = 1} \Big([s_{ij} - 1]_+ + [\beta - s_{ij}]_+ \Big) + \sum_{(\mathbf{x}_i,\, \mathbf{x}_j) \in \mathbf{P}, y_{ij}^b = 0} \Big([s_{ij} - \alpha]_+ + [-s_{ij}]_+ \Big) \Bigg),$$
$$\frac{\partial \mathcal{L}_{\mathbf{s}}}{\partial \mathbf{V}} = \frac{\partial \mathcal{L}_{\mathbf{s}}}{\partial \mathbf{s}} \cdot \frac{\partial \mathbf{s}}{\partial \mathbf{V}}, \tag{9}$$

$$\frac{\partial \mathcal{L}_{\boldsymbol{y}^c}}{\partial \boldsymbol{y}^c} = \frac{1}{2N} \Bigg(\sum_{(\mathbf{x}_i,\, \mathbf{x}_j) \in \mathbf{P}} (y_{ij}^c - s_{ij}) + \sum_{(\mathbf{x}_i,\, \mathbf{x}_j) \in \mathbf{P}, y_{ij}^b = 1} \Big([y_{ij}^c - 1]_+ + [\beta - y_{ij}^c]_+ \Big) + \sum_{(\mathbf{x}_i,\, \mathbf{x}_j) \in \mathbf{P}, y_{ij}^b = 0} \Big([y_{ij}^c - \alpha]_+ + [-y_{ij}^c]_+ \Big) \Bigg). \tag{10}$$

We adopt a quadratic form of the hinge loss to transform the objective functions (4) and (5) into an unconstrained problem. It can derive penalty from the original loss with back propagation, referring to Equation (9) and (10). That is the further violation of constraints is, the larger gradient is. The procedure is depicted in Algorithm 1.

5 Experiments

In this section, we evaluate the efficiency of our proposed ZSML by multi-level transfer setting on four public benchmark datasets, transferring to unseen categories (intra-dataset ZSML) and unseen datasets (inter-dataset ZSML). We evaluate the clustering performance of all experiments, by three metrics: Accuracy (ACC), Normalized Mutual Information (NMI), and Purity [Cai *et al.*, 2008].

Type	Neurons/ stride	Output size	FLOPS
Conv	2×8 / 10	$96 \times 261 \times 1$	400K
Norm, ReLU		$96 \times 261 \times 1$	
FC	$1,000 \times 1$	$1,000 \times 1$	25M
ReLU		$1,000 \times 1$	
FC	$1,000 \times 1$	$1,000 \times 1$	1M
FC	1×1	1×1	0.1K
Total			26.4M

Table 1: The network architecture of our model.

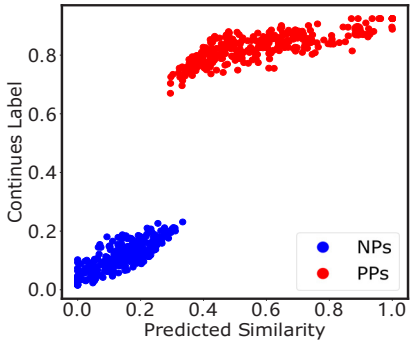

Figure 5: The distribution of predicted similarity and continuous label after training $20,000$ times on Caltech.

5.1 Implementation Details

The Caffe package [Jia *et al.*, 2014] is used throughout the experiments. The base learning rate is set to 0.002 and iteration times are set to $20,000$. It takes about 6 minutes for training on an NVIDIA TITAN X GPU, since our model only requires 26.4 million FLOPS, as summarized in Table 1. There are four hyper-parameters λ, k, α, and β and we set them to 0.1, 96, 0.3, and 0.7 respectively. β always equals to $1 - \alpha$. As depicted in Figure 5, we randomly pick 400 points in Caltech, and plot the distribution between predicted similarity and continuous label after training $20,000$ times. We can observe that the predicted similarity resides on two smooth manifolds respectively for NPs and PPs.

We conduct our experiments on Caltech-101 [Wah *et al.*, 2011b], COIL-20 [Nene *et al.*, 1996], ImageNet-20[1] [Deng *et al.*, 2009], and MSRC-v1, of which the details are summarized in Table 2. All images are represented by concatenating three popular descriptors, which are histogram of oriented gradient (HOG), GIST, and local binary pattern (LBP) respectively. The dimensions of three descriptors are 1152, 512 and 944 separately and then each data point is represented by a 2608-dimensional vectors. In the following parts, we refer to the four datasets as Caltech, COIL, ImageNet, and MSRC for brief.

We compare our method with five metric learning algorithms, i.e., 1) The Euclidean metric; 2) Information-theoretic metric learning (ITML) [Davis *et al.*, 2007]; 3) Distance metric learning for large margin nearest neighbor classification (LMNN) [Weinberger and Saul, 2009]; 4) Geometric mean metric learning (GMML) [Zadeh *et al.*, 2016]; 5) Closed-form training of Mahalanobis distance for supervised clustering (MLCA) [Law *et al.*, 2016].

[1]We randomly choose 20 categories and 150 images per category from ImageNet, referred to as ImageNet-20.

DATASET	Caltech	COIL	ImageNet	MSRC
#Classes	101	20	17	8
#Images	$8,677$	$1,440$	$2,550$	240
#Feature	HOG $(1,152)$, GIST (512), LBP (944)			

Table 2: Datasets descriptions with the details of the feature types, number of classes, and number of images.

5.2 Result Analysis

In this part, we will demonstrate our quantitative results and give some analysis about the figures. In all of the tables, we underline the best performance, and bold the second best.

In intra-dataset ZSML task, we train our model on some categories and test the learned metric on the other categories which belong to the same dataset. We conduct two group of experiments, including Caltech $\rightarrow$ Caltech and ImageNet $\rightarrow$ ImageNet. For both of them, we randomly pick 7 classes (more than $1,000$ data points) for testing and the remaining for training. Table 3 reports the ACC, NMI and purity metrics, and we can see that our approach outperforms peer methods by a large margin for all the three metrics. In inter-dataset ZSML task, we train our model on source dataset, and measure the similarity of target dataset. We conduct four groups of experiments, which are Caltech $\rightarrow$ COIL, Caltech $\rightarrow$ ImageNet, Caltech $\rightarrow$ MSRC, and ImageNet $\rightarrow$ Caltech respectively. When Caltech and ImageNet serve as source datasets, we pick all examples of them for training. When Caltech and ImageNet serve as target datasets, we pick the categories used in the testing phase of intra-dataset ZSML for testing. We use all examples in COIL and MSRC for testing, when COIL and MSRC are the target datasets. Table 4 and 5 reports the ACC, NMI and purity metrics. We can see that our method achieves strong transferability, although the great gap exists in different datasets. Furthermore, the Euclidean distance performances moderately in most cases, as it requires no training phase and the over-fitting problem will not occur.

We further concatenate our model after the Lifted algorithm [Oh Song *et al.*, 2016] to verify the effectiveness on large scale image dataset CUB_200_2011 [Wah *et al.*, 2011a]. Following the experiment protocol and evaluation metrics used in Lifted, we conduct the experiment and Table 6 reports the results, from which we can see that our model can significantly improve the representation power under zero-shot setting.

5.3 Ablation Study

We fix $\alpha = 0.3$ ($\beta = 1 - \alpha = 0.7$), and change the number of convolution kernels k to 16, 64, 96, and 128 respectively. On the other hand, parameter α varies in four values: $\{0.1, 0.2, 0.3, 0.4\}$ when fix $k = 96$. As illustrated in Table 7 and Table 8, k and α shows a positive tendency until $k = 96$ and $\alpha = 0.3$. Hence, we fix α to 0.3 and k to 96 throughout our experiments.

METHOD	Caltech → Caltech			ImageNet→ ImageNet		
	ACC	NMI	Purity	ACC	NMI	Purity
Euclidean	**53.66**	43.76	83.65	39.43	19.68	39.52
ITML	42.74	5.76	54.14	21.02	13.26	25.38
LMNN	52.37	44.54	83.58	**45.81**	**29.56**	**33.22**
GMML	66.28	**54.03**	**86.77**	43.12	27.58	31.92
MLCA	52.51	14.98	56.72	38.46	22.34	34.11
Ours	66.28	58.11	90.57	51.81	33.79	51.81

Table 3: Clustering results of intra-dataset transferability analysis. We train on some categories and measure the similarity of other categories in Caltech (ImageNet).

METHOD	Caltech → COIL			Caltech → ImageNet		
	ACC	NMI	Purity	ACC	NMI	Purity
Euclidean	61.81	75.86	67.22	39.43	19.68	39.52
ITML	12.92	10.21	13.89	18.38	3.24	18.95
LMNN	58.33	79.01	69.10	**46.00**	**31.93**	**49.62**
GMML	**71.94**	**89.75**	**80.00**	36.00	21.27	39.62
MLCA	9.86	8.85	1.97	20.00	3.82	20.67
Ours	75.97	88.63	83.13	57.72	40.24	58.29

Table 4: Clustering results of inter-dataset transferability analysis. We train our model on Caltech and measure the similarity of COIL and ImageNet.

METHOD	Caltech → MSRC			ImageNet→ Caltech		
	ACC	NMI	Purity	ACC	NMI	Purity
Euclidean	63.75	63.41	68.75	**53.66**	**43.76**	**83.65**
ITML	16.25	4.68	17.08	11.13	2.08	7.88
LMNN	**75.00**	**74.96**	**80.83**	45.00	43.66	40.62
GMML	63.75	63.41	68.75	43.25	43.41	40.29
MLCA	16.67	6.45	18.75	21.61	13.42	20.75
Ours	84.58	78.49	84.58	55.32	46.36	85.89

Table 5: Clustering results of inter-dataset transferability analysis. We train our model on Caltech (ImageNet) and measure the similarity of MSRC (Caltech).

METHOD	CUB → CUB			
	R@1	R2@1	R@4	R@8
Lifted	47.2	58.9	70.2	80.2
Lifted+Ours	48.9	59.3	72.1	83.9

Table 6: Comparison of the clustering performance, which proves the effectiveness of our model when cooperating with Lifted algorithm.

k	Caltech → ImageNet			Caltech→MSRC		
	ACC	NMI	Purity	ACC	NMI	Purity
$k = 16$	49.71	35.00	52.59	81.22	76.56	77.33
$k = 64$	51.88	38.48	54.47	81.36	72.11	79.13
$k = 96$	57.72	40.24	58.29	84.58	78.49	84.58
$k = 128$	52.14	40.72	55.29	84.50	77.29	83.23

Table 7: Comparison of the classification accuracy in % of varying k on inter-dataset ZSML task

α	Caltech →ImageNet			Caltech→MSRC		
	ACC	NMI	Purity	ACC	NMI	Purity
$\alpha = 0.1$	52.48	35.69	52.57	66.25	64.27	66.25
$\alpha = 0.2$	52.61	38.47	48.94	78.96	77.26	78.90
$\alpha = 0.3$	57.72	40.24	58.29	84.58	78.49	84.58
$\alpha = 0.4$	54.88	38.48	54.47	83.53	76.84	84.02

Table 8: Comparison of the classification accuracy in % of varying α on inter-dataset ZSML task.

METHOD	Caltech → COIL			Caltech → ImageNet		
	ACC	NMI	Purity	ACC	NMI	Purity
LMNN	58.33	79.01	69.10	46.00	31.93	49.62
GMML	71.94	89.75	80.00	36.00	21.27	39.62
LR+Reloss	77.57	91.46	84.44	49.62	32.95	49.62

Table 9: Comparison of the clustering performance when deprecating the multi-nonlinear module, which purposes to verify the effectiveness of regression loss.

To verify the priority of our proposed regression loss (Reloss), we switch the multi-nonlinear relations (MR) to linear relation (LR). For LR, we first minus two feature vectors of two data points, and employ S to project into similarity space. Table 9 reports the ACC, NMI and Purity metrics of LR + Reloss compared to LMNN and GMML. We can observe that Reloss based optimization is significant for improving the performance, mainly because of it relaxes the binary constraint employed on data pairs and the generalization capability is enhanced.

6 Conclusion

In this paper, we investigated zero-shot metric learning issue, which aims at measuring the similarity of data points from unseen categories or even unseen datasets. We reformulated a family of specific-shaped convolutional kernels to combine the data pair, which is capable capturing the multi-nonlinear relations. Furthermore, we designed a novel continuous-supervised regression loss, which can effectively preserve the continuous intra-data manifold. To sum up, our model greatly extends the transferability by learning the multi-nonlinear yet continuous relations. Extensive experiments, including intra-dataset ZSML and inter-dataset ZSML, verified the rationality and effectiveness of our proposed method.

Acknowledgments

Our work was also supported by the National Natural Science Foundation of China under Grant 61572388 and 61703327, Key R&D Program-The Key Industry Innovation Chain of Shaanxi under Grant 2017ZDCXL-GY-05-04-02, 2017ZDCXL-GY-05-02, and 2018ZDXM-GY-176, and the National Key R&D Program of China under Grant 2017YFE0104100.

References

[Cai *et al.*, 2008] Deng Cai, Xiaofei He, Xiaoyun Wu, and Jiawei Han. Non-negative matrix factorization on manifold. In *ICDM*, pages 63–72, 2008.

[Davis *et al.*, 2007] Jason V Davis, Brian Kulis, Prateek Jain, Suvrit Sra, and Inderjit S Dhillon. Information-theoretic metric learning. In *ICML*, pages 209–216, 2007.

[Deng *et al.*, 2009] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009.

[Donahue *et al.*, 2013] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: a deep convolutional activation feature for generic visual recognition. 50(1):I–647, 2013.

[Farhadi *et al.*, 2010] Ali Farhadi, Ian Endres, and Derek Hoiem. Attribute-centric recognition for cross-category generalization. In *CVPR*, pages 2352–2359. IEEE, 2010.

[Griffin and Perona, 2008] Gregory Griffin and Pietro Perona. Learning and using taxonomies for fast visual categorization. In *CVPR*, pages 1–8. IEEE, 2008.

[Guillaumin *et al.*, 2009] Matthieu Guillaumin, Jakob Verbeek, and Cordelia Schmid. Is that you? metric learning approaches for face identification. In *ICCV*, pages 498–505, 2009.

[Hadsell *et al.*, 2006] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, volume 2, pages 1735–1742, 2006.

[Hoi *et al.*, 2006] Steven CH Hoi, Wei Liu, Michael R Lyu, and Wei-Ying Ma. Learning distance metrics with contextual constraints for image retrieval. In *CVPR*, volume 2, pages 2072–2078, 2006.

[Huang *et al.*, 2015] Sheng Huang, Mohamed Elhoseiny, Ahmed Elgammal, and Dan Yang. Learning hypergraph-regularized attribute predictors. In *CVPR*, pages 409–417, 2015.

[Huang *et al.*, 2016] Chen Huang, Chen Change Loy, and Xiaoou Tang. Local similarity-aware deep feature embedding. In *NIPS*, pages 1262–1270, 2016.

[Jia *et al.*, 2014] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ACM*, pages 675–678, 2014.

[Lampert *et al.*, 2014] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-based classification for zero-shot visual object categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 36(3):453–465, 2014.

[Law *et al.*, 2016] Marc T Law, Yaoliang Yu, Matthieu Cord, and Eric P Xing. Closed-form training of mahalanobis distance for supervised clustering. In *CVPR*, pages 3909–3917, 2016.

[Lecun *et al.*, 1998] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. 86(11):2278–2324, Nov 1998.

[Marszalek and Schmid, 2007] Marcin Marszalek and Cordelia Schmid. Semantic hierarchies for visual object recognition. In *CVPR*, pages 1–7. IEEE, 2007.

[Mensink *et al.*, 2013] Thomas Mensink, Jakob Verbeek, Florent Perronnin, and Gabriela Csurka. Distance-based image classification: Generalizing to new classes at near-zero cost. *TPAMI*, 35(11):2624–2637, 2013.

[Nene *et al.*, 1996] Sameer A Nene, Shree K Nayar, Hiroshi Murase, et al. Columbia object image library (coil-20). 1996.

[Oh Song *et al.*, 2016] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *CVPR*, pages 4004–4012, 2016.

[Roweis *et al.*, 2004] Sam Roweis, Geoffrey Hinton, and Ruslan Salakhutdinov. Neighbourhood component analysis. *NIPS*, 17:513–520, 2004.

[Toshev and Szegedy, 2014] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In *CVPR*, pages 1653–1660, 2014.

[Wah *et al.*, 2011a] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical report, 2011.

[Wah *et al.*, 2011b] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[Weinberger and Saul, 2009] Kilian Q Weinberger and Lawrence K Saul. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10(Feb):207–244, 2009.

[Xing *et al.*, 2003] Eric P Xing, Michael I Jordan, Stuart J Russell, and Andrew Y Ng. Distance metric learning with application to clustering with side-information. In *NIPS*, pages 521–528, 2003.

[Xiong *et al.*, 2012] Caiming Xiong, David Johnson, Ran Xu, and Jason J Corso. Random forests for metric learning with implicit pairwise position dependence. In *SIGKDD*, pages 958–966, 2012.

[Ye *et al.*, 2016] Han-Jia Ye, De-Chuan Zhan, Xue-Min Si, Yuan Jiang, and Zhi-Hua Zhou. What makes objects similar: A unified multi-metric learning approach. In *NIPS*, pages 1235–1243, 2016.

[Zadeh *et al.*, 2016] Pourya Zadeh, Reshad Hosseini, and Suvrit Sra. Geometric mean metric learning. In *ICML*, pages 2464–2471, 2016.

[Zhou *et al.*, 2004] Denny Zhou, Olivier Bousquet, Thomas N Lal, Jason Weston, and Bernhard Schölkopf. Learning with local and global consistency. In *NIPs*, pages 321–328, 2004.