Prototype Relaxation With Robust Principal Component Analysis for Zero Shot Learning

YONG MA1, HUAQI MAO1, HAOFENG ZHANG1, AND WENBO WANG2

1School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China
2Nanjing Audit University, Nanjing 211815, China

Corresponding author: Wenbo Wang (mz1909004@stu.nau.edu.cn)

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 61872187, and in part by the 111 Program under Grant B13022.

ABSTRACT Zero Shot Learning (ZSL) has been attracting increasing attention due to its powerful ability of recognizing objects of unseen classes. As one type of ZSL methods, the low rank based strategy has achieved remarkable success. However, traditional low rank based methods are often based on the assumption that a variety of visual features from a same class can be projected to a single attribute by ignoring the background information and other noisy interference in visual features. This assumption is unreasonable and often leads to bad performance when there is big variance within a class. In this paper, a novel method called Prototype Relaxation with Robust Principal Component Analysis (RPCA) is proposed to relax this assumption by adding a sparse noise constraint. In addition, to avoid the confusion between similar classes, an orthogonal constraint is employed to disperse all the class prototypes, including both seen and unseen classes, in latent space. Furthermore, to alleviate the domain shift problem, vectors from latent space are exploited to reconstruct visual features and semantic attributes respectively. Besides, the hubness problem is also mitigated by applying the max probability model in all three spaces. Extensive experiments are conducted on four popular datasets and the results demonstrate the superiority of this method.

INDEX TERMS Image classification, prototype relaxation, robust principal component analysis, zero shot learning.

I. INTRODUCTION

Due to the rapid development of deep learning, the technique of image classification has gained remarkable achievement [1], such as Deep Residual Network (ResNet) [2] can achieve over 95% top-5 accuracy on the 1000-category image dataset — ImageNet [3], which has been proven to surpass the recognition ability of human beings. However, these image classification methods can only survive on the close set setting, when there comes a sample of new category which did not appear in the training set, they will definitely get wrong result.

To solve the open set image classification problem, Lampert et al. proposed a novel method called Zero Shot Learning (ZSL) [4], which can recognize the samples of unseen classes by transferring the knowledge learned from seen classes with semantic attribute bridging. For example, one can recognize a new category of objects which he has never seen before by being told how they are similar to other known objects [5]. Similarly, ZSL transfers this knowledge, which is called semantic embeddings or attributes, from seen classes to unseen ones [6]–[9].

So far, an increasing number of researchers have been devoting to ZSL, and many methods have been proposed. These methods can be simply divided into two classes by their compatibility with new category. One is unseen feature synthesis based method, which first trains a generative model with Generative Adversarial Net (GAN) [10] to synthesize sample of unseen classes, and then trains a fully supervised model. Another is direct method, which trains a classification model with only the samples of seen classes and then directly classify unseen data with this model. In the second category, low rank based methods can achieve state-of-the-art performance, they constrain the projection matrix from visual space to attribute space to be low rank, and add this constraint to other loss functions. For example, Ding et al. attached it to a attribute dictionary constraint and utilized an ensemble learning strategy to optimize them [11]. Liu et al. embedded it
Prototype Relaxation with RPCA based ZSL. Niu et al. tried to use a Semantic Autoencoder (SAE) [12], and employed a trace optimization method to solve it [13]. However, all these methods are based on an unreasonable assumption that varieties of visual features from one category can simultaneously be projected to a single attribute, and they ignore that the visual features still contain background information or other noisy factors, which will prevent them to be projected to a single vector. The illustration of the low rank constraint is shown in Fig. 1(A).

To solve this problem, we relax this assumption and add a sparse constraint to it to contain the noise information, which is named Prototype Relaxation with Robust Principal Component Analysis (RPCA) based ZSL method and illustrated in Fig. 1(B). This relaxation can perfectly model the projection and well solve the interference from the noise. In addition, because some attributes in unseen classes are very similar to others, it will lead to wrong classification, especially for the more realistic and challenging GZSL setting. For example, “the humpback whale” has very similar attributes to that of “blue whale”, only several entries like color are different among all the 85 dimensions. Therefore, to disperse class prototypes, we build a medium latent space, where the prototypes projected from different attributes are constraint to be orthogonal to each other and normalized. Furthermore, the projection from visual space to attribute space or latent space often suffers from the domain shift problem [12], so we employ the concept from SAE [12] to reconstruct the visual features and semantic attributes respectively. In the prediction stage, to alleviate the hubness problem, which is caused by using only the attribute space or the latent space, we combine them with the visual space together and take the maximum probability as the final classification result. As a supplement to inductive ZSL, we also extend this method to transductive setting by employing the learned parameters of inductive model. The contributions of our work can be summarized as follows,

1) To avoid the unreasonable projection assumption in traditional low rank based ZSL methods, we proposed a prototype relaxation with RPCA based approach to relax it by adding a sparse item to incorporate the noisy redundant information;
2) An orthogonal constraint in latent space is employed to disperse all class prototypes by making them normalized and orthogonal to each other;
3) Self reconstruction is applied to alleviate the domain shift problem, and a probabilistic prediction in all three spaces are utilized to further mitigate the hubness problem;
4) Experiments for both inductive setting and transductive setting are conducted on four popular datasets, and the results on both ZSL and GZSL demonstrate the superiority of the proposed method.

The main content of this paper is organized as follows: In section II we briefly introduce the existing methods for ZSL and GZSL. Section III describes the proposed method in detail. Section IV gives the experimental results of comparison with existing methods on several metrics. Finally in section V, we conclude this paper.

II. RELATED WORKS

A. ZERO SHOT LEARNING

Zero Shot Learning (ZSL) models try to classify unseen samples by transferring the knowledge learned from seen classes with semantic embeddings bridging. So far, thousands of researchers have been devoting to this research domain. The earliest efforts such as Direct Attribute Projection (DAP) [4] estimate the labels by learning probabilistic attribute classifiers. In Attribute Label Embedding (ALE) [15] and SJE [16], Akata et al. projected visual features into semantic space via a bilinear compatibility constraint. CONvex combination of Semantic Embeddings (CONSE) [17] and Semantic Similarity Embedding (SSE) [18] try to build unseen attributes automatically from the instances of seen categories to reduce the requirement of manual attributes. Furthermore, some researchers such as Kodirov et al. [12] introduced the concept of Auto-Encoder and directly use the Euclidean distance to constrain the similarity of projected vectors in both visual and attribute spaces. In addition, Long et al. in [19] proposed to use the attributes of unseen classes to synthesize unseen visual features, and then train a supervised model with seen and synthesized unseen visual features. Thereafter, due to the powerful ability of sample synthesis, an increasing number of Generative Adversarial Net (GAN) [10] based ZSL method has been proposed [20–22]. However, these generative methods all suffer form the same problem as the close-set classification that when there is a totally new category it should be retrained by adding the synthesized samples of the new class.
The most relevant to ours are the low rank based methods [11], [13], [14], [23]. Ding et al. in [11] assumed that the projection matrix from visual space to attribute space should have the characteristic of low rank, and exploited a constraint on singular values to solve the problem. Liu et al. in [13] made the same assumption and apply the concept of SAE to constrain the projection matrix, which is solved by replacing the low rank constraint to a nuclear norm. Meng et al. [14], [23], [25] in 2017 put forward a new split of several popular datasets for GZSL, and released a benchmark of some recent ZSL methods [24], [25], [26]. Liu et al. designed a Deep Calibration Network (DCN) to enable simultaneous calibration of deep networks on the confidence of source classes and uncertainty on unseen classes as well.

III. METHODOLOGY

A. PROBLEM DEFINITION

Let \( S = \{s_1, \ldots, s_p\} \) denotes a set of seen classes and \( \mathcal{U} = \{u_1, \ldots, u_q\} \) denotes a set of unseen classes, where \( p \) and \( q \) are the number of seen and unseen classes respectively. The two sets are disjoint, that is to say, \( S \cap \mathcal{U} = \emptyset \). Besides, \( A^s = \{a^s_1, \ldots, a^s_q\} \in \mathbb{R}^{d_s \times p} \) and \( A^u = \{a^u_1, \ldots, a^u_q\} \in \mathbb{R}^{d_u \times q} \) represent for the corresponding seen and unseen class level semantic representations, such as attributes or word embeddings, where \( d_a \) is the dimension of attribute vector.

Given a set of labeled training data of seen classes \( X^s = \{x^s_1, \ldots, x^s_i, \ldots, x^s_N^s\} \in \mathbb{R}^{d_s \times N_s} \), where \( d_s \) is the dimensionality of a single feature vector \( x^s_i \), and \( N_s \) is the number of training data. Each feature \( x^s_i \) is simultaneously associated with a label \( y_i \in S \) and its corresponding attribute \( a^s_i \in A^s \). Let \( X^u = \{x^u_1, \ldots, x^u_j, \ldots, x^u_N^u\} \in \mathbb{R}^{d_u \times N_u} \) represents for a set of test data, which is not assigned with its corresponding labels and semantic representations, where \( N_u \) is the number of test data. The objective of ZSL is to predict the labels of the test data \( X^u \) by learning a classifier \( \mathcal{F} : X^u \rightarrow \mathcal{U} \) with the training data \( X^s \) and the whole attribute set \( A^s \cup A^u \).

B. PROTOTYPE RELAXATION WITH RPCA BASED ZSL

In this subsection, we will present our novel prototype relaxation with RPCA based ZSL method, followed by an effective solution, and the whole framework of the method is shown in Fig. 2.

We first define a \( d_L \) dimensional latent space, which is hoped to be discriminative for both seen and unseen classes.
Suppose \( K \in \{0,1\}^{(p+q) \times N} \) is the prototype matrix, including both seen and unseen classes, in latent space, the projected vectors of visual features \( X \) should satisfy the following constraint,

\[
W^T_1X = L + E,
\]

\[s.t. L = KB,
\]

where, \( B \in \{0,1\}^{(p+q) \times N} \) is the one-hot label matrix for visual feature \( X' \), \( E \in \mathbb{R}^{d_1 \times N} \) is the sparse noise around the class prototypes, and \( W_1 \in \mathbb{R}^{d_1 \times d_2} \) is the projection matrix from visual space to latent space.

It is known that \( B \) is low rank because there are many samples belonging to a single category, which leads to that \( L \) is also low rank. Besides, \( E \) is hoped to be sparse noise around the prototypes, thus we assume that it is \( \ell_1 \) norm sparse and define the following constraint function,

\[
\mathcal{L}_{r_pca} = \|L\|_s + \lambda_1\|E\|_1,
\]

\[s.t. W^T_1X = L + E, L = KB,
\]

where, \( \|\cdot\|_s \) is the nuclear norm, which constrains the value to be low rank, and \( \lambda_1 \) is the balancing coefficient.

Since the semantic preserving can alleviate the domain shift problem [12], we apply the autoencoder in latent space for both visual features and semantic attributes,

\[
\begin{align*}
&W^T_1X = L + E \\
&W_1(L + E) = X \\
&W^T_2A = K \\
&W_2K = A,
\end{align*}
\]

where, \( W_2 \in \mathbb{R}^{d_2 \times d_1} \) is the projection matrix from attribute space to latent space. Because \( W^T_1X = L + E \) is a hard constraint, we use a relaxed manner to define the loss function for this autoencoder,

\[
\mathcal{L}_{ae} = \lambda_2(\|W^T_1X - KB\|_F^2 + \|X - W_1KB\|_F^2) + \lambda_3(\|W^T_2A - K\|_F^2 + \|W_2K - A\|_F^2),
\]

where, \( \lambda_2 \) and \( \lambda_3 \) are balancing coefficients, and \( \|\cdot\|_F \) is the Frobenius norm.

The latent space is hoped to be discriminative, but the dimension of latent space is limited, so it is impossible to scatter all the prototypes far way from each other. Here, we use an alternative strategy that all the class prototypes are constrained to be normalized and orthogonal to each other. Therefore, we define a loss function as follows,

\[
\mathcal{L}_{or} = \lambda_4\|K^TK - I\|_F^2,
\]

where, \( \lambda_4 \) is also a balancing coefficient. Combing all the items in Eq.2, Eq.3 and Eq.5, we can obtain the final loss function,

\[
\mathcal{L} = \|L\|_s + \lambda_1\|E\|_1 + \lambda_2(\|W^T_1X - L\|_F^2 + \|X - W_1L\|_F^2)
+ \lambda_3(\|W^T_2A - K\|_F^2 + \|W_2K - A\|_F^2)
+ \lambda_4\|K^TK - I\|_F^2 + \lambda_5\|L - KB\|_F^2,
\]

\[s.t. W^T_1X = L + E,
\]

where, \( \lambda_5 \) is a balancing coefficient too, and it controls the importance of the last item.

### C. SOLUTION

Since Eq. 6 is not joint convex over all variables, there is no close-form solution simultaneously. Thus, we propose an iterative optimization strategy to update a single unresolved variable each time. Because proper initialization parameters can not only improve the model performance but also increase the convergence speed, we further split the solution into two sub problems, i.e., initialization for \( W_1, W_2 \) and \( K \), and iterative optimization of all the variables.

1) **INITIALIZATION**

In conventional iterative optimization usually exploits random initialization for some variables. However, random initialization often lead to slow convergence and bad performance when there are too many variables, especially in Eq. 6, we need to randomly initialize four variables, which will make it hard to converge. Here, we first utilize a truncated version of Eq. 6 by removing the parts of \( L \) and \( E \) to initialize \( W_1 \) and \( W_2 \), which can be represented as,

\[
\mathcal{L}_{I_V} = \|W^T_1X - KB\|_F^2 + \|X - W_1KB\|_F^2
+ \lambda_3(\|W^T_2A - K\|_F^2 + \|W_2K - A\|_F^2)
+ \lambda_4\|K^TK - I\|_F^2.
\]

By taking the derivative of Eq. 8 with respected to \( W_1 \) and set it to zero, we can obtain the following representation,

\[
XX^TW_1 + W_1KBB^TK = 2XB^TK^T.
\]

By setting \( \hat{A} = XX^T, \hat{B} = KBB^TK \) and \( \hat{C} = 2XB^TK^T \), Eq. 8 is converted to a well-known Sylvester equation which can be solved efficiently by the Bartels-Stewart algorithm [30], and it can be implemented with a single line of code \( W_1 = \text{sylvester}(\hat{A}, \hat{B}, \hat{C}) \) in MATLAB.\(^1\) Similarly, \( W_2 \) can also be solved with the same Sylvester equation by setting \( \hat{A} = AA^T, \hat{B} = KK^T \) and \( \hat{C} = 2AK^T \).

Similar as \( W_1 \) and \( W_2 \), we take the derivative of Eq. 7 with respected to \( K \), and obtain the following equation,

\[
\frac{\partial \mathcal{L}_{I_V}}{\partial K} = 2(\hat{K}B - W^T_1X)\hat{B}^T + 2W^T_1(W_1KB - X)\hat{B}^T
+ 2\lambda_3(\hat{K} - W^T_2A) + W^T_2(W_2K - A)
+ 4\lambda_4(K^TK - I).
\]

Because there is a third order item in Eq. 10, it is hard to directly solve the closed-form solution. Here, we adopt the

\(^1\)https://uk.mathworks.com/help/matlab/ref/sylvester.html
Stochastic Gradient Descent (SGD) strategy,

\[ K = K - \alpha \frac{\partial \mathcal{L}_T}{\partial K}, \]  

(11)

where, \( \alpha \) is the learning rate.

2) OPTIMIZATION

Instead of adopting the traditional Singular Value Thresholding (SVT) [31] to solve the nuclear norm minimization in Eq. 6, we exploit a regularization term that guarantees the low rank characteristic of optimized \( L \). Mathematically, we have the following equation,

\[ \|L\|_* = tr((L^T L)^{\frac{1}{2}}) = tr(L^T (L L^T)^{-\frac{1}{2}} L) \]

\[ = tr(L^T H L), \]  

(12)

where \( H = (LL^T)^{-\frac{1}{2}} \). Besides, due to the existence of the hard constraint, we apply the Augmented Lagrangian Multiplier (ALM), and Eq. 6 can be converted as,

\[ \mathcal{L} = tr(L^T H L) \]

\[ + \lambda_1 \|E\|_1 + \lambda_2 (\|W_1^T X - L\|_F^2 + \|X - W_1 L\|_F^2) \]

\[ + \lambda_3 (\|W_2^T A - K\|_F^2 + \|W_2 K - A\|_F^2) \]

\[ + \lambda_4 (K^T K - I)^2 + \lambda_5 \|L - KB\|_F^2 \]

\[ + \frac{\mu}{2} \|E - (W_1^T X - L)\|_F^2 \]

\[ + \langle R, E - (W_1^T X - L) \rangle, \]

(13)

where \( \mu \) is a penalty parameter and \( R \) is a Lagrangian multiplier.

After the initialization, \( W_1, W_2 \) and \( K \) are close to the correct values. Therefore, we exploit them as the initial values to optimization all the five variables and the Lagrangian multiplier \( R \) until the termination criterion is met. Updating \( R \) and \( \mu \) is trivial, which can be found in supplementary material. In the following, we will describe how to update \( E, L, K, W_1 \) and \( W_2 \) one by one.

Update \( E \): The subproblem of Eq. 13 w.r.t. \( E \) is as follows,

\[ E = \arg \min_E \mathcal{L}_E \]

\[ = \arg \min_E \|E\|_1 + \frac{\mu}{2} \|E - (W_1^T X - L)\|_F^2 \]

\[ + \|R, E - (W_1^T X - L)\rangle \]

\[ = \arg \min_E \|E\|_1 + \frac{\mu}{2} \|E - (W_1^T X - L - \frac{R}{\mu})\|_F^2 \]

\[ = S_{\lambda_1/\mu}(W_1^T X - L - \frac{R}{\mu}), \]  

(14)

where \( S_{\tau}(Y) \) is a Soft Thresholding (ST) method, which can be implemented by \( S_{\tau}(Y) = \text{sign}(Y) \circ \max(|Y| - \tau, 0) \) [32], and \( \circ \) is element-wise multiply.

Update \( L \): The subproblem of Eq. 13 w.r.t. \( L \) is as follows,

\[ L = \arg \min_L \mathcal{L}_L \]

\[ = \arg \min \langle \|L^T H L\| + \lambda_5 \|L - KB\|_F^2 \]

\[ + \lambda_2 (\|W_1^T X - L\|_F^2 + \|X - W_1 L\|_F^2) \]

\[ + \frac{\mu}{2} \|E - (W_1^T X - L)\|_F^2 \]

\[ + \langle R, E - (W_1^T X - L) \rangle. \]  

(15)

By taking the derivative of \( \mathcal{L}_L \), w.r.t. \( L \), we can obtain the following equation,

\[ 2HL + 2\lambda_5 (L - KB) + \mu (E - (W_1^T X - L)) + R \]

\[ + 2\lambda_2 ((L - W_1^T X) + W_1^T (W_1 L - X)) = 0 \]

\[ \iff \]

\[ ((2\lambda_2 + 2\lambda_5 + \mu)I + 2H + 2\lambda_2 W_1^T W_1)L \]

\[ = (4\lambda_2 + \mu)W_1^T X + 2\lambda_5 KB - \mu E - R. \]

(16)

Then, we can obtain the solution of \( L \) as,

\[ L = ((2\lambda_2 + 2\lambda_5 + \mu)I + 2H + 2\lambda_2 W_1^T W_1)^{-1} \]

\[ ((4\lambda_2 + \mu)W_1^T X + 2\lambda_5 KB - \mu E - R). \]

(17)

Update \( K \): The subproblem of Eq. 13 w.r.t. \( K \) is as follows,

\[ K = \arg \min_K \mathcal{L}_K \]

\[ = \arg \min_K \|W_2^T A - K\|_F^2 + \|W_2 K - A\|_F^2 \]

\[ + \lambda_4 (K^T K - I)^2 + \lambda_5 \|L - KB\|_F^2 \]

(18)

Then the derivative of Eq. 18 can be obtained as,

\[ \frac{\partial \mathcal{L}_K}{\partial K} = 2\lambda_3 ((K - W_1^T A) + W_2 (W_2 K - A)) \]

\[ + 4\lambda_5 (K^T K - I) + 2\lambda_3 (KB - L)B^T, \]  

(19)

and it can be solved with same SGD method as Eq. 11.

Update \( W_1 \) and \( W_2 \): The subproblem of Eq. 13 w.r.t. \( W_1 \) is as follows,

\[ \mathcal{L}_{W_1} = \arg \min_{W_1} \mathcal{L}_{W_1} \]

\[ = \arg \min_{W_1} \|W_1^T X - L\|_F^2 + \|X - W_1 L\|_F^2 \]

\[ + \frac{\mu}{2} \|E - (W_1^T X - L)\|_F^2 \]

\[ + \langle R, E - (W_1^T X - L) \rangle. \]  

(20)

By taking the derivative of \( \mathcal{L}_{W_1} \), w.r.t. \( W_1 \) and set it to zero, we can obtain the following equation,

\[ 2\lambda_2 (X (X^T W_1 - L^T) + (W_1 L - X)L^T) \]

\[ - \mu X (E^T - X^T W_1 + L^T) - XR^T = 0 \]

\[ \iff \]

\[ (2\lambda_2 + \mu)XX^T W_1 + 2\lambda_2 W_1 LL^T \]

\[ = (4\lambda_2 + \mu)XL^T + XR^T + \muXE^T. \]

(21)

If we set \( \tilde{A} = (2\lambda_2 + \mu)XX^T, \tilde{B} = 2\lambda_2 LL^T \) and \( \tilde{C} = (4\lambda_2 + \mu)XL^T + XR^T + \muXE^T \), Eq. 21 is converted to
$\tilde{A}W_1 + W_1B = \tilde{C}$, which can be solve with Sylvester equation. Similarly, by setting $\tilde{A} = AA^T$, $\tilde{B} = KK^T$ and $\tilde{C} = 2AK^T$, we can also obtain the solution of $W_2$ with the same Sylvester equation. The whole algorithm can be found in supplementary material.

Algorithm 1 The Detailed Algorithm of RPCA Based ZSL

**Input:** The training data $X$, its corresponding labels $B$ and class level attributes $A$.

The hyper-parameters $\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \mu, \rho$ and the learning rate $\alpha$;

The iterative number $\text{iter}_1$ for initializing $W_1$ and $W_2$ and the iterative number $\text{iter}_2$ for optimizing all variables.

**Output:** The learned projection matrices $W_1$ and $W_2$.

1. Randomly initialize $K$;
2. for all $k = 1 \rightarrow \text{iter}_1$ do
3. Update $W_1$ with Sylvester equation by setting $\tilde{A} = XX^T$, $\tilde{B} = KKB^TK^{-1}$ and $\tilde{C} = 2XB^TK^{-1}$;
4. Update $W_2$ with Sylvester equation by setting $\tilde{A} = AA^T$, $\tilde{B} = KK^T$ and $\tilde{C} = 2AK^T$;
5. for all $m = 1 \rightarrow 25$ do
6. Compute the derivative of $\partial L_{TV}$ w.r.t. $K$ with Eq. 10, and update $K$ with Eq. 11;
7. end for
8. end for
9. Randomly initialize $R$;
10. Initialize $H$ with $H = (LL^T)^{-\frac{1}{2}}$ and $L = WX$;
11. for all $k = 1 \rightarrow \text{iter}_2$ do
12. $E$ with Eq. 14;
13. $L$ with Eq. 17;
14. for all $m = 1 \rightarrow 25$ do
15. Update $H$ with $H = (LL^T)^{-\frac{1}{2}}$;
16. Compute the derivative of $L_K$ w.r.t. $K$ with Eq. 19, and update $K$ with $K = K - \frac{\partial L_{TV}}{\partial X}$;
17. end for
18. Update $W_1$ with Sylvester equation $\tilde{A}W_1 + W_1B = \tilde{C}$ by setting $\tilde{A} = (2\lambda_2 + \mu)XX^T$, $\tilde{B} = 2\lambda_2LL^T$ and $\tilde{C} = (4\lambda_2 + \mu)XL^T + XR^T + \muXE^T$;
19. Update $W_2$ with Sylvester equation $\tilde{A}W_2 + W_2B = \tilde{C}$ by setting $\tilde{A} = AA^T$, $\tilde{B} = KK^T$ and $\tilde{C} = 2AK^T$;
20. Update $R$ by $R = R + \mu(E - (W_1^TX - L))$;
21. Update the parameter $\mu$ by $\mu = \rho \mu$;
22. end for
23. Return the learned $W_1$ and $W_2$.

**D. TRANSDUCTIVE SETTING**

Since $W_1$ and $W_2$ are generated only with seen data, it is impossible to fully tackle the domain shift problem. The most effective method is to include the unlabeled unseen data in training phase, which is called transductive setting and first proposed by Fu et al. [33]. It is known that the projection matrices $W_1$ and $W_2$ of seen classes are different from that of unseen classes due to the domain shift problem, but they are very similar because of the application of semantic attributes. Therefore, we modify the inductive equation with unlabeled unseen data to convert it to transductive setting, which can be represented as,

$$L'' = ||L_u||_F + \beta_1 ||E_u||_1 + \beta_2 ||W_1^uX_u - L_u||^2_F + ||X_u - W_1uL_u||^2_F \right) + \beta_3 ||W_2uA_u - K_u||^2_F + ||A_u - W_2uK_u||^2_F \right) + \beta_4 ||W_1 - W_1uL_u||^2_F + ||W_2 - W_2uK_u||^2_F \right) + \beta_5 ||L_u - K_uB_u||^2_F + \frac{\mu}{2} ||E - (W_1^TX_u - L_u)\right)^2_F + \frac{\mu}{2} ||E - (W_1^TX_u - L_u)\right)^2_F

$$

where $\beta_1, \beta_2, \beta_3, \beta_4$ and $\beta_5$ are balancing coefficients, 1 is the all one vector. By replacing nuclear norm with trace form, and converting the constraint into Augmented Lagrange Multiplier item, Eq. 22 can be represented as,

$$L'' = tr(L_u^TH_uL_u) + \beta_1 ||E_u||_1 + \beta_2 ||W_1^uX_u - L_u||^2_F + ||X_u - W_1uL_u||^2_F \right) + \beta_3 ||W_2uA_u - K_u||^2_F + ||A_u - W_2uK_u||^2_F \right) + \beta_4 ||W_1 - W_1uL_u||^2_F + ||W_2 - W_2uK_u||^2_F \right) + \beta_5 ||L_u - K_uB_u||^2_F + \frac{\mu}{2} ||E - (W_1^TX_u - L_u)\right)^2_F + \frac{\mu}{2} ||E - (W_1^TX_u - L_u)\right)^2_F

$$

where $H_u = (L_uL_u^T)^{-\frac{1}{2}}$. Besides, we also remove the orthogonal constraint from Eq. 23 because there are only unseen classes needed to be processed, and the item $||W_2 - W_2uK_u||^2_F$ can constrain it to maintain the characteristic of orthogonality.

Eq. 23 is not joint convex for all the variables, thus we use the same iterative optimization strategy as inductive setting to solve it.

**Update $B_u$:** The subproblem of Eq. 23 w.r.t. $B_u$ is as follows,

$$L_{B_u} = ||L_u - K_uB_u||^2_F \right),

\text{s.t. } B_u \in [0, 1]^{q \times N_u}, \quad 1^T B_u = 1^T.

$$

Due to the existence of the discrete constraint, Eq. 24 cannot be directly solved. We expand it with the form of trace,

$$B_u = \arg \min \limits_{B_u} L_{B_u}

\right) + 2tr(B_0^T K_0 B_u L_u)

+ tr(K_u^T K_u B_u B_0^T L_u)

= \arg \min \limits_{B_u} - 2tr(B_0^T K_0 L_u).

$$

According to the discrete constraint, we can obtain the following solution for a single column $j$ of $B_u$,

$$b_{ij} = \begin{cases} 1 & i = \arg \max \limits_j (|K_j^T L_u|) \\ 0 & \text{otherwise,} \end{cases}

$$
where $K_u$ is initialized with $W_2A_u$, $L_u$ is initialized with $W_1X_u$.

**Update $E_u$:** The subproblem of Eq. 23 w.r.t. $E_u$ is as follows,

$$E_u = \arg\min_{E_u} \mathcal{L}_{E_u}$$

$$= \arg\min_{E_u} \beta_1 \|E\|_1 + \frac{\mu}{2} \|E - (W^T_{1u}X_u - L_u)\|^2_F$$

$$+ \langle R, E - (W^T_{1u}X_u - L_u) \rangle$$

$$= \arg\min_{E_u} \beta_1 \|E\|_1 + \frac{\mu}{2} \|E - (W^T_{1u}X_u - L_u) - \frac{R}{\mu} \|^2_F$$

$$= S_{\beta_1/\mu}(W^T_{1u}X_u - L_u - \frac{R}{\mu}), \quad (27)$$

where $L_u$ is initialized with $L_u = W^T_{1u}X_u$.

**Update $L_u$:** The subproblem of Eq. 23 w.r.t. $L_u$ is as follows,

$$L_u = tr(I^T_nH_nL_u) + \beta_5 \|L_u - K_uB_u\|^2_F$$

$$+ \beta_2 \|W^T_{1u}X_u - L_u\|^2_F + \|X_u - W_{1u}L_u\|^2_F$$

$$+ \langle R, E_u - (W^T_{1u}X_u - L_u) \rangle$$

$$
\iff

$$L_u = tr(I^T_nH_nL_u) + \beta_5 \|L_u - K_uB_u\|^2_F$$

$$+ \beta_2 \|W^T_{1u}X_u - L_u\|^2_F + \|X_u - W_{1u}L_u\|^2_F$$

$$+ \langle R, E_u - (W^T_{1u}X_u - L_u) \rangle. \quad (28)$$

By taking the derivative of $\mathcal{L}_{L_u}$ w.r.t. $L_u$, we can obtain the following equation,

$$2H_nL_u + 2\beta_5(L_u - K_uB_u) + \mu(E_u - (W^T_{1u}X_u - L_u))$$

$$+ R_u + 2\beta_2((L_u - W^T_{1u}X_u) + W^T_{1u}(W_{1u}L_u - X_u)) = 0$$

$$\iff$$

$$(2\beta_2 + 2\beta_5 + \mu)I + 2H_n + 2\beta_2W^T_{1u}W_{1u}L_u$$

$$= (4\beta_2 + \mu)W^T_{1u}X_u + 2\beta_5K_uB_u - \mu E_u - R_u. \quad (29)$$

Then, we can obtain the solution of $L_u$ as,

$$L_u = ((2\beta_2 + 2\beta_5 + \mu)I + 2H_n + 2\beta_2W^T_{1u}W_{1u})^{-1}$$

$$(4\beta_2 + \mu)W^T_{1u}X_u + 2\beta_5K_uB_u - \mu E_u - R_u. \quad (30)$$

**Update $K_u$:** The subproblem of Eq. 23 w.r.t. $K_u$ is as follows,

$$K_u = \beta_3\left(\|W^T_{2u}A_u - K_u\|^2_F + \|A_u - W_2K_u\|^2_F\right)$$

$$+ \beta_5 \|L_u - K_uB_u\|^2_F . \quad (31)$$

Set the derivative of Eq. 31 w.r.t. $K_u$ to zero, we can obtain,

$$\beta_3(I + W^T_{2u}W_{2u})K_u + \beta_5K_uB_uB^T_u$$

$$= 2\beta_3W^T_{2u}A_u + \beta_5L_uB^T_u . \quad (32)$$

which can be solve with Sylvester equation by setting $\tilde{A} = \beta_3(I + W^T_{2u}W_{2u})$, $\tilde{B} = \beta_5B_uB^T_u$ and $\tilde{C} = 2\beta_3W^T_{2u}A_u + \beta_5L_uB^T_u$.

**Update $W_{1u}$ and $W_{2u}$:** The subproblem of Eq. 23 w.r.t. $W_{1u}$ is as follows,

$$\mathcal{L}_{W_{1u}} = \beta_2(\|W^T_{1u}X_u - L_u\|^2_F + \|X_u - W_{1u}L_u\|^2_F)$$

$$+ \beta_4 \|W_1 - W_{1u}\|^2_F + \frac{\mu}{2} \|E - (W^T_{1u}X_u - L_u)\|^2_F$$

$$+ \langle R, E - (W^T_{1u}X_u - L_u) \rangle.$$ 

$$\iff$$

$$2\beta_2(X_uX^T_uW_{1u} - L_u) + (W_{1u}L_u - X_uL_u^T) - XR^T$$

$$+ \beta_4(W_1 - W_{1u}) - \mu X(E^T - XT^TW_{1u} + L) = 0$$

$$\iff$$

$$(2\beta_2 + \mu)X_uX^T_uW_{1u} + W_{1u}(2\beta_2L_uL_u^T + \beta_4I)$$

$$= (4\beta_2 + \mu)X_uL_u^T + X_uR_u^T + \mu X_uE^T_u + \beta_5W_1. \quad (34)$$

If we set $\tilde{A} = (2\beta_2 + \mu)X_uX^T_u$, $\tilde{B} = 2\beta_2L_uL_u^T + \beta_4I$ and $\tilde{C} = (4\beta_2 + \mu)X_uL_u^T + X_uR_u^T + \mu X_uE^T_u + \beta_4W_1$, Eq. 34 is converted to $\tilde{A}W_{1u} + W_{1u}\tilde{B} = \tilde{C}$, which can be solve with Sylvester equation. Similarly, by setting $\tilde{A} = \beta_3A_uB^T_u$, $\tilde{B} = \beta_3K_uK_u^T + \beta_4I$ and $\tilde{C} = 2\beta_3AK^T + \beta_4W_2$, we can also obtain the solution of $W_{2u}$ with the same Sylvester equation. The whole algorithm can be found in Alg. 2.

**Algorithm 2** The Detailed Algorithm of Transductive Setting for RPCA Based ZSL

**Input:** The unseen data $X_u$, unseen class level attributes $A_u$, and the initial projection matrices $W_1$ and $W_2$ learned from inductive setting; The hyper-parameters $\beta_1$, $\beta_2$, $\beta_3$, $\beta_4$, $\beta_5$, $\mu$, and $\rho$; The iterative number $iter$.

**Output:** The learned label matrix $B_u$ of unseen classes.

**Initialize** $L_u$ with $W^T_{1u}X_u$;

**Initialize** $K_u$ with $W^T_{2u}A_u$;

**Initialize** $H_u$ with $H = (L_uL_u^T)^{-\frac{1}{2}}$;

**for all** $k = 1 \rightarrow iter$ **do**

**Update** $B_u$ with Eq. 26;

**Update** $E_u$ with Eq. 27;

**Update** $L_u$ with Eq. 30;

**Update** $H_u$ with $H_u = (L_uL_u^T)^{-\frac{1}{2}}$;

**Update** $K_u$ with Eq. 32;

**Update** $W_1$ with Sylvester equation $\tilde{A}W_{1u} + W_{1u}\tilde{B} = \tilde{C}$ by setting $\tilde{A} = (2\beta_2 + \mu)X_uX^T_u$, $\tilde{B} = 2\beta_2L_uL_u^T + \beta_4I$ and $\tilde{C} = (4\beta_2 + \mu)X_uL_u^T + X_uR_u^T + \mu X_uE^T_u + \beta_4W_1$;

**Update** $W_2$ with Sylvester equation $\tilde{A}W_{2u} + W_{2u}\tilde{B} = \tilde{C}$ by setting $\tilde{A} = \beta_3A_uB^T_u$, $\tilde{B} = \beta_3K_uK_u^T + \beta_4I$ and $\tilde{C} = 2\beta_3AK^T + \beta_4W_2$;

**Update** $R_u$ by $R_u = R_u + \mu(E_u - (W^T_{1u}X_u - L_u))$;

Update the parameter $\mu$ by $\mu = \rho\mu$;

**end for**

Return the learned $B_u$. 
E. Zero Shot Image Classification

Due to the fact that the hubness problem often appears when conducting prediction in low dimensional spaces [34] such as attribute space or latent space, so it is necessary to transfer the classification into visual space or all the three spaces. Since the projection matrices $W_1$ and $W_2$ have been obtained after the above optimization, we can compute the embeddings of a visual feature $x_i$ in latent space and visual space, and which can be represented as $W_1^T x_i$ and $W_2 W_1^T x_i$, respectively. Similarly, the embeddings of a semantic attribute $a_j$ in latent space and visual space can be represented as $W_2^T a_j$ and $W_1 W_2^T a_j$, respectively. Therefore, we can make classification in all three spaces. First, we define the classification probabilities in each space as follows,

$$
\begin{align*}
    p_{ij}^v &= e^{-d(x_i, W_1 W_2^T a_j)} \\
    p_{ij}^l &= e^{-d(W_1^T x_i, W_2^T a_j)} \\
    p_{ij}^a &= e^{-d(W_2 W_1^T x_i, a_j)}
\end{align*}
$$

where $d(\cdot, \cdot)$ is the Euclidean distance. We select label of the highest probability in Eq. 35 as the final result, which can be represented as,

$$
    c = \arg \max_{j=1, \ldots, q} (\max(p_{ij}^v, p_{ij}^l, p_{ij}^a)).
$$

Eq. 35 and Eq. 36 are defined for conventional ZSL and transductive ZSL, and they can be extended to GZSL by replacing $a_j^v$ to $a_j \in A^v \cup A^l$.

IV. Experiments

A. Datasets

In our experiments, we employ four popular benchmark datasets, i.e., SUN (SUN attribute) [35], CUB (Caltech-UCSD-Birds 200-2011) [36], AWA(Animals with Attributes) [4] and aPY(Attribute Pascal and Yahoo) [37]. SUN is a type of fine-grained dataset, which contains many different visual scenes and CUB is also a fine-grained dataset, and it is consisted of 200 bird-species. AWA is a coarse-grained dataset of 50 classes of animals. APY has 20 classes from Pascal VOC [38] for training and 12 classes from Yahoo [37] for test. The other details of these datasets can be found in Tab. 1, where ‘SS’ refers to number of Seen Samples in training, ‘TS’ is the number of samples from unseen classes for test, and ‘TR’ is for seen ones. In addition, we adopt the split strategy which is proposed by [25].

B. Experimental Setting

We employ the extracted deep features with ResNet [2] for all the four datasets as our input, and the attributes and class split strategy are the same as that in [25]. Since there are many hyper-parameters in our methods, we adopt the cross validation to choose them. Here, we emphasize the difference of ZSL cross-validation to conventional machine learning approaches. Compared to inner-splits of training samples within each class, ZSL problem requires inter-splits by in turn regarding part of seen classes as unseen. In addition, when initializing $W_1$ and $W_2$ in the first step, the ratio of the optimal parameters of $\lambda_2, \lambda_3$ and $\lambda_4$ is fixed for the next step, thus the cross validation for the second step only contains three hyper-parameters, including $\lambda_1, \lambda_5$ and the ratio of them to the item in the first step. In transductive setting, the ratio of $\beta_1, \beta_2, \beta_3$ and $\beta_5$ is set to be the same as that of $\lambda_1, \lambda_2, \lambda_3$ and $\lambda_5$, which can greatly reduce the search time for cross validation. Besides, the initial value of $\mu$ is 0.1, and the $\rho$ is set as 0.1 too. The three parameters of the iteration times are all set as 80. The core part of the source code and more experimental results can be found in the supplementary material.

C. Results on ZSL

In this subsection, we test our method on the setting of conventional ZSL setting on all four datasets listed in Tab. 1, and report the results in Tab. 2. Concretely, besides the baseline results recorded in [25], we also provide the results of four other methods, including TVN [46], VZSL [47], LESAE [13] and LESD [11], among which LESAE and LESD are low...
TABLE 3. Comparison with state-of-the-art baselines on GZSL setting. '-' means not reported.

| Method  | SUN | CUB | AWA | aPY |
|---------|-----|-----|-----|-----|
|         |  ts | tr  | H   | ts  | tr  | H  | ts  | tr  | H  |
| DAP [4] | 4.2 | 23.1| 7.5 | 1.7 | 67.9| 3.3 | 0.0 | 88.7| 0.0 |
| CONSE [17]| 6.8 | 39.9| 11.6| 1.6 | 72.2| 3.1 | 0.4 | 88.6| 0.8 |
| CMT [39] | 8.1 | 21.8| 11.8| 7.2 | 49.8| 12.6| 0.9 | 87.6| 1.8 |
| SIE [18] | 2.1 | 36.4| 4.0 | 8.5 | 46.9| 14.4| 7.0 | 80.5| 12.9|
| LATEM [40]| 14.7 | 28.8 | 19.5 | 15.2 | 57.3 | 24.0 | 7.3 | 71.7 | 13.3 |
| ALE [41] | 21.8 | 33.1 | 26.3 | 23.7 | 62.8 | 34.4 | 16.8 | 76.1 | 27.5 |
| DEVISE [42] | 16.9 | 27.4 | 20.9 | 23.8 | 53.0 | 32.8 | 13.4 | 68.7 | 22.4 |
| SIE [18] | 14.7 | 30.5 | 19.8 | 23.5 | 59.2 | 33.6 | 11.3 | 74.6 | 19.6 |
| ESSL [43] | 11.0 | 27.9 | 15.8 | 12.6 | 63.8 | 21.0 | 6.6 | 75.6 | 12.1 |
| SAE [12] | 8.8 | 18.0 | 11.8 | 7.8 | 54.0 | 13.6 | 1.8 | 77.1 | 3.5 |
| SYNC [44]| 7.0 | 43.3 | 13.4 | 11.5 | 70.9 | 19.8 | 8.9 | 87.3 | 16.2 |
| PRESERVE [49]| 20.8 | 37.2 | 26.7 | 24.6 | 54.3 | 33.9 | - | - | - |
| CDL [50] | 21.5 | 34.7 | 26.5 | 23.5 | 55.2 | 32.9 | 28.1 | 73.5 | 40.6 |
| GFZSL [45] | 0.0 | 39.6 | 0.0 | 0.0 | 45.7 | 0.0 | 1.8 | 80.3 | 3.5 |
| LAGO [51] | 18.8 | 33.1 | 23.9 | 21.8 | 73.6 | 33.7 | 23.8 | 67.0 | 35.1 |
| PSEUDO [52] | 19.0 | 32.7 | 24.0 | 23.0 | 51.6 | 31.8 | 22.4 | 80.6 | 35.1 |
| KERNEL [53] | 21.0 | 31.0 | 25.1 | 24.2 | 63.9 | 35.1 | 18.3 | 79.3 | 29.8 |
| TVN [46] | 18.2 | 28.9 | 22.3 | 21.6 | 47.5 | 29.7 | 18.2 | 87.5 | 30.2 |
| VZSL [47] | 15.2 | 23.8 | 18.6 | 17.1 | 37.1 | 23.8 | 22.3 | 77.5 | 34.6 |
| LESAE [13] | 21.9 | 34.7 | 26.9 | 24.3 | 53.0 | 33.3 | 19.1 | 70.2 | 30.0 |
| LESD [11] | 15.2 | 19.8 | 17.2 | 14.6 | 38.5 | 21.2 | 12.6 | 71.0 | 21.4 |
| Our Method | 22.3 | 35.2 | 27.3 | 28.2 | 51.1 | 36.3 | 36.0 | 76.5 | 48.8 |

D. RESULTS ON GZSL

Since GZSL is more realistic and challenging, we add more state-of-the-art methods into comparison, such as PRESERVE [49], CDL [50], LAGO [51], PSEUDO [52] and KERNEL [53]. Besides, due to the fact that the transductive setting assumes the test data belongs to the unseen classes, which conflicts with the definition of GZSL, we do not conduct experiment on this setting. The results of our method and the baselines are all recorded in Tab. 3. From this table, it can be seen that our method can outperform all the other methods on the metrics of ts and H, especially on AWA and aPY that we can obtain 8.2% and 10.3% respectively on H comparing to the best method CDL. All the baseline methods have high value on tr but perform bad on ts, and thus leads to bad performance on H, which means that these methods are over-fitting on seen classes, while our method can obtain a balance between seen classes and unseen classes. The main reason of such phenomenon is caused by the adaption of orthogonal constraint, which processes the attributes of both the seen classes and the unseen classes simultaneously.

E. ABLATION STUDY

1) THE EFFECTS OF RPCA AND RELAXATION

In this experiment, we will discuss two cases. The first one is that whether the prototype relaxation strategy adopted in our method can really improve the performance, and the second one is whether the sparse constraint on E is effective. For the first one, we remove \( \|L\|_* + \lambda_1 \|E\|_1 \) from Eq. 6, and optimize the remaining part with same parameters as that used in Tab. 2 and Tab. 3. The results of three metrics on four datasets are shown in Fig. 3. We can find that the performance without \( E \) is between that with RPCA and without RPCA, which shows that the low rank constraint on \( L \) is effective and the sparse constraint can further improve the performance.
2) THE EFFECTS OF AUTO-ENCODER (AE) AND ORTHOGONAL CONSTRAINT (OC)

Since we have employed AE and OC modules in our method to improve the performance, it is necessary to evaluate their contributions to the final result. To verify the effect of AE, we remove the $\lambda_3$ part of Eq. 4 from the final loss function of Eq. 6, re-optimize it and conduct the same experiments on the four datasets. Similarly, we also remove Eq. 5 from Eq. 6, and conduct the same operations as those for AE. The experimental results on both ZSL and GZSL are recorded in Tab. 4. From this table, it can be clearly discovered that when either AE or OC is removed, the performance is significant decreased, and when both of them are excluded, the final results decline to near half of their original values, especially on GZSL. Therefore, this phenomenon reveals that both AE and OC play important roles in improving the performance.

F. DIMENSION OF LATENT SPACE

In this section, we conduct experiment on four datasets to show how does the dimension of the latent space influence the final performance. The results are illustrated in Fig. 5, where the X-axis stands for the dimension of the latent space, and it is represented with the multiple of the number of categories. Y-axis denotes $ts$ on ZSL in Fig. 5(a) and $H$ on GZSL in Fig. 5(b). It can be clearly discovered that there is a common property on the four datasets: both ZSL and GZSL can reach the best performance when the dimension is about two times to the number of categories, and it is lower when the dimension is increased or decreased. Since it is known that there is at least the same dimension as the category number to make all classes to be orthogonal to each other, the dimension should be larger than the category number. Besides, too high dimension may bring too much redundant information and lead to performance degradation.

G. CONVERGENCE ANALYSIS

Since the proposed method is optimized with an iterative strategy, it is necessary to show whether it can converge. In this subsection, we illustrate the convergence curves on four datasets in Fig. 6. From this figure, it can be clearly find that the proposed method can converge when the iterative number is larger than 80. To be specific, the convergence...
speed on SUN is a litter slower than that on other three datasets, which is cause by that SUN has much more categories than the others, so it need more iteration times to find feasible projection model to disperse all the classes.

**H. DISTRIBUTION IN LATENT SPACE**

The objective of orthogonal constraint in latent space is to disperse all classes and make them more discriminative, and it is the reason why the performance of our method can outperform the baselines, especially on GZSL. To have a more intuitive understanding, we employ t-SNE [54] on AWA to illustrate the distributions of the samples of unseen classes in this space. Meanwhile, we also show the distributions of the original visual features, the embedded vectors in attribute space with LESD, and the projected embeddings in attribute space with LESAE. The distribution maps are illustrated in Fig. 7. Our method can obtain the best distribution, and LESD plays the worst, which is caused by that our method can disperse all the unseen classes while the other two methods cannot. LESAE is better than LESD because LESAE applies SAE to reconstruct visual features, which encourage the embeddings to preserve the characteristic of visual features and make the distribution of them to be similar as that of original features.

**TABLE 5. The mean average precision (mAP) for zero shot image retrieval.**

| Methods  | SUN  | CUB  | AWA  | aPY  | Average |
|----------|------|------|------|------|---------|
| SSE [18] | 58.9 | 4.7  | 46.25| 15.4 | 31.3    |
| JSLE [56]| 76.5 | 23.9 | 66.5 | 32.7 | 49.9    |
| Sync [44]| 74.3 | 34.3 | 65.4 | 30.4 | 51.1    |
| ISEF [57]| 52.7 | 25.3 | 68.1 | 36.9 | 45.8    |
| MFMR [58]| 77.4 | 30.6 | 70.8 | 45.6 | 56.2    |
| LESD [11]| 76.6 | 31.3 | 71.2 | 40.3 | 54.9    |
| GSDL [55]| 79.2 | 34.2 | 73.6 | 44.8 | 58.0    |
| Ours     | 80.6 | 40.0 | 74.7 | 45.8 | 60.3    |

**FIGURE 7. The distributions of our method and other two state-of-the-art methods.**

**FIGURE 8. The top-5 retrieval results on AWA.**
confusion between similar classes. Furthermore, vectors from latent space are exploited to reconstruct visual features and semantic attributes respectively to alleviate the domain shift problem. In addition, the hubness problem is also mitigated by applying the max probability model in the combined three spaces. Extensive experiments on four dataset are conducted and the results show the superiority of the proposed method.

REFERENCES

[1] H. Zhang, Y. Long, and L. Shao, “Zero-shot hashing with orthogonal projection for image retrieval,” Pattern Recognit. Lett., vol. 117, pp. 201–209, Jan. 2019.

[2] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 770–778.

[3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2009, pp. 248–255.

[4] C. H. Lampert, H. Nickisch, and S. Harmeling, “Learning to detect unseen object classes by between-class attribute transfer,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2009, pp. 951–958.

[5] H. Zhang, Y. Long, and L. Shao, “Zero-shot learning and hashing with binary visual similars,” Multimedia Tools Appl., vol. 78, no. 17, pp. 24147–24165, Sep. 2019.

[6] C. Tang, X. Yang, J. Lv, and Z. He, “Zero-shot learning by mutual information estimation and maximization,” Knowl.-Based Syst., vol. 194, Apr. 2020, Art. no. 105490.

[7] Y. Wang, H. Zhang, Z. Zhang, Y. Long, and L. Shao, “Learning discriminative domain-invariant prototypes for generalized zero shot learning,” Knowl.-Based Syst., vol. 1966, Mar. 2020, Art. no. 105797.

[8] Z. Ji, K. Chen, J. Wang, Y. Yu, and Z. Zhang, “Multi-modal generative adversarial network for zero-shot learning,” Knowl.-Based Syst., vol. 197, Jun. 2020, Art. no. 105847.

[9] H. Zhang, J. Liu, Y. Yao, and Y. Long, “Pseudo distribution on unseen classes for generalized zero shot learning,” Pattern Recognit. Lett., vol. 135, pp. 451–458, Jul. 2020.

[10] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in Proc. Annu. Conf. Neural Inf. Process. Syst., 2014, pp. 2672–2680.

[11] Z. Ding, M. Shao, and Y. Fu, “Low-rank embedded ensemble semantic dictionary for zero-shot learning,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 2050–2058.

[12] E. Kodirov, T. Xiang, and S. Gong, “Semantic AutoEncoder for zero-shot learning,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 3174–3183.

[13] Y. Liu, Q. Gao, J. Li, J. Han, and L. Shao, “Zero-shot learning via low-rank embedded semantic AutoEncoder,” in Proc. 27th Int. Joint Conf. Artif. Intell., Jul. 2018, pp. 2490–2496.

[14] L. Niu, A. Veeraraghavan, and A. Sabharwal, “Weably supervised learning meets zero-shot learning: A hybrid approach for fine-grained classification,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 7171–7180.

[15] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid, “Label-embedding for image classification,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 38, no. 7, pp. 1425–1438, Jul. 2016.

[16] Z. Akata, S. Reed, D. Walter, H. Lee, and B. Schiele, “Evaluation of output embeddings for fine-grained image classification,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2015, pp. 2927–2936.

[17] M. Norouzi, T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G. S. Corrado, and J. Dean, “Zero-shot learning by convex combination of semantic embeddings,” in Proc. Int. Conf. Learn. Represent., 2014, pp. 1–9.

[18] Z. Zhang and V. Saligrama, “Zero-shot learning via semantic similarity embedding,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Dec. 2015, pp. 4166–4174.

[19] Y. Long, L. Liu, L. Shao, F. Shen, G. Ding, and J. Han, “From zero-shot learning to conventional supervised classification: Unseen visual data synthesis,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 1627–1636.

[20] H. Zhang, Y. Long, L. Liu, and L. Shao, “Adversarial unseen visual feature synthesis for zero-shot learning,” Neurocomputing, vol. 329, pp. 12–20, Feb. 2019.

[21] Y. Xian, T. Lorenz, B. Schiele, and Z. Akata, “Feature generating networks for zero-shot learning,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2015, pp. 2927–2936.

[22] Y. Xian, Z. Akata, G. Sharma, Q. Nguyen, M. Hein, and B. Schiele, “Latent embeddings for zero-shot classification,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 69–77.

[23] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid, “Label-embedding for attribute-based classification,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2013, pp. 819–826.

[24] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. A. Ranzato, and T. Mikolov, “Device: A deep visual-semantic embedding model,” in Proc. Annu. Conf. Neural Inf. Process. Syst., 2013, pp. 2121–2129.

[25] B. Romera-Paredes and P. Torr, “An embarrassingly simple approach to zero-shot learning,” in Proc. Int. Conf. Mach. Learn., 2015, pp. 2152–2161.
[44] S. Changpinyo, W.-L. Chao, B. Gong, and F. Sha, “Synthesized classifiers for zero-shot learning,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 5327–5336.

[45] V. K. Verma and P. Rai, “A simple exponential family framework for zero-shot learning,” in Proc. Eur. Conf. Mach. Learn. Princ. Pract. Knowl. Discovery Databases. Springer, 2017, pp. 792–808.

[46] H. Zhang, Y. Long, Y. Guan, and L. Shao, “Triple verification network for generalized zero-shot learning,” IEEE Trans. Image Process., vol. 28, no. 1, pp. 506–517, Jan. 2019.

[47] W. Wang, Y. Pu, V. K. Verma, K. Fan, Y. Zhang, C. Chen, P. Rai, and L. Carin, “Zero-shot learning via class-conditioned deep generative models,” in Proc. AAAI Conf. Artif. Intell., 2018, pp. 1–8.

[48] J. Song, C. Shen, Y. Yang, Y. Liu, and M. Song, “Transductive unbiased embedding for zero-shot learning,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 1024–1033.

[49] S. Biswas and Y. Amnadani, “Preserving semantic relations for zero-shot learning,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 7603–7612.

[50] H. Jiang, R. Wang, S. Shan, and X. Chen, “Learning class prototypes via structure alignment for zero-shot recognition,” in Proc. Eur. Conf. Comput. Vis., Sep. 2018, pp. 118–134.

[51] Y. Atzmon and G. Chechik, “Probabilistic and-or attribute grouping for zero-shot learning,” in Proc. UAI, 2018, pp. 1–11.

[52] T. Long, X. Xu, Y. Li, F. Shen, J. Song, and H. T. Shen, “Pseudo transfer with marginalized corrupted attribute for zero-shot learning,” in Proc. ACM Multimedia Conf. Multimedia Conf. MM, 2018, pp. 1802–1810.

[53] H. Zhang and P. Koniusz, “Zero-shot kernel learning,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 7670–7679.

[54] L. V. D. Maaten and G. Hinton, “Visualizing data using t-SNE,” J. Mach. Learn. Res., vol. 9, pp. 2579–2605, Nov. 2008.

[55] Z. Zhang and V. Saligrama, “Zero-shot learning via joint latent similarity embedding,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 6034–6042.

[56] M. Bucher, S. Herbin, and F. Jurie, “Improving semantic embedding consistency by metric learning for zero-shot classification,” in Proc. Eur. Conf. Comput. Vis., 2016, pp. 6034–6042.

[57] X. Xu, F. Shen, Y. Yang, D. Zhang, H. T. Shen, and J. Song, “Matrix tri-factorization with manifold regularizations for zero-shot learning,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 3798–3807.

**YONG MA** received the Ph.D. degree from the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China, in 2015. He is currently an Assistant Professor with the School of Computer Science and Engineering, Nanjing University of Science and Technology. His research interests include machine learning, computer vision, and robotics.

**HUAQI MAO** received the B.S. degree in software engineering from the Chongqing University of Technology, Chongqing, China, in 2017. He is currently pursuing the master’s degree with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China. His current research interests include image classification and deep learning.

**HAOFENG ZHANG** received the B.Eng. and Ph.D. degrees from the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China, in 2003 and 2007 respectively. From December 2016 to December 2017, he was an Academic Visitor with the University of East Anglia, Norwich, U.K. He is currently a Professor with the School of Computer Science and Engineering, Nanjing University of Science and Technology. His research interests include computer vision and robotics.

**WENBO WANG** is currently pursuing the master’s degree with the School of Information Engineering, Nanjing Audit University. His main research interests include image recognition and robotics.

---

**YONG MA** received the Ph.D. degree from the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China, in 2015. He is currently an Assistant Professor with the School of Computer Science and Engineering, Nanjing University of Science and Technology. His research interests include machine learning, computer vision, and robotics.

**HUAQI MAO** received the B.S. degree in software engineering from the Chongqing University of Technology, Chongqing, China, in 2017. He is currently pursuing the master’s degree with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China. His current research interests include image classification and deep learning.

**HAOFENG ZHANG** received the B.Eng. and Ph.D. degrees from the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China, in 2003 and 2007 respectively. From December 2016 to December 2017, he was an Academic Visitor with the University of East Anglia, Norwich, U.K. He is currently a Professor with the School of Computer Science and Engineering, Nanjing University of Science and Technology. His research interests include computer vision and robotics.

**WENBO WANG** is currently pursuing the master’s degree with the School of Information Engineering, Nanjing Audit University. His main research interests include image recognition and robotics.