Matrix Tri-Factorization with Manifold Regularizations for Zero-shot Learning

Xing Xu, Fumin Shen, Yang Yang, Dongxiang Zhang, Heng Tao Shen* and Jingkuan Song
Center for Future Media & School of Computer Science and Engineering
University of Electronic Science and Technology of China, China

Abstract

Zero-shot learning (ZSL) aims to recognize objects of unseen classes with available training data from another set of seen classes. Existing solutions are focused on exploring knowledge transfer via an intermediate semantic embedding (e.g., attributes) shared between seen and unseen classes. In this paper, we propose a novel projection framework based on matrix tri-factorization with manifold regularizations. Specifically, we learn the semantic embedding projection by decomposing the visual feature matrix under the guidance of semantic embedding and class label matrices. By additionally introducing manifold regularizations on visual data and semantic embeddings, the learned projection can effectively capture the geometrical manifold structure residing in both visual and semantic spaces. To avoid the projection domain shift problem, we devise an effective prediction scheme by exploiting the test-time manifold structure. Extensive experiments on four benchmark datasets show that our approach significantly outperforms the state-of-the-arts, yielding an average improvement ratio by 7.4% and 31.9% for the recognition and retrieval task, respectively.

1. Introduction

Conventional visual recognition systems usually require an enormous amount of manually labeled training data to achieve good classification accuracy, typically several thousand images for each class to be learned [8, 40]. Due to the ever increasing number of available images and categories to be recognized, it then becomes infeasible to label images for each possible class. For example, this problem is essentially crippling when classifying fine-grained object classes [9] such as species of animals or brands of consumer products, since the number of labeled images for these classes may be far from sufficient to directly build high-quality classifiers.

Zero-shot learning (ZSL) [19, 17] has long been believed to hold the key to the above problem. ZSL aims to recognize new-coming instances (*e.g.*, images) of unseen classes with only labeled instances of seen classes that are available for training. Without labeled examples, classifiers for unseen classes are obtained by transferring knowledge learned from the seen classes. This is typically achieved by exploring a semantic embedding space where the seen and unseen classes can be related in. The spaces used by most existing works are based on attributes [11, 17, 27, 41] and word2vec representations [12, 23, 24, 33]. In such spaces, each class name can be represented by a high dimensional binary/continuous vector based on a pre-defined attribute ontology, or a vast unannotated text corpus by natural language processing.

Given a semantic embedding space, the semantic relation between an unseen class and each seen class can be measured as the distance between their semantic embedding vectors. However, as a test image is represented by a visual feature vector, its similarity to unsee classes cannot be obtained by directly measuring it with the semantic embedding vectors of unseen classes. To solve this problem, several existing ZSL approaches [1, 12, 33, 31, 42, 6, 21] rely on directly learning a projection function between the visual feature space and the semantic embedding space from the labeled images of seen classes. Then the prediction for a test image can be performed by mapping the visual feature via the projection function and measuring similarity with the unseen classes in the semantic embedding space.

However, these projection based methods still have several primary shortcomings. Firstly, the intrinsic manifold structure residing in both the visual feature space and the semantic embedding space are not well explored when learning the projection function. Secondly, these methods suffer from the projection domain shift problem [13, 16, 37], i.e. the visual feature mapping (projection function) learned from the seen class data may not generalize well to the unseen class data. The main reason is that the test data distribution of unseen classes in the projection space could be different from the estimation obtained by the learned projection based on the training data of seen classes. Thirdly, existing projection-based approaches still have a large gap with the ideal performance under the generalized ZSL setting [7], where test data are from both seen and unseen classes and they are required to be predicted into the joint

*Corresponding author: Heng Tao Shen.

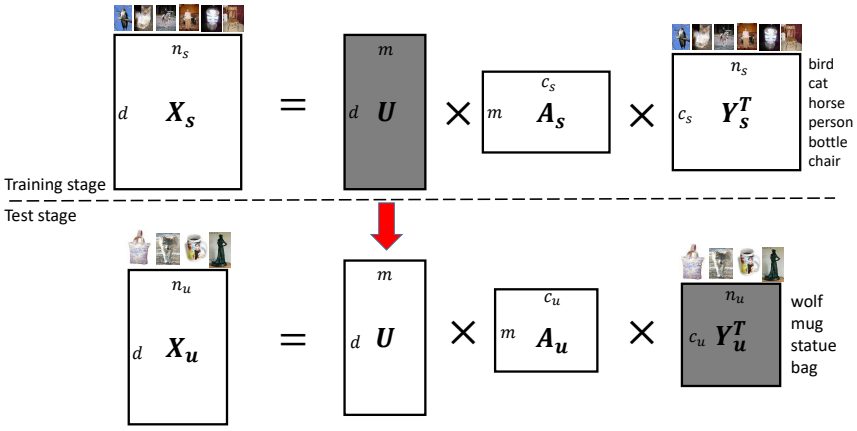


Figure 1. The proposed MFMR framework for ZSL. Note that the white blocks are observed matrices, while the gray ones are unknown matrices to be learned. At the training stage, we factorize the visual feature matrix $\mathbf{X}_s$ of seen instances by the semantic embedding matrix $\mathbf{A}_s$ and label matrix $\mathbf{Y}_s$ of seen classes and the latent projection matrix. At test stage, we use learned projection $\mathbf{U}$, together with the semantic embedding matrix $\mathbf{A}_u$ of unseen class, to inference the label matrix $\mathbf{Y}_u$ for the test instances by decomposing $\mathbf{X}_u$. The latent variables are constrained with manifold regularizations, which is essentially different from the work in [31].

label space of both types of classes.

In this paper, we tackle the aforementioned problems in existing projection-based ZSL methods by developing a novel approach, termed Matrix tri-Factorization with Manifold Regularizations (MFMR), as illustrated in Figure 1. Specifically, at training stage, MFMR learns a projection matrix by decomposing the visual feature matrix of the training instances to three matrices, among which two are explicitly provided, i.e. the matrices of semantic embeddings and class labels of seen classes. This constraint ensures the learned projection matrix by MFMR effectively constructs the mapping from the visual feature space to the semantic embedding space with the prior supervised information provided by the two observed matrices. Meanwhile, two manifold regularizers that model the manifold structure of both the visual feature space and the attribute space are integrated to the factorization procedure, which enhances the capability of the learned projection matrix on preserving the geometric structures of the training data in two spaces. At test stage, MFMR directly estimates the class label matrix of all test instance jointly via an effective prediction scheme. In particular, given the observed projection matrix (learned at the training stage) and the semantic embedding matrix of unseen classes, MFMR performs similar factorization procedure on the visual feature matrix of the test instances while further exploiting the manifold structure in them, hence overcomes the projection domain shift problem.

The main contributions of our work are three-folds:

- We propose a novel ZSL approach, termed MFMR, by utilizing the matrix tri-factorization framework with manifold regularizations on visual feature and semantic embedding spaces.
- We develop an effective prediction scheme for MFMR to jointly estimate the class labels of all test instances, where the beneficial manifold structure of test data is well exploited for performance improvement.
- We conduct extensive experiments on four benchmark ZSL datasets, which validate the superiority of MFMR over state-of-the-art approaches on zero-shot recognition and retrieval task. The robustness of MFMR on balancing the prediction for seen and unseen classes is also verified in additional evaluation under the generalized ZSL setting.

The remainder of the paper is organized as follows. In the next section, we briefly review the related methods for ZSL. Then, we introduce our approach, followed by experimental results with comprehensive analysis on four benchmark datasets. Finally, we draw the conclusion.

2. Related Work

Existing ZSL methods differ in how to transfer the knowledge from seen to unseen classes. Given the semantic embeddings of classes, most existing approaches are grouped into similarity based and projection based methods. The similarity based approaches [30, 25] rely on learning an n-way discrete classifier for the seen classes in the visual feature space, and then use it to compute the visual similarity between an image of unseen class to those of the seen classes. In contrast, the projection based approaches first map the visual features of test instances to the semantic space, and then determine the relatedness of unseen classes and test instances by various semantic relatedness measures [17, 1, 14, 42]. Specifically, with available semantic embedding of classes, Akata *et al*. [1] proposed a mod-

el that implicitly projects the visual features and semantic embeddings onto a common space where the compatibility between any pair of them can be measured. In [31], a simpler and efficient linear model with a principled choice of regularizers was proposed to obtain much better results under the same principle. Our work also seeks effective projection by decomposing the visual feature matrix under the guidance of semantic embedding and class label matrices. The decomposition is accomplished based on matrix tri-factorization, which is different from these methods. Like the recent works [38, 20, 29, 14, 37, 6] that addressed the importance of modeling the intrinsic manifold structure, our work integrates two manifold regularizers to account for the geometric information underlying the visual feature and semantic embedding spaces. In general, our work empirically shows more accurate predictions with high efficiency.

Recently Fu *et al*. [13] addressed the projection domain shift problem that potentially exists in the projection based methods, and they proposed a transductive multi-view embedding framework to solve this problem. Kodirov *et al*. [16], Zhang and Saligrama [44] further studied this problem and proposed to exploit the unseen class data structure in the learning procedure via unsupervised domain adaptation scheme and structured prediction scheme, respective. Our method also mines the test-time data information for performance improvement. However, we should point out that compared with the above methods, our method has no access to the unseen class data during training, thus it is more practical for the problem setting of ZSL.

To evaluate the models generated by ZSL, most existing ZSL approaches [26, 39, 15, 22, 43, 6] adapt the settings in the seminal work of Lampert *et al*. [17], and focus on discriminate among unseen classes without the instances of seen classes during the test stage. This setting may be unrealistic since in the real world, it is common to encounter instances in both seen and unseen classes during the test stage. Recently, Chao *et al*. [7] advocated a generalized ZSL setting, where models generated by ZSL need to predict test data from both seen and unseen classes in their joint label space. This generalized setting is able to provide more objective evaluation. We evaluate our method under both settings and the results shows the robustness of our method on trading off between recognizing test data from seen classes and unseen classes.

3. The Proposed Approach

3.1. Problem Statement

Let $\mathcal{S}$ denotes a set of c_s seen classes and $\mathcal{U}$ a set of c_u unseen classes. These two sets of labels are disjoint, i.e. $\mathcal{S} \cap \mathcal{U} = \emptyset$. Each of the classes in the two sets is represented by an m-dimensional semantic embedding (*e.g.*, attributes) vector. These semantic embeddings of seen and unseen classes can be denoted in matrices $\mathbf{A}_s = \{\mathbf{a}_i^s\}_{i=1}^{c_s}$ and $\mathbf{A}_u = \{\mathbf{a}_j^u\}_{j=1}^{c_u}$, where $\mathbf{a}_i$ and $\mathbf{a}_j$ are the vectors for i-th seen class and j-th unseen class, respectively. Suppose we are given a training set $\mathcal{D}_S = \{(\mathbf{x}_i^s, \mathbf{y}_i^s)\}_{i=1}^{n_s}$, where for the i-th labeled image, $\mathbf{x}_i^s$ denotes its d-dimensional feature vector and $\mathbf{y}_i^s$ is the one-hot class label vector with the label belongs to $\mathcal{S}$. Besides, a test set $\mathcal{D}_u = \{(\mathbf{x}_j^u, \mathbf{y}_j^u)\}_{j=1}^{n_u}$ is provided, where $\mathbf{x}_j^u$ is also a d-dimensional feature vector extracted from the j-th unlabeled test image and $\mathbf{y}_j^u$ is the to-be-predicted class label vector with the label from $\mathcal{U}$. For simplicity, we denote the indices of the training and test sets as $\mathcal{I} = \{s, u\}$.

Generally, ZSL is inherently a two stage process: training and test. At the training stage, knowledge of seen classes is learned from the data of $\mathbf{X}_s$, $\mathbf{A}_s$ and $\mathbf{Y}_s$. Then at the test stage, the learned knowledge is transferred to unseen classes to predict $\mathbf{Y}_u$ given $\mathbf{X}_u$ and $\mathbf{A}_u$.

3.2. The General Framework of MFMR

The main idea behind our approach MFMR is shown by the diagram in Figure. 1. At the training stage, we learn a projection from the labelled training instances consisting of seen classes only, under the matrix tri-factorization framework with manifold regularizers. At the test stage, the class labels of test instances are jointly predicted by exploiting the manifold structure residing in them.

As a projection constructs the map between the visual feature space and the semantic embedding space which further bridges for knowledge transfer from seen classes to unseen ones, we assume that an effective projection needs to 1) *maximize the empirical likelihood of the visual features of both training and test instances*; and 2) *preserve the geometric manifold structure residing in both visual feature space and semantic embedding space*.

3.2.1 Learning the Projection

To meet the first requirement, in MFMR, we propose to learn a projection that serves as the common latent factors underlying both training and test data. To achieve this goal, we customize the matrix tri-factorization [10] framework to the visual feature matrix $\mathbf{X}_s$ of the labeled training instances from seen classes. The factorization procedure performs feature-instance co-clustering to estimate the empirical likelihood of $\mathbf{X}_s$, resulting in three matrices that minimize the estimation error as

$$\min_{\mathbf{U},\mathbf{V}_s} \|\mathbf{X}_s - \mathbf{U}\mathbf{A}_s\mathbf{V}_s^\top\|^2, \tag{1}$$

where $\|\cdot\|^2$ is the Frobenius norm of a matrix. In Eq. 1, $\mathbf{U} = \{\mathbf{u}_i\}_{i=1}^m \in \mathbb{R}^{d\times m}$ is the projection, each $\mathbf{u}_i$ represents a visual feature cluster for each semantic embedding. $\mathbf{V}_s = \{\mathbf{v}_i\}_{i=1}^{c_s} \in \mathbb{R}^{n_s\times c_s}$, each $\mathbf{v}_i$ represents an instance cluster for each seen class (i.e. the instances of similar semantics would lie in the same cluster). These two matrices are the co-clustering results on the row vectors (fea-

tures) and column vectors (instances) of $\mathbf{X}_s$ respectively. The third matrix, i.e. the semantic embeddings $\mathbf{A}_s$ of seen classes is introduced to associate $\mathbf{U}$ and $\mathbf{V}_s$. The advantage of using observed $\mathbf{A}_s$ as the bridge is that the mapping between visual features and seen classes can be implicitly constructed. Similarly, the mapping at test stage can be accomplished when using semantic embeddings of unseen classes. It is notable that with the class label matrix $\mathbf{Y}_s$ of training instances from seen classes, the instance clusters for seen classes can be directly obtained. Therefore, a rational strategy is to enforce $\mathbf{V}_s = \mathbf{Y}_s$ to ensure the instance clusters decomposed from $\mathbf{X}_s$ to be consistent with the one obtained from $\mathbf{Y}_s$.

3.2.2 Modeling the Manifold Structure

For the second requirement, to preserve the manifold structure, we separately consider the visual feature matrix $\mathbf{X}_s$ in terms of instance space (per column) and feature space (per row). The goal is to encode into the projection matrix the underlying geometrical information of these spaces.

Consider two instances $\mathbf{x}^s_{*i}$, $\mathbf{x}^s_{*j}$ from $\mathbf{X}_s$ (i.e. two column vectors). If they are close on the intrinsic data manifold, then their instance clusters should be close as well (i.e. belong to the same class). Under the manifold assumption [5], the geometric structure can be modeled by a nearest neighbor graph in the instance space. Consider an instance graph G^I_s with n_s vertices (instances). Then the affinity matrix $\mathbf{W}^I_s \in \mathbb{R}^{n_s \times n_s}$ in G^I_s can be defined based on the visual similarities of pairwise instances [3], as

$$(\mathbf{W}^I_s)_{ij} = \begin{cases} cos(\mathbf{x}^s_{*i}, \mathbf{x}^s_{*j}) & \mathbf{x}^s_{*i} \in N_k(x^s_{*j}),\ or\ \mathbf{x}^s_{*j} \in N_k(x^s_{*i}) \\ 0, & otherwise \end{cases} \tag{2}$$

where $N_k(x)$ denotes the top-k nearest neighbors of the ith instance $\mathbf{x}^s_{*i}$. By denoting $\mathbf{Q}^I_s = diag(\sum_i (\mathbf{W}^I_s)_{ij})$, then preserving the visual feature manifold structure in D_s is derived to minimize the instance manifold regularizer

$$R^I_s = \frac{1}{2}\sum_{ij} \|\mathbf{v}_i - \mathbf{v}_j\|^2 (\mathbf{W}^I_s)_{ij} = tr(\mathbf{V}_s^\top (\mathbf{Q}^I_s - \mathbf{W}^I_s)\mathbf{V}_s). \tag{3}$$

Then consider each dimension of the features in $\mathbf{X}_s$ (i.e. each row), they are assumed to be sampled from a distribution supported by the semantic embedding space [29]. Therefore, we construct a feature graph G^F_s with d vertices and each representing a feature in set D_s. Similarly, the affinity matrix $\mathbf{W}^F_s \in \mathbb{R}^{d \times d}$ in G^F_s can be defined as

$$(\mathbf{W}^F_s)_{ij} = \begin{cases} cos(\mathbf{x}^s_{i*}, \mathbf{x}^s_{j*}) & \mathbf{x}^s_{i*} \in N_p(\mathbf{x}^s_{j*}),\ or\ \mathbf{x}^s_{j*} \in N_p(\mathbf{x}^s_{i*}) \\ 0, & otherwise \end{cases} \tag{4}$$

where $N_k(\mathbf{x}^s_{i*})$ denotes the top-k nearest neighbors of the ith dimension of feature $\mathbf{x}^s_{i*}$. Let $\mathbf{Q}^F_s = diag(\sum_i (\mathbf{W}^F_s)_{ij})$, preserving the geometric structure in the $\mathbf{X}_s$ further includes minimizing the feature manifold regularizer

$$R^F_s = \frac{1}{2}\sum_{ij} \|\mathbf{u}_i - \mathbf{u}_j\|^2 (\mathbf{W}^F_s)_{ij} = tr(\mathbf{U}^\top (\mathbf{Q}^F_s - \mathbf{W}^F_s)\mathbf{U}). \tag{5}$$

As each dimension of the features is clustered onto the semantic embedding space, the feature manifold regularizer in Eq. 5 implicitly reflects the manifold structure of semantic embedding space.

3.2.3 Objective Function

By integrating the two manifold regularizers into Eq. 1, the final objective function for learning the projection in MFM-R can be formulated as

$$\begin{aligned} &\min_{\mathbf{U},\mathbf{V}_s \geq 0} \mathcal{O}_s = \|\mathbf{X}_s - \mathbf{U}\mathbf{A}_s\mathbf{V}_s^\top\|^2 + \gamma R^I_s + \lambda R^F_s, \\ &s.t. \quad \mathbf{U}^\top \mathbf{1}_d = \mathbf{1}_m, \ \ \mathbf{V}_s^\top \mathbf{1}_{n_s} = \mathbf{1}_{c_s}. \end{aligned} \tag{6}$$

where γ, λ are regularization coefficients, and $\mathbf{1}_\Delta$, $\Delta \in \{d, m, c_s\}$ is the vector of ones. The ℓ_1 normalization constraints on each column of $\mathbf{U}$ and $\mathbf{V}_s$ are used to make the optimization well-defined. It indicates that learning the projection function in Eq. 6 seamlessly incorporates the manifold structure of both the visual feature space and the semantic embedding space, underlying the co-clustering procedure on $\mathbf{X}_s$.

As discussed in Section 3.2.1, the formulation in Eq. 6 can be further simplified by setting $\mathbf{V}_s$ equal to $\mathbf{Y}_s$ as in [20]. Due to the observed $\mathbf{Y}_s$, R^I_s becomes a constant variable, thus Eq. 6 reduces to optimize the parameter $\mathbf{U}$ as

$$\begin{aligned} &\min_{\mathbf{U} \geq 0} \mathcal{O}_s = \|\mathbf{X}_s - \mathbf{U}\mathbf{A}_s\mathbf{Y}_s^\top\|^2 + \lambda R^F_s, \\ &s.t. \quad \mathbf{U}^\top \mathbf{1}_d = \mathbf{1}_m. \end{aligned} \tag{7}$$

3.2.4 Solution

We now discuss the solution to the optimization problem of Eq. 7. As Eq. 7 belongs to the constrained optimization problem, we add Lagrange functions for the parameters $\mathbf{U}$ to it, which is then formulated as

$$\min_{\mathbf{U}} \mathcal{O}_s + tr(\mathbf{\Omega}_s(\mathbf{U}^\top \mathbf{1}_d - \mathbf{1}_m)(\mathbf{U}^\top \mathbf{1}_d - \mathbf{1}_m)^\top), \tag{8}$$

where $\mathbf{\Omega}_s \in \mathbb{R}^{m \times m}$ is the Lagrange multipliers for the constraints to $\mathbf{U}$. Similar as the derivations in [20], the updating rules for $\mathbf{U}$ can be achieved by using the Karush-Kuhn-Tucker (KKT) complementarity condition [4] and setting its derivations to zero, which leads to the following updating formula:

$$\mathbf{U} \leftarrow \mathbf{U} \circ \sqrt{\frac{\mathbf{X}_s\mathbf{Y}_s\mathbf{A}_s^\top + \lambda \mathbf{W}^I_s\mathbf{U}}{\mathbf{U}\mathbf{A}_s\mathbf{Y}_s^\top\mathbf{Y}_s\mathbf{A}_s^\top + \lambda \mathbf{Q}^I_s\mathbf{U}_s}}, \tag{9}$$

where $\circ$ denotes the element-wise operations for the matrix computation.

Algorithm 1 MFMR with common prediction scheme.

Input: Matrices $\mathbf{X}_s$, $\mathbf{A}_s$ and $\mathbf{Y}_s$ from D_s, matrices $\mathbf{X}_u$, $\mathbf{A}_u$ from D_u, and parameters k, λ.
Output: $\mathbf{Y}_u$ for instances in test set $\mathcal{D}_u$.

1: Normalize $\mathbf{X}_s$ and $\mathbf{X}_u$ with ℓ_2 normalization, build feature graphs G_s^F, initialize $\mathbf{U}$ as random positive matrix and set $\mathbf{V}_s = \mathbf{Y}_s$.
2: **repeat**
3: Update $\mathbf{U}$ by Eq. 9.
4: Normalize each column of $\mathbf{U}$ by ℓ_1 normalization.
5: **until** Objective function of Eq. 7 converges.
6: Compute $\mathbf{Y}$ according to Eq. 10, given $\mathbf{X}_u$, $\mathbf{A}_u$ and $\mathbf{U}$.

Once the projection $\mathbf{U}$ is learned, at the test stage, given the visual feature vector $\mathbf{x}_j^u$ for the j-th test instance, its class label $\mathbf{y}_j^u$ can be obtained via a common prediction scheme similar to previous projection based approaches [1, 18, 16]. Specifically, for $\mathbf{x}_j^u$, its projection in the semantic embedding space be computed as $\mathbf{U}^{-1}\mathbf{x}_j^u$, which is then compared with the semantic embedding vectors $\{\mathbf{a}_u^l\}_{l=1}^{c_u}$ of unseen classes by cosine distance measure. Finally, $\mathbf{y}_j^u$ can be obtained as follows:

$$\mathbf{y}_j^u = \arg\min_l dist(\mathbf{U}^{-1}\mathbf{x}_j^u, \mathbf{a}_l^u), \quad l \in [1, c_u]. \tag{10}$$

where $dist(,)$ denotes the cosine distance metric.

Algorithm 1 summarizes the details of MFMR with this common prediction scheme. The learning procedure of MFMR primarily performs the updating rule of Eq. 9, which requires time complexity of $\mathcal{O}(dmn_sT + d^2n_s)$ with T being the number of total iterations (generally $T \leq 100$ in our experiments), Therefore, the time complexity of MFMR is linear to the number of training instances, which is efficient in practice.

3.3. Joint Prediction for Test-time Data

Due to the projection domain shift from training data to test data, the recognition performance using the common prediction scheme is suboptimal. Indeed, estimating the test-time data distribution benefits the learned projections to elaborately adapt the projected features of the test instances with the semantic embeddings of corresponding unseen classes. Specifically, we develop a joint prediction scheme, where the labels of the test instances are predicted jointly, to effectively exploit the manifold structure residing in the test data. Specifically, we factorize the visual feature matrix $\mathbf{X}_u$ of the test instances similar to the training stage in Eq. 6 as,

$$\min_{\mathbf{V}_u \geq 0} \mathcal{O}_u = \|\mathbf{X}_u - \mathbf{U}\mathbf{A}_u\mathbf{V}_u^\top\|^2 + \gamma R_u^I, \tag{11}$$
$$s.t. \quad \mathbf{V}_u^\top \mathbf{1}_{n_u} = \mathbf{1}_{c_u}.$$

The parameter to be solved in Eq. 11 is the class label matrix $\mathbf{V}_u$ of all test instances. Note that R_u^I is the instance manifold regularizer constructed from the test instances.

Algorithm 2 MFMR with joint prediction scheme.

Input: Matrices $\mathbf{X}_u$, $\mathbf{A}_u$ and $\mathbf{U}$, parameters p, γ;
Output: $\mathbf{Y}_u$ for the instances in test set $\mathcal{D}_t$.

1: Normalize $\mathbf{X}_u$ with ℓ_2 normalization, build instance graph G_u^I, initialize $\mathbf{V}_u$ as random positive matrix.
2: **repeat**
3: Update $\mathbf{V}_u$ by Eq. 13.
4: Normalize each column of $\mathbf{V}_u$ by ℓ_1 normalization.
5: **until** Objective function of Eq. 11 converges.
6: Compute $\mathbf{Y}_u$ according to Eq. 14, given $\mathbf{V}_u$.

Similarly, to solve $\mathbf{V}_u$, we also add Lagrange mutipliers $\mathbf{\Theta}_u \in \mathbb{R}^{c_u \times c_u}$ in Eq. 11 as

$$\min_{\mathbf{V}_u} \mathcal{O}_u + tr(\mathbf{\Theta}_u(\mathbf{V}_u^\top \mathbf{1}_{n_u} - \mathbf{1}_{c_u})(\mathbf{V}_u^\top \mathbf{1}_{n_u} - \mathbf{1}_{c_u})^\top). \tag{12}$$

By setting the derivation of $\mathbf{V}_u$ to zero, its updating formula is formulated as

$$\mathbf{V}_u \leftarrow \mathbf{V}_u \circ \sqrt{\frac{\mathbf{X}_u^\top \mathbf{U}\mathbf{A}_u + \gamma \mathbf{W}_u^F \mathbf{V}_u}{\mathbf{V}_u\mathbf{A}_u^\top \mathbf{U}^\top \mathbf{U}\mathbf{A}_u + \gamma \mathbf{Q}_u^F \mathbf{V}_u}}. \tag{13}$$

With the learned $\mathbf{V}_s$ at the test stage, the labels of the test instances can be obtained by selecting the entity with the largest score in each row ($\mathbf{V}_u^l$) of $\mathbf{V}_u$ as

$$\mathbf{Y}_u = \arg\max_l (\mathbf{V}_u^l), \quad l \in [1, c_u]. \tag{14}$$

Algorithm 2 depicts the details of MFMR with joint prediction scheme. At the test stage, the joint prediction scheme mainly performs the updating rule in Eq. 13, with time complexity of $\mathcal{O}(dmn_uT + dn_u^2)$ (including the nearest neighbor graph construction cost).

4. Experiments

4.1. Experimental Setup

Datasets. We use four popular benchmark datasets: 1) Animals with Attributes (AwA) [17], 2) Caltech UCSD Birds (CUB) [36], 3) aPascal-aYahoo (aPY) [11] and 4) SUN Attribute (SUN) [28] in our experiments. The datasets are diverse enough to contain coarse-grained and fine-grained categories in different domains, including animals, vehicles and natural scenes. Table 1 summarizes the statistics in each dataset. Note that we adopt the same splits of training/test instances and seen/unseen classes as in [1, 31, 6, 42, 43].

Semantic embeddings. For AwA and aPY datasets, we directly utilize the provided class-level attribute vector of continuous value. For CUB and SUN datasets which have binary attribute vector image-level, we take the mean of attribute vectors of all images from the same classes to generate class-level attribute vector.

Visual features. It has been shown in literature that the deep features work remarkably better than the low-level features as they lead to good intra-class separation [2]. Thus

Table 1. Statistics of different datasets, where "*/*" in columns of "Images" and "Classes" stands for the number of training/test images and seen/unseen classes, respectively.

Datasets	Images	Attributes	Classes
AwA	24,295 / 6,180	85	40 / 10
CUB	8,855 / 2,933	312	150 / 50
aPY	12,695 / 2,644	64	20 / 12
SUN	14,140 / 200	102	707 / 10

for all the datasets, we utilize the deep features extracted from popular CNN architecture. Specifically, we use two types of deep features: 4096-dim VGG [32] features for all datasets provided from [42] and 1024-dim GoogLeNet [34] features for AwA, CUB and SUN available from [6].

Implementation details. In our experiments, we denote the two variants of our methods according to the prediction scheme used as MFMR and MFMR-joint, which use the common and joint prediction scheme, respectively. Three model parameters in the two methods are worth investigation: the manifold regularization coefficients λ, γ, and the number k of feature and instance clusters. We report their averaged results over five runs with the optimal parameters chosen on each dataset. For each dataset, we randomly selected the images of 20% of the seen classes to formulate the validation set and used the remaining images for training. Our evaluation is rather comprehensive as we compared our methods with 10 existing ZSL methods. We not only refer to the published results but also re-evaluate some of those methods with provided implementation codes to provide objective assessment. All the experiments are conducted on a PC which has 4-core 3.3GHz CPUs with 16GB RAM.

4.2. Benchmark Comparison

4.2.1 Results on Zero-shot Recognition

The recognition task concerns the correctness of each task instance. Thus, we use accuracy to measure the overall recognition performance. Our methods are compared against various state-of-the-art alternatives: 1) the classification based method, i.e. ConSE [25]; 2) the projection based methods, i.e. DAP [17], ALE [1], ESZSL [31], SSE-INT/ReLU [42], JSLE [43] and SynC [6]; 3) the transductive method, i.e. TMV-HLP [13].

The performance of different methods, evaluated on all datasets using two different deep features, is summarized in Table 2. Generally, for our two methods and most of the counterparts, using VGG features obtains better performance than using GoogLeNet features on AwA and SUN datasets but worse on CUB. Therefore, it indicates that both two deep features have advantages relying on the specific dataset. Our MFMR performs much better than the typical projection-based approaches of ALE and ESZSL, indicating that MFMR learns more effective projection than these methods. Since MFMR incorporates the pairwise visual similarities when constructing the affine matrices for modeling manifold structure of the training data, it is also superior to the similarity-based approach of ConSE. Overall our MFMR can achieve slightly better performance on average against recent proposed methods such as SynC-struct and JSLE, showing the effectiveness of the matrix tri-factorization on learning projection and the advantage of modeling the manifold structure.

When using the joint prediction scheme, our MFMR-joint significantly outperforms JSLE on all datasets using VGG features. On average, MFMR-joint gains 7.4% performance compared with SynC-structure using VGG features on all datasets. It is notable that our MFMR-joint also clearly outperforms TMV-HLP which benefits from exploring the data structure of test instances. Thus our method is more effective to enhance the learned projection and to overcome the test-time projection domain shift problem as opposed to TMV-HLP which incorporates test instance in learning procedure under the transductive setting.

4.2.2 Results on Zero-shot Retrieval

In the retrieval task, a semantic embedding vector of an unseen class is used as a query to retrieve top matched test images. The performance is measured by mean average precision (mAP). As the retrieval task is not widely evaluated in prior studies, it's also an important contribution of this work provide a comprehensive comparison with the state-of-the-art methods of SSE-INT/SSE-ReLU, JSLE and SynC.

Table 3 lists comparative results in terms of mAP for all datasets using VGG features. We can see that our MFMR obtains mAP score of 56.2% compared with the result (51.1%) from the best counterpart of SynC-struct. It thus again validates the superiority of our MFMR on learning more effective projection. Furthermore, our MFMR-joint significantly and consistently gains dramatic performance improvement against SynC-struct by 31.9% on average. The superior performance of MFMR-joint in retrieval is due to the useful manifold structure explored in the test instances, which enhances the learned projection to better match the test instances to the semantic embeddings of unseen classes.

Table 3. Retrieval performance comparison (%) in terms of mAP. The best results on each dataset are highlighted in bold font.

Method	aPY	AwA	CUB	SUN	Ave.
SSE-INT [42]	15.4	46.25	4.7	58.9	31.3
SSE-ReLU [42]	14.1	42.6	3.7	44.6	26.2
JSLE [43]	32.7	66.5	23.9	76.5	49.9
SynC-ovo [6]	29.6	64.3	30.4	72.1	49.1
SynC-struct [6]	30.4	65.4	34.3	74.3	51.1
MFMR	45.6	70.8	30.6	77.4	56.2
MFMR-joint	**55.9**	**82.8**	**47.5**	**83.2**	**67.4**

Table 2. Comparison in terms of accuracy (%) for zero-shot recognition task on all datasets using deep features of VGG and GoogLeNet (numbers in parentheses). Here ‡ means the numbers of the method are cited from the original paper, while § means partial numbers are obtained from our implementation. "-" means no repeated result available yet. The best results on each dataset are highlighted in bold font.

Method	aPY	AwA	CUB	SUN	Average
DAP [17]‡	38.2 (-)	57.2 (60.5)	44.5 (39.1)	72.0 (44.5)	- (-)
ALE [1]‡	- (-)	61.9 (53.8)	40.3 (40.8)	- (53.8)	- (-)
ConSE [25]§	37.6 (-)	61.6 (63.3)	- (36.2)	- (51.9)	- (-)
ESZSL [31]§	24.2 (22.1)	75.2 (64.5)	44.5 (34.5)	82.1 (76.7)	56.5 (49.4)
TMV-HLP [13]‡	- (-)	80.5 (-)	47.9 (-)	- (-)	- (-)
SSE-INT [42]§	44.2 (39.7)	71.5 (73.2)	30.2 (29.3)	82.2 (77.4)	57.0 (54.9)
SSE-ReLU [42]§	46.2 (43.1)	76.3 (74.9)	30.4 (28.6)	82.5 (78.1)	58.9 (56.2)
JSLE [43]§	50.4 (48.2)	79.1 (77.8)	41.8 (38.6)	83.8 (84.0)	63.8 (62.1)
SynC-ovo [6]§	47.2 (41.3)	77.3 (69.7)	48.8 (53.4)	79.5 (78.0)	63.2 (60.6)
SynC-struct [6]§	48.9 (44.2)	78.6 (72.9)	50.3 (**54.5**)	81.5 (80.0)	64.8 (62.9)
MFMR	48.2 (46.4)	79.8 (76.6)	47.7 (46.2)	84.0 (81.5)	64.9 (62.7)
MFMR-joint	**56.8** (**54.3**)	**83.5** (**79.3**)	**53.6** (51.4)	**84.5** (**83.0**)	**69.6** (**67.0**)

4.2.3 Results on Generalized Zero-shot Recognition

The generalized zero-shot recognition measures the capability of the recognition system on trading off the prediction of seen and unseen classes. In our experiment, we reorganize the AwA dataset by composing test data as a combination of images from both seen and unseen classes. Specifically, we extract 20% of the data points from the seen classes and merge them with the data from the unseen classes to form the test set, resulting in a new training/test split with 19,452 images of seen classes for training, 4,843 images of seen classes combined with 6,180 images of unseen classes for test.

Let $\mathcal{T} = \mathcal{S} \cup \mathcal{U}$ denote the joint label space of seen and unseen classes, we evaluate the recognition performance in terms of accuracy on $\mathcal{U} \to \mathcal{U}, \mathcal{S} \to \mathcal{S}, \mathcal{U} \to \mathcal{T}, \mathcal{S} \to \mathcal{T}$, and the Area Under Seen-Unseen accuracy Curve (AUSUC). Detailed definitions of these metrics can be referred to [7], and all of them assess the recognition model on balancing the prediction of seen and unseen classes in the $\mathcal{T}$.

We compare our methods with the counterparts ConSE and SynC under the generalized ZSL setting, with the comparative results on AwA with VGG features provided in Table 4.2.3. In general, the performance of all the methods degrades under the generalized ZSL setting, i.e. the accuracy scores of $\mathcal{U} \to \mathcal{U}$ and $\mathcal{S} \to \mathcal{S}$ are larger than those of $\mathcal{U} \to \mathcal{T}$ and $\mathcal{S} \to \mathcal{T}$. Nevertheless, it is notable that the performance decay of our methods is less serious than ConSE and SynC, especially on the metrics of $\mathcal{U} \to \mathcal{U}$ and $\mathcal{U} \to \mathcal{T}$. In addition, our methods obtain larger AUSUC scores compared with the counterparts, and MFMR-joint achieves the best performance on all the five metrics. It indicates that our methods are more robust to trade off the prediction between the seen and unseen classes. Besides, modelling the test-time manifold structure is consistently beneficial under the generalized ZSL setting. Figure. 2 plots the Seen-Unseen accuracy Curves [7] of all the methods on AwA under the generalized ZSL setting. It can be observed that our methods generally have larger areas (corresponding to AUSUC score in Table 4) compared with the counterparts, which again indicates that our methods is more effective to balance $\mathcal{U} \to \mathcal{T}$ and $\mathcal{S} \to \mathcal{T}$.

Table 4. Comparative results of our methods and several alternatives for AwA under the generalized ZSL setting. The best results on each measure are highlighted in bold font.

Method	$\mathcal{U} \to \mathcal{U}$	$\mathcal{S} \to \mathcal{S}$	$\mathcal{U} \to \mathcal{T}$	$\mathcal{S} \to \mathcal{T}$	AUSUC
ConSE [25]	72.1	72.1	9.8	69.8	0.438
SynC-ovo [6]	76.4	**77.6**	1.1	**75.7**	0.509
SynC-struct [6]	79.6	76.8	1.8	76.1	0.533
MFMR	79.9	76.1	13.4	75.6	0.550
MFMR-joint	**81.2**	76.9	**18.4**	75.6	**0.571**

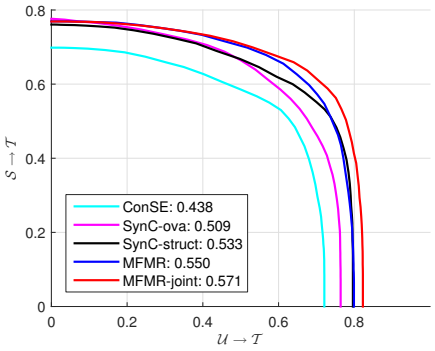
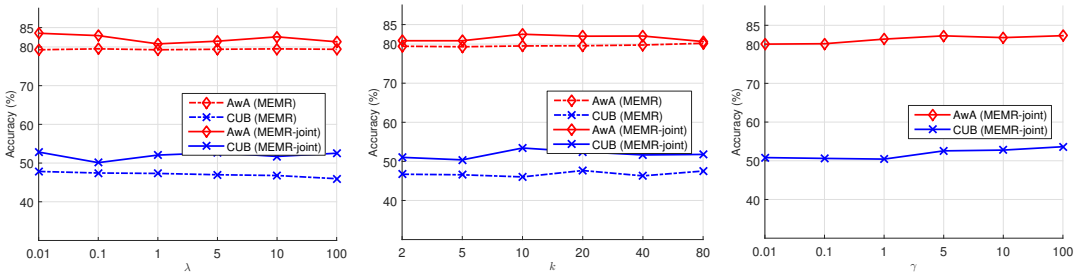


Figure 2. Comparison of The Seen-Unseen accuracy Curves between our methods and several alternatives for AwA under the generalized ZSL setting.

4.3. Detailed Analysis

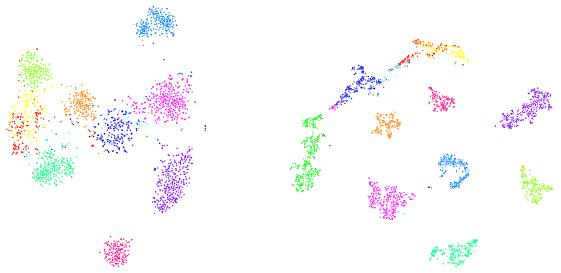
4.3.1 Effects of the Learned Projection

To better understand the effects of the learned projection by our methods, we employ t-SNE [35] to visualize the projected features of the test instances in the semantic embedding space. Figure. 3 illustrates the visualized distributions



(a) Accuracy with various λ (b) Accuracy with different k (c) Accuracy with different γ

Figure 4. Parameter sensitivity of MFMR and MFMR-joint on AwA and CUB datasets.



(a) MFMR (b) MFMR-joint

Figure 3. t-SNE visualization comparison of the projected features obtained by MFMR and MFMR-joint in semantic embedding space on aPY.

of the projected VGG features of the test instances on aPY dataset by our MFMR and MFMR-joint. As we can observe, in Figure. 3 (a), the projected features by MFMR prone to forming separate clusters for the twelve unseen classes with considerable small overlaps, showing the effectiveness of the learned projection for recognition. Moreover, the cluster of each unseen class becomes more compact and the gaps between each class are much clearer in Fig 3 (b) of MFMR-joint. The reason is that MFMR-joint well explores the manifold structure of the test instance, which benefits the learned projections to correctly align the projected features of the test instances with the semantic embeddings of their corresponding unseen classes.

4.3.2 Parameter Sensitivity Analysis

To study the effects of the parameters in our methods on the unseen class data, we take the zero-shot recognition results in terms of accuracy on AwA and CUB datasets with respect to different parameters values. Specifically, MFMR and MFMR-joint share two parameters λ and k at the training stage, and MFMR-joint owns an additional parameter γ that accounts for the joint prediction scheme. In our experiments, we vary one parameter at each time while fixing the others to their optimal values for both methods by using VGG features.

The three sub-figures in Figure. 4 illustrate the impact of each parameter on our methods. We can observe that for different method, the optimal value of each parameter is variant, sometimes depending on the specific dataset. For example, MFMR is robust to a large range of λ (*e.g.*, $\lambda \in [0.01, 10]$) on the two datasets, while MFMR-joint is sensitive to specific range of λ (*e.g.*, $\lambda \in [0.1, 100]$ and $\lambda \in [0.1, 1]$ on AwA and CUB, respectively). In addition, the value of parameter k has more effect on MFMR-joint than MFMR, that $k \in [10, 20]$ is optimal for MFMR-joint on two datasets. Since the test-time data distribution shifts from the estimation based on the projection learned by the training data, MFMR-joint requires to adapt appropriate value of k to correctly model the manifold structure of the test instances. At last, with considerable larger value of γ, *e.g.*, $\gamma \in [5, 100]$, MFMR-joint gains performance improvement consistently on the two datasets, as larger γ again forces MFMR-joint to associate the projections of the test instances to the semantic embeddings of corresponding unseen classes more accurately.

5. Conclusion

In this paper, we described a simple yet effective framework for ZSL that was able to outperform the current state-of-the-art approaches on a standard collection of ZSL datasets. The main idea was to leverage the sophisticated technique of matrix tri-factorization with manifold regularizers to alleviate the limitations in previous projection based ZSL approaches. Furthermore, an effective prediction scheme was developed to exploit the manifold structure of test data that accounts for the risk of test-time domain shift. Extensive evaluations validated the efficacy of our framework on the conventional ZSL problem, and showed its robustness on the generalized ZSL problem.

Acknowledgements. This work was supported in part by the National Natural Science Foundation of China under Project 61602089, Project 61502081, Project 61572108, Project 61632007, and the Fundamental Research Funds for the Central Universities under Project ZYGX2014Z007, Project ZYGX2015J055, Project ZYGX2016KYQD114.

References

[1] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid. Label-embedding for attribute-based classification. In *CVPR*, pages 819–826, 2013. 1, 2, 5, 6, 7

[2] Z. Akata, S. E. Reed, D. Walter, H. Lee, and B. Schiele. Evaluation of output embeddings for fine-grained image classification. In *CVPR*, pages 2927–2936, 2015. 5

[3] M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396, 2003. 4

[4] S. Boyd and L. Vandenberghe. *Convex Optimization*. 2004. 4

[5] D. Cai, X. He, X. Wang, H. Bao, and J. Han. Locality preserving nonnegative matrix factorization. In *IJCAI*, pages 1010–1015, 2009. 4

[6] S. Changpinyo, W. Chao, B. Gong, and F. Sha. Synthesized classifiers for zero-shot learning. In *CVPR*, pages 580–587, 2016. 1, 3, 5, 6, 7

[7] W.-L. Chao, S. Changpinyo, B. Gong, and F. Sha. An empirical study and analysis of generalized zero-shot learning for object recognition in the wild. In *ECCV*, pages 52–68, 2016. 1, 3, 7

[8] J. Deng, A. C. Berg, K. Li, and L. Fei-Fei. What does classifying more than 10,000 image categories tell us? In *ECCV*, pages 71–84, 2010. 1

[9] J. Deng, J. Krause, and L. Fei-Fei. Fine-grained crowdsourcing for fine-grained recognition. In *CVPR*, pages 580–587, 2013. 1

[10] C. Ding, T. Li, W. Peng, and H. Park. Orthogonal nonnegative matrix t-factorizations for clustering. In *KDD*, pages 126–135, 2006. 3

[11] A. Farhadi, I. Endres, D. Hoiem, and D. A. Forsyth. Describing objects by their attributes. In *CVPR*, pages 1778–1785, 2009. 1, 5

[12] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov. Devise: A deep visual-semantic embedding model. In *NIPS*, pages 2121–2129, 2013. 1

[13] Y. Fu, T. M. Hospedales, T. Xiang, and S. Gong. Transductive multi-view zero-shot learning. *TPAMI*, 37(11):2332–2345, 2015. 1, 3, 6, 7

[14] Z. Fu, T. A. Xiang, E. Kodirov, and S. Gong. Zero-shot object recognition by semantic manifold distance. In *CVPR*, pages 2635–2644, 2015. 2, 3

[15] P. Kankuekul, A. Kawewong, S. Tangruamsub, and O. Hasegawa. Online incremental attribute-based zero-shot learning. In *CVPR*, pages 3657–3664, 2012. 3

[16] E. Kodirov, T. Xiang, Z. Fu, and S. Gong. Unsupervised domain adaptation for zero-shot learning. In *ICCV*, pages 2452–2460, 2015. 1, 3, 5

[17] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*, pages 951–958, 2009. 1, 2, 3, 5, 6, 7

[18] C. H. Lampert, H. Nickisch, and S. Harmeling. Attribute-based classification for zero-shot visual object categorization. *TPAMI*, 36(3):453–465, 2014. 5

[19] H. Larochelle, D. Erhan, and Y. Bengio. Zero-data learning of new tasks. In *AAAI*, pages 646–651, 2008. 1

[20] M. Long, J. Wang, G. Ding, D. Shen, and Q. Yang. Transfer learning with graph co-regularization. In *AAAI*, 2012. 3, 4

[21] Y. Long, L. Liu, F. Shen, and L. Shao. From zero-shot learning to conventional supervised classification: Unseen visual data synthesis. In *CVPR*, 2017. 1

[22] T. Mensink, E. Gavves, and C. G. M. Snoek. COSTA: co-occurrence statistics for zero-shot classification. In *CVPR*, pages 2441–2448, 2014. 3

[23] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013. 1

[24] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, pages 3111–3119, 2013. 1

[25] M. Norouzi, T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G. Corrado, and J. Dean. Zero-shot learning by convex combination of semantic embeddings. *NIPS*, pages 410–418, 2013. 2, 6, 7

[26] M. Palatucci, D. Pomerleau, G. E. Hinton, and T. M. Mitchell. Zero-shot learning with semantic output codes. In *NIPS*, pages 1410–1418. 2009. 3

[27] D. Parikh and K. Grauman. Relative attributes. In *ICCV*, pages 503–510, 2011. 1

[28] G. Patterson, C. Xu, H. Su, and J. Hays. The SUN attribute database: Beyond categories for deeper scene understanding. *IJCV*, 108(1-2):59–81, 2014. 5

[29] Y. Pei, N. Chakraborty, and K. Sycara. Nonnegative matrix tri-factorization with graph regularization for community detection in social networks. In *IJCAI*, pages 2083–2089, 2015. 3, 4

[30] M. Rohrbach, M. Stark, G. Szarvas, I. Gurevych, and B. Schiele. What helps where - and why? semantic relatedness for knowledge transfer. In *CVPR*, pages 910–917, 2010. 2

[31] B. Romera-Paredes and P. H. S. Torr. An embarrassingly simple approach to zero-shot learning. In *ICML*, pages 2152–2161, 2015. 1, 2, 3, 5, 6, 7

[32] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. 6

[33] R. Socher, M. Ganjoo, C. D. Manning, and A. Y. Ng. Zero-shot learning through cross-modal transfer. In *NIPS*, pages 935–943, 2013. 1

[34] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, pages 1–9, 2015. 6

[35] L. van der Maaten and G. E. Hinton. Visualizing high-dimensional data using t-sne. *JMLR*, 9:2579–2605, 2008. 7

[36] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. 5

[37] D. Wang, Y. Li, Y. Lin, and Y. Zhuang. Relational knowledge transfer for zero-shot learning. In *AAAI*, pages 2145–2151, 2016. 1, 3

[38] H. Wang, F. Nie, H. Huang, and C. H. Q. Ding. Dyadic transfer learning for cross-domain image classification. In *ICCV*, pages 551–556, 2011. 3

[39] X. Yu and Y. Aloimonos. Attribute-based transfer learning for object categorization with zero/one training example. In *ECCV*, pages 127–140, 2010. 3

[40] H. Zhang, Z. Kyaw, S.-F. Chang, and T.-S. Chua. Visual translation embedding network for visual relation detection. In *CVPR*, 2017. 1

[41] H. Zhang, Z.-J. Zha, Y. Yang, S. Yan, Y. Gao, and T.-S. Chua. Attribute-augmented semantic hierarchy: towards bridging semantic gap and intention gap in image retrieval. In *ACM MM*, pages 33–42, 2013. 1

[42] Z. Zhang and V. Saligrama. Zero-shot learning via semantic similarity embedding. In *ICCV*, pages 4166–4174, 2015. 1, 2, 5, 6, 7

[43] Z. Zhang and V. Saligrama. Zero-shot learning via joint latent similarity embedding. In *CVPR*, pages 2124–2132, 2016. 3, 5, 6, 7

[44] Z. Zhang and V. Saligrama. Zero-shot recognition via structured prediction. In B. Leibe, J. Matas, N. Sebe, and M. Welling, editors, *ECCV*, pages 533–548, 2016. 3