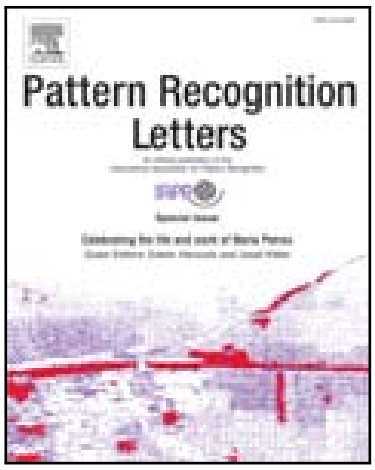

Zero-Shot Learning via Discriminative Representation Extraction

Teng Long, Xing Xu, Fumin Shen, Li Liu, Ning Xie, Yang Yang

PII: S0167-8655(17)30350-1
DOI: 10.1016/j.patrec.2017.09.030
Reference: PATREC 6942

To appear in: *Pattern Recognition Letters*

Received date: 30 May 2017
Revised date: 20 August 2017
Accepted date: 21 September 2017

Please cite this article as: Teng Long, Xing Xu, Fumin Shen, Li Liu, Ning Xie, Yang Yang, Zero-Shot Learning via Discriminative Representation Extraction, *Pattern Recognition Letters* (2017), doi: 10.1016/j.patrec.2017.09.030

Highlights

- We show that discriminative visual representation can help zero shot learning
- Supervised discriminative representation learning on seen classes can be transferred partly to unseen classes
- Comparing with Large Margin, aggregated representation may be more informative in zero shot learning.

Pattern Recognition Letters
journal homepage: www.elsevier.com

Zero-Shot Learning via Discriminative Representation Extraction

Teng Long[a,b], Xing Xu[a], Fumin Shen[a,**], Li Liu[c], Ning Xie[a], Yang Yang[a]

[a] *Center for Future Media and School of Computer Science and Engineering, University of Electronic Science and Technology of China, China*
[b] *Fujian Provincial Key Laboratory of Information Processing and Intelligent Control (Minjiang University), Fuzhou 350121, China*
[c] *School of Computing Science, University of East Anglia, UK*

ABSTRACT

Zero-shot learning (ZSL) aims to recognize classes whose samples did not appear during training. Existing research focuses on mapping deep visual feature to semantic embedding space explicitly or implicitly. However, ZSL improvements led by discriminative feature transformation is not well studied. In this paper, we propose a ZSL framework that maps semantic embeddings to a discriminative representation space, which are learned in two different ways: Kernelized Linear Discriminant Analysis (KLDA) and Central-loss based Network (CLN). KLDA and CLN can both force samples to be intra-class aggregation and inter-class separation. With the learned discriminative representations, we map class embeddings to representation space using Kernelized Ridge Regression (KRR). Our experiments show that both KLDA+KRR and CLN+KRR surpass state-of-art approaches in both recognition and retrieval task.

1. Introduction

Conventional recognition systems require thousands of labeled images for each class to achieve good recognition performance (Deng et al., 2010). Recently, the need of fine-grained recognition (Deng et al., 2013) grows rapidly. Therefore, the number of different objects become extremely large, and labeling becomes neither economical nor practical.

Learning with labeled data is not how humans learn to understand the world. When facing zero-shot tasks, humans can deduce and analyze from related knowledge. For instance, assuming one person has never seen a panda before, after reading "bear like animal with large, distinctive black patches around its eyes, over the ears, and across its round body". He can recognize "panda" as he saw a panda for the first time, even after being briefly perplexed because facing an "unseen" class. Zero-shot learning imitates the process how humans recognize objects that he had never seen.

ZSL divides categories into two disjoint sets: seen classes and unseen classes. At train stage, only labeled instances from seen classes are available. Without labeled examples, ZSL learns classifiers for unseen classes by transferring knowledge from the seen classes. Fig. 1 shows ZSL problem. Without seen samples, it seems to be impossible to infer images that do not reside on seen manifold. However, we can exploit multi-modal information to assign labels to unseen classes because images are not the only information source. We can take advantage of side information to learn unseen classifiers. One of most commonly used side information is called semantic em-

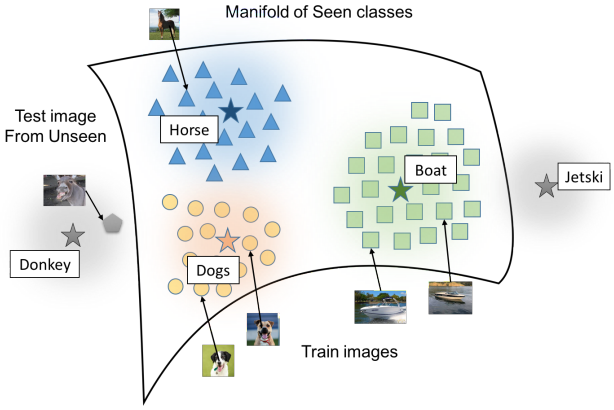

Fig. 1. Zero shot learning problem. Each class has a template (denoted by a star), To classify an unseen image, we assign it a label that corresponds to the nearest template in feature space. However, this is hard due to we only have access to seen templates at train stage. Side semantic information is needed to determine where an unseen template locates.

**Corresponding author. Email: fumin.shen@gmail.com

bedding, which contains semantic information like similarity relationships between classes. Semantic embeddings are usually easily obtained, therefore we always have full access to all classes embeddings. ZSL then can be achieved by resorting to a common semantic embedding space in which seen and unseen classes are related(Al-Halah et al., 2016).

Most existing ZSL approaches utilize attributes (Farhadi et al., 2009; Lampert et al., 2009; Parikh and Grauman, 2011) and word2vec representations (Mikolov et al., 2013; Mikolov, Tomas et al., 2013; De Boom et al., 2016) as semantic spaces. In semantic embedding spaces, each class name can be represented by a vector in a supervised way based on a pre-defined attribute ontology, or an unsupervised way based on vast unannotated text corpus.

Given a semantic embedding space, we can measure the similarity between classes by calculating their distances. ZSL assumes that the spatial relationship between classes in semantic embedding space is similar to the spatial relationship in visual feature space. Several existing ZSL approaches (Akata et al., 2013; Frome et al., 2013; Romera-Paredes and Torr, 2015; Zhang and Saligrama, 2015) learn a projection function from visual features space to semantic embedding space. At test stage, the inference of an unseen image is performed by first mapping image visual feature to semantic embedding space, then measuring the similarities of projections with unseen classes in this embedding space.

However, major projection based approaches still have several shortcomings. Firstly, Visual features can be suboptimal for zero shot tasks as they were learned designed from multi-shot recognition task. Secondly, the $V \rightarrow S$ mapping (mapping from visual feature space to semantic embedding space) maps n samples to one embedding vector. As image number is far beyond class number, this mapping can be inefficient and noisy. Thirdly, (Zhang et al., 2017) shows that inferring a test image in visual space instead of semantic embedding space may lead to suboptimal results due to hubness (Dinu et al., 2014) problem.

In this paper, we focus on learning discriminative zero shot visual representation to tackle problems mentioned above. Specifically, we add a preprocessing stage before training stage. At preprocess stage, we reduce visual feature dimension into a representation space in a semi-supervised learning manner. Samples within this representation space have high inter-class variation and low intra-class variation. After dimension reduction, each seen class averages over all images within that class to obtain a "template", i.e., we simplify "n images corresponding to one class" relationship to "one template correspond to one class". At train stage, a nonlinear regression was learned on train data to infer class "template" in representation space using semantic embeddings as inputs. At test stage, as one test sample arrives, we first reduce its dimension to representation space, we calculate its similarities to unseen class templates to decide which class it belongs to.

To better conduct ZSL, (Xian et al.) argue that in standard ZSL experiment setting, some unseen classes appear before test stage. Standard ZSL experiment setting extract image features from Deep Neural Networks, which is pre-trained on ImageNet dataset. In this setting, several test classes are among ImageNet classes. This may violate zero shot learning definition that unseen classes should not appear before test stage. Therefore, standard experiment setting may lead to flawed conclusions. One way to calibrate experiment setting is to readjust seen/unseen split to follow zero shot definition. With adjusted seen/unseen split in (Xian et al.), flawed factor was eliminated, we may better analysis the underlying ZSL factors. We follow both experiment settings to conduct our experiments. Our analysis shows that Large Margin and Aggregated are two key factors in ZSL.

Our contributions are mainly three-folds: 1) We verify that supervised discriminant training on seen classes can benefit unseen classes. 2) We disclose that Aggregated visual representation works even better than Large Margin representation in zero shot recognition problem 3) We conduct extensive experiments on four major benchmark datasets, which validate the superiority of our approach over state-of-the-art approaches on zero-shot recognition and retrieval task.

The structure of the paper is as follows. Section 2 introduce related work, Sections 3 define DZSL model and detail how we to train this model. Section 4 discuss the performance of our model, including traditional experiment train/test class split and newly proposed train/test class split. Section 5 analyze experiment conducted on large scale datasets and Section 6 concludes this work.

2. Related Work

The pioneer work in ZSL can be traced to (Larochelle et al., 2008), where it verified that when test images belong to some classes that are not available at training stage, a machine learning system can still figure out what a test image is. Due to the importance of zero-shot learning, the number of proposed approaches has increased steadily recently.The number of new zero-shot learning approaches proposed every year was increasing.

Existing approaches differ in how they transfer knowledge between seen and unseen classes. Most existing approaches are grouped into similarity based and projection based approaches.

Methods based on compatibility functions include ALE (Xian et al., 2016) SJE (Akata et al., 2015) LatEM (Xian et al., 2016). Each of these approaches learns a bilinear compatible function of visual space to semantic space. This compatible function represents how well a sample is compatible with a class label. The learning procedure force correct labeling to have a higher compatible score. The advantage of this approach is to take into account the characteristics of large margin. However, the shortcomings are: do not make full use of space in the neighborhood information.

Similarity based approaches represents unseen classes as a mixture of seen classes proportion includes SSE (Zhang and Saligrama, 2015), SynC (Changpinyo et al., 2016a) and GZSL (Chao et al., 2016). SSE represents each class as a linear combination of Seen class. SynC and GZSL extract the effective information of the Seen class of the embedding space - a set of basis, representing the categories as the coordinates of the set of basis in the embedding space. And assume that in visual space,

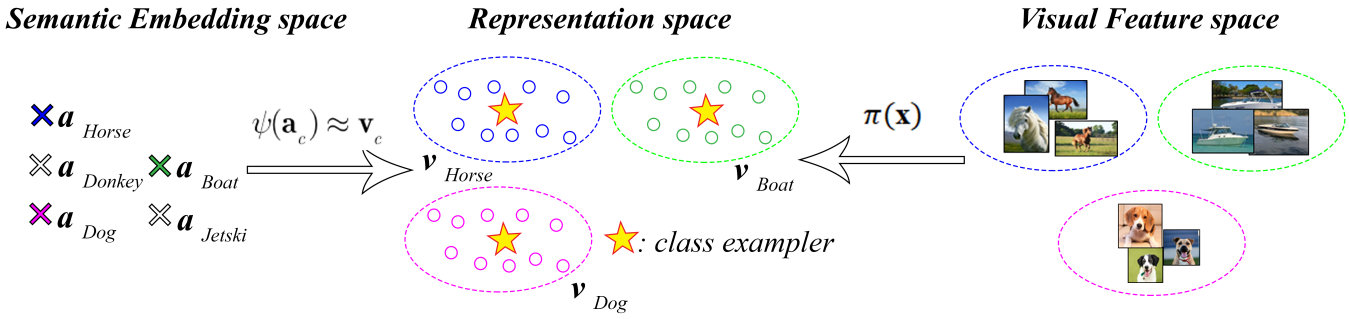

Fig. 2. Visual features and semantic embeddings are closely related. Given visual features, a more aggregated representation can be learned by $\pi(\mathbf{x}) = \mathbf{v}$. With the help of Kernel Ridge Regression (KRR), semantic embedding can be mapped to representation space by $\psi(\mathbf{a_c}) \approx \mathbf{v}_c$. We denote class center in representation space as a class template. Samples with a class are distributed around its template. This property can be advantageous for recognition and and retreival for nearest search.

this group of basis can effectively represent seen and unseen categories. These approaches take advantage of the rich seen class information, reducing the computational complexity, but did not reveal the key factors to improve the accuracy of ZSL.

Recently, matrix decomposition was found to have nice properties as not only did it represents a linear transformation but also can it be viewed to maintain common information in a decomposed sub matrix. Methods based on matrix decomposition assume sample matrix $\mathbf{X}$ and label matrix $\mathbf{Y}$ has linear relationship. Therefore decompose based approaches optimized $\|\mathbf{Y} - \mathbf{W^T X}\|_F$, in which $\mathbf{W}$ can be decomposed in various ways. ESZSL (Romera-Paredes and Torr, 2015) decompose $\mathbf{W} = \mathbf{V^T S}$, where $\mathbf{V}$ is intrinsic to all images and $\mathbf{S}$ is class-dependent.

Moving one step further, (Qiao et al., 2016) decompose the matrix $\mathbf{V} = \mathbf{W_x^T W_z}$. By regularization to suppress the noise in word embedding. However, this approach requires embedding to be interpretable, which is difficult to achieve in practical applications. MFMR (Xu et al., 2017) decomposes the matrix W via matrix three decomposition and manifold regularization, which addressed the domain shift problem. Matrix Decomposition based approach has clear physical meaning. However, these approaches depict transformations between visual space and semantic linearly, this may ignore nonlinear characters resides on datasets.

Methods based on large margin mechanism includes SSE (Zhang and Saligrama, 2015), JLSE (Zhang and Saligrama, 2016) and SJE (Akata et al., 2015). These approaches force (x) to have a large margin. However, the large margin mechanism is only one factor among many decisive factors. We show that centralized feature is more powerful as it forced centralized visual representation.

Usually, Standard zero shot experiment setting forbid seen classes to appear at test stage. In reality, one cannot assume that one image purely comes from unseen classes. Seen classes and unseen classes will both appear at test time. To solve this problem, DeVise (Frome et al., 2013) ConSE (Norouzi, Mohammad et al., 2014) SynC (Changpinyo et al., 2016a), GZSL (Chao et al., 2016) applied generalized zero shot learning setting using ImageNet dataset (Deng et al., 2009). Recently, (Xian et al.) shows that standard setting may be noisy when analysis reasons of the performance of zero shot learning.

3. Method

3.1. Problem statement

let $\mathcal{S} = \{1, 2, 3, \cdots, S\}$ denotes a set of classes that contains S seen classes, and $\mathcal{U} = \{S + 1, S + 2, \cdots, S + U\}$ denotes another set of classes that contains c_u unseen classes. This two sets are disjoint, e.g $\mathcal{S} \cap \mathcal{U} = \emptyset$. each class in the two sets can be represented by a m-dimensional semantic embedding vector (e.g. attributes) $\mathbf{a_i}$, where $\mathbf{a_i}$ and is embedding vectors for i-th class. Training samples are denoted by$\{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^{n_s}$, where n_s is the number of train samples, $\mathbf{x_i^s}$ denotes i-th train sample which is a d-dimensional feature vector.

3.2. General Framework

The main idea of our approach is shown in Fig. 2. At training stage. Firstly, we learn a discriminative visual representation using train data in seen classes. Then, we learn a projection from semantic space to representation space. Comparing with projectioning visual representation to semantic space. This approach has two major advantages: 1) Inferring class in visual representation space directly; 2) Efficient computation.

Let class C_k has n_k samples, mapping from semantic embedding space to visual representation space is a one-to-many mapping. As the number of visual samples is much larger than the number of semantic embeddings, learning a projection from visual space to semantic embeddings space may require much more computational resources. More importantly, for larger data sets like ImageNet, mapping in the former way become incomputable. We simplify this mapping to a one-to-one mapping by resorting to "template". Our approach represents a class in visual representation space as a "class template" vector $\mathbf{v}_c$. a "template" can be viewed as a standard sample of a class. Therefore, it's reasonable to use the statistical mean of all samples as the template. We take the statistic average as following Eq. (1).

$$\mathbf{v}_c = \frac{1}{n_k} \sum_{\mathbf{x}_i \in C_k} \mathbf{x}_i. \tag{1}$$

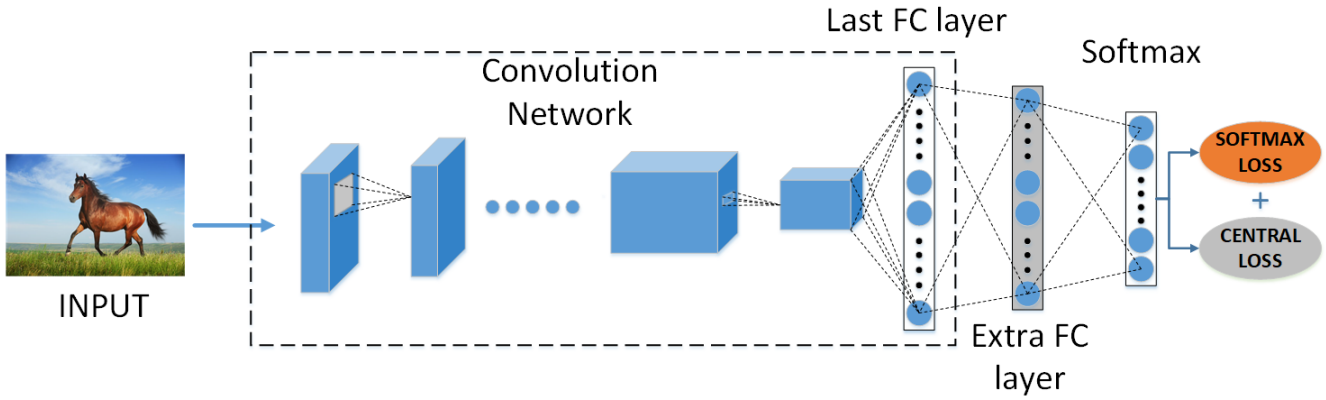

Fig. 3. Modified deep convolution network. We add an FC layer which contains d' neuron units between last fully-connected layer and softmax layer. The propose of this layer is to reduce feature dimension in a aggregated way. Addition to softmax loss, a central loss is added to supervise aggregated feature learning.

When one test sample arrives, the test sample finds its nearest template in the visual representation space and assign the template's label to this sample. Intuitively, this approach is similar to the effect of mapping feature space to semantic space, but the amount of computation will be much smaller.

3.3. Visual representation transformation

It is expected that visual features contain less noise and more information. This requires us to learn a transformation of visual features. (Changpinyo et al., 2016b) used PCA to obtain visual representations, PCA reduced the noise level in visual features. However, PCA transform features in an unsupervised way. Unsupervised feature transformation does not make full use of label information. Therefore, labels information leaves room for improvement.

Supervised feature transformationHardoon et al. (2006); Baudat and Anouar (2000); Wen et al. (2016) utilized labels information to obtain supervised visual representations, which achieve intra-class aggregation, and inter-class separation characteristics. We refer this two character as **Large Margin** and **Aggregated**. In this paper, we consider two different feature transformation method, KLDAHardoon et al. (2006) and CLN(Wen et al., 2016). KLDA and CLN exploit labels information in different manners, we found that the common factor in both transformations improved ZSL performance while the comparison between this two methods prompt us **Aggregated** feature might be more powerful in ZSL.

3.3.1. KLDA repsentation transformation

As a first option, we use KLDA to transform visual features. Different from regular KLDA, in ZSL, KLDA was conducted on seen classes but the learned transformation was used on both seen and unseen classes. With transformed features, the template can be obtained by Eq. (2). Comparing with PCA, KLDA has the properties of **Large Margin** and **Aggregated**. Our experiments show that such properties are also effectively transferred to the unseen classes. In a traditional classification task, KLDA may lead to better discriminative transformation. However, in ZSL problem, this discriminative power remains unclear. Our experiment in section 4.1 shows that even only trained on seen classes, KLDA based representation achieve better performance compared with PCA.

$$\mathbf{y}(\mathbf{x}_t) = (\mathbf{A}^*)^T \begin{pmatrix} \mathbf{K}(\mathbf{x_1}, \mathbf{x_t}) \\ \mathbf{K}(\mathbf{x_2}, \mathbf{x_t}) \\ \vdots \\ \mathbf{K}(\mathbf{x_{n_s}}, \mathbf{x}) \end{pmatrix}. \tag{2}$$

Here $\mathbf{K}(\mathbf{x_i}, \mathbf{x_j})$ is the inner product of $\mathbf{x}_i$ and $\mathbf{x}_j$ in nonlinear transformation space. $\mathbf{A}^*$ can be computed by Eq. (3)

$$\begin{aligned} \mathbf{A}^* &= eigvecs((\sum_{j=1}^{c} \mathbf{K}_j(\mathbf{I} - \mathbf{1}_{l_j})\mathbf{K}_j^T)^{-1}\mathbf{M}), \\ \mathbf{M} &= (\sum_{j=1}^{c} l_j(\mathbf{M}_j - \mathbf{M}_*)(\mathbf{M}_j - \mathbf{M}_*)^T). \end{aligned} \tag{3}$$

Note that $\mathbf{M}$ can be computed follow (Mika et al., 1999). $eigvecs(\cdot)$ is a operator that extract first k eigenvectors of its input matrix. For dimension reduction propose, k is set to be equal to d'. After dimension reduction, our template can be computed by Eq. (4).

$$\mathbf{v}_c = \frac{1}{n_k} \sum_{\mathbf{x}_i \in C_k} \mathbf{y}(\mathbf{x})_i. \tag{4}$$

3.3.2. Centralized repsentation transformation

As an alternative of KLDA, we proposed CLN based feature transformation. CLN utilizes neural networks to learn aggregated representation by combining softmax loss with a central loss function by punishing data samples whose representation are far away from their corresponding "class center" in representation space. A "class center" is the statistical average of all samples' feature within one class. Combined loss punishes not only the classification error, but also the samples deviation

Table 1. Statistics of different datasets, where "*/*" in columns represents image number in ST-1 setting/ image number in ST-2 setting. Parenthesis in Train/Test split column means ST-1(ST-2) classes split.

Dataset information					Train stage (ST-1/ST-2)	Train stage (ST-1/ST-2)		Test stage (ST-1/ST-2)	
dataset	Detail	Att	Classes	Train/Test Classes Split	Total	Seen	Unseen	Seen	Unseen
AWA	coarse	85	50	40/10	30K	24295/19832	0/0	0/2580	6180/5685
aPY	coarse	64	32	20/12	11K	12695/5932	0/0	0/1764	2644/7924
CUB	fine	312	200	150/50	30K	8855/7057	0/0	0/4958	2933/2967
SUN	fine	102	717	707/10(645/72)	15K	14140/10320	0/0	0/1483	200/1440

from class centers. Optimizing combined loss function, we can obtain aggregated representation.

Specifically, we add an additional fully-connected(FC) layer after last fully-connected layer on deep convolution network. To simplify models, this additional layer does not contain activation function. This additional FC layer has d' units, where d' is the dimension of the space that we transform to. Note that d' can be tuned as a hyperparameter. The modified network was shown in Fig. 3. In our modified deep convolution network, representations are supervised by softmax loss combined with central loss, The combined loss function was given in Eq. (5) as

$$\begin{aligned}\mathcal{L} &= \mathcal{L}_S + \lambda\mathcal{L}_C, \\ &= -\sum_{i=1}^{m} log\frac{e^{W_{y_i}^T\mathbf{z}_i+b_{y_i}}}{\sum_{j=1}^{n} e^{W_j^T\mathbf{z}_i+b_j}} + \frac{\lambda}{2}\sum_{i=1}^{m}\|\mathbf{z}_i-\mathbf{c}_{y_i}\|_2^2.\end{aligned} \tag{5}$$

In Eq. (5), $\mathbf{z}_i = f(\mathbf{x}_i)$ where $f(\mathbf{x}_i)$ means to use the activation of our additional fully-connected layer as representation of $\mathbf{x}_i$. We optimize last FC layer's parameters W while keeping other layers frozen. λ denotes a trade-off between softmax loss and central loss. $\mathbf{c}_{y_i} \in \mathbb{R}^{d\prime}$ denotes the center of y_i-th class in representation space. Supervised by the combined loss function, $\mathbf{c}_{y_i}$ will be updated as representation changes each train iteration. To avoid mini-batch perturbation in centers updating process, an hyper-parameter α was introduced to damp the updating process. The training process used Algorithm 1.

Algorithm 1: Visual representation learning algorithm

Input : training data $\mathbf{X}, \mathbf{y}$
Output: FC layer parameters W and class centers $\mathbf{c}_{y_i}|y_i = 1, 2, \cdots, n_s$
Result: Additional CLN that transform features to aggregated representations

Initiallize network with pre-trained GoogLeNet parameters. Initialize $\mathbf{c}_{y_i}$ and W. Set hyperparameters λ and α
$t \leftarrow 0$
while *not converged* **do**
  $t \leftarrow t + 1$
  Compute joint loss by $\mathcal{L} = \mathcal{L}_S^t + \lambda\mathcal{L}_C^t$
  Update additional FC layer by $W^{t+1} = W^t - \mu^t \cdot \frac{\partial\mathcal{L}}{\partial W^t}$
  Update centers by $\mathbf{c}_{y_i}^{t+1} = \mathbf{c}_{y_i}^t - \alpha \cdot \Delta\mathbf{c}_{y_i}^t$
end

3.4. Predict template from semantic embeddings

After obtained discriminative representations by KLDA or CLN. We conduct ZSL with a regression model. For each class c, we learn a mapping $\psi(\cdot)$ that $\psi(\mathbf{a}_c) \approx \mathbf{v}_c$. This mapping can be modeled as a small data regression problem as each sample in this problem correspond to one class and class number is small. Benefited from small data character, we make full use of Kernel tricks to achieve non-linearity without much computation overhead. In this paper, we use Kernel Ridge Regression (Saunders et al., 1998) model.Given training templates and semantic embeddings, we learn d' kernel ridge regressors with RBF kernel, each of them predicts one dimension of $\mathbf{v}_c$ from their corresponding semantic representations. We learn a regressor by solving Eq. (6).

$$\begin{aligned}&\min_{\mathbf{w}_i}\frac{1}{N}\sum_{i=1}^{N}(y_n - \mathbf{w} - \phi(\mathbf{x}_n))^2 + \frac{\lambda}{N}\mathbf{w_i^T}\mathbf{w_i}, \\ &= \min_{\beta_i}\frac{1}{N}(\beta_\mathbf{i}^\mathbf{T}\mathbf{K}^\mathbf{T}\mathbf{K}\beta_\mathbf{i} - 2\beta_\mathbf{i}^\mathbf{T}\mathbf{K}^\mathbf{T}\mathbf{y_i} + \mathbf{y}^\mathbf{T}\mathbf{y_i}) + \frac{\lambda}{N}\beta_\mathbf{i}^\mathbf{T}\mathbf{K}\beta_\mathbf{i}.\end{aligned} \tag{6}$$

This problem can be solved in closed-form Eq. (7)

$$\beta_\mathbf{i}^* = (\lambda\mathbf{I} + \mathbf{K})^{-\mathbf{1}}\mathbf{y_i}. \tag{7}$$

Note that $i = 1, 2, \cdots, d'$, $\mathbf{K}$ is the kernel matrix computed using RBF-kenerl. λ is a hyper-parameter to be tuned. The resulting $\psi(\cdot) = (\mathbf{w_1}\phi(\cdot), \mathbf{w_2}\phi(\cdot), \cdots, \mathbf{w_{d'}}\phi(\cdot))^T$, where $\mathbf{w}_i = \sum_{i=1}^{N}\beta_n\phi(\mathbf{x_n})$ is computed using i-th regressor.

4. Experiment

Datasets Our experiment uses 4 popular datasets:

Animals with Attributes (AwA) (Lampert et al., 2009) aPascal-aYahoo (aPY) (Farhadi et al., 2009) Caltech UCSD Birds (CUB) (Wah et al., 2011) SUN Attribute (SUN) (Patterson et al., 2014).

AwA is a dataset consist of 50 categories of animal images, each category is represented by 85 numeric attribute values. aPY has 32 categories including animals and common objects, each category has 64 attributes for each class. CUB consists of 200 different species of birds, thus we can explore fine-grained knowledge transfer. SUN has a vast number of 717 categories that each contains 20 images. Train/test category split in 4 datasets varies in different evaluation protocols.

We adopt two different evaluation protocol. Standard setting (ST-1) is popular in previous research (Akata et al., 2013;

Table 2. Zero-shot recognition task comparison on all datasets using GoogLeNet features. In this table, ‡ means partial results are obtained from out implementation, while §means results were cited from original paper. '*' means results were obtained using VGG features. '–' means no avaible results yet. Within each column, the best is in red and the 2nd best is in blue. We measure top-1 accuracy in %.

approach	AWA	APY	CUB	SUN	Average
DAP‡ (Lampert et al., 2014)	60.5	–	39.1	44.5	–
ALE‡ (Akata et al., 2013)	53.8	–	40.8	53.8	–
ConSE§(Norouzi, Mohammad et al., 2014)	63.3	–	36.2	51.9	–
ESZSL§(Romera-Paredes and Torr, 2015)	64.5	17.1	34.5	18.7	33.7
SSE-INT‡ (Zhang and Saligrama, 2015)	71.5*	44.2*	30.2*	82.2*	57.0
SSE-ReLU‡ (Zhang and Saligrama, 2015)	76.3*	46.2	30.4*	82.5	58.9
JSLE‡ (Zhang and Saligrama, 2016)	73.0	**48.3**	35.4	78.0	58.7
SynC-ova§(Changpinyo et al., 2016a)	69.7	34.2	53.4	78.0	58.8
SynC-struct‡ (Changpinyo et al., 2016a)	72.9	38.7	54.5	80.0	61.5
MFMR§(Xu et al., 2017)	76.6	41.8	46.2	81.5	61.5
MFMR-Joint§(Xu et al., 2017)	79.3	47.8	51.4	**83.0**	65.3
PCA+SVR§(Changpinyo et al., 2016b)	78.6	38.8	54.43	81.0	63.2
KLDA+KRR	**79.3**	**46.4**	**58.4**	**83.0**	**66.5**
CLN+KRR	**81.0**	44.9	**58.6**	**84.0**	**67.1**

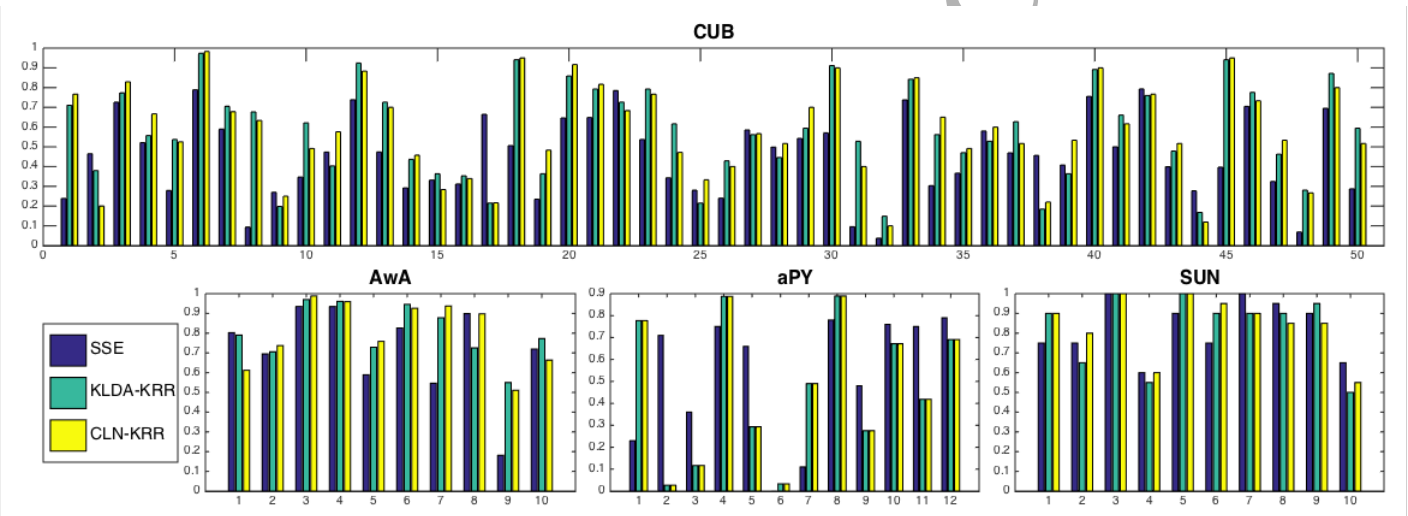

Fig. 4. Class-level recognition accuracy comparison, where y-axis denote accuracy and x-axis denotes the indexes of unseen classes in the corresponding datasets. SSE act as baseline method as it achieved state of the art.

Romera-Paredes and Torr, 2015; Zhang and Saligrama, 2016; Chao et al., 2016; Changpinyo et al., 2016a). Recently, a new setting (ST-2)(Xian et al.) was proposed to overcome the weakness in ST-1. ST-2 was proposed because ST-1 may violate zero shot problem setting. ST-2 show that ImageNet 1K classes contain some unseen classes within AwA, aPY, CUB and SUN. Extracting deep features uses deep convolution network pre-trained on ImageNet and therefore violate zero shot problem setting. After eliminating pre-trained class out of unseen classes, ST-2 evaluate zero shot performance more fairly. Therefore, the major difference between ST-1 and ST-2 is the train/test split. We conduct the experiment under both ST-1 and ST-2 protocol. Statistics of all datasets were shown in Table 1.

Semantic embeddings For AwA and aPY datasets, we directly utilize the provided class-level continuous attribute vector. For CUB and SUN datasets which have image-level attribute vector, thus, we generate class-level attribute vector by averaging attribute vectors over all images within one class.

Visual features (Akata et al., 2015) shows that deep features lead to better class separation than handcrafted features (Zhou et al., 2016, 2017). We use deep features extracted from deep CNNs. In this paper, we use 1024-dim GoogLeNet (Szegedy et al., 2015) and 2048-dim ResNet (He et al., 2016) features for AWA, CUB and SUN dataset available from (Changpinyo et al., 2016a) and extracted aPY features using Caffe (Jia et al., 2014).

Implementation details In our experiments, we address two variants of ZSL approach: KLDA-KRR and CLN-KRR. Hyperparameters in the two approaches are worth investigation: For KLDA-KRR model, $d', \gamma_1$are tuned for KLDA model, γ_2 is tuned for RBF-kernel in KRR, λ is tuned for regression regularization. Similarly, for CLN-KRR model,α, d', λ and γ are going to be tuned. In CLN-KRR model, we update the network with Gradient-based approach. Gradient-based approach update model with mini-batch data. Within one mini-batch, Our model update template position of each class by averaging over samples within each class. For datasets who has a large class

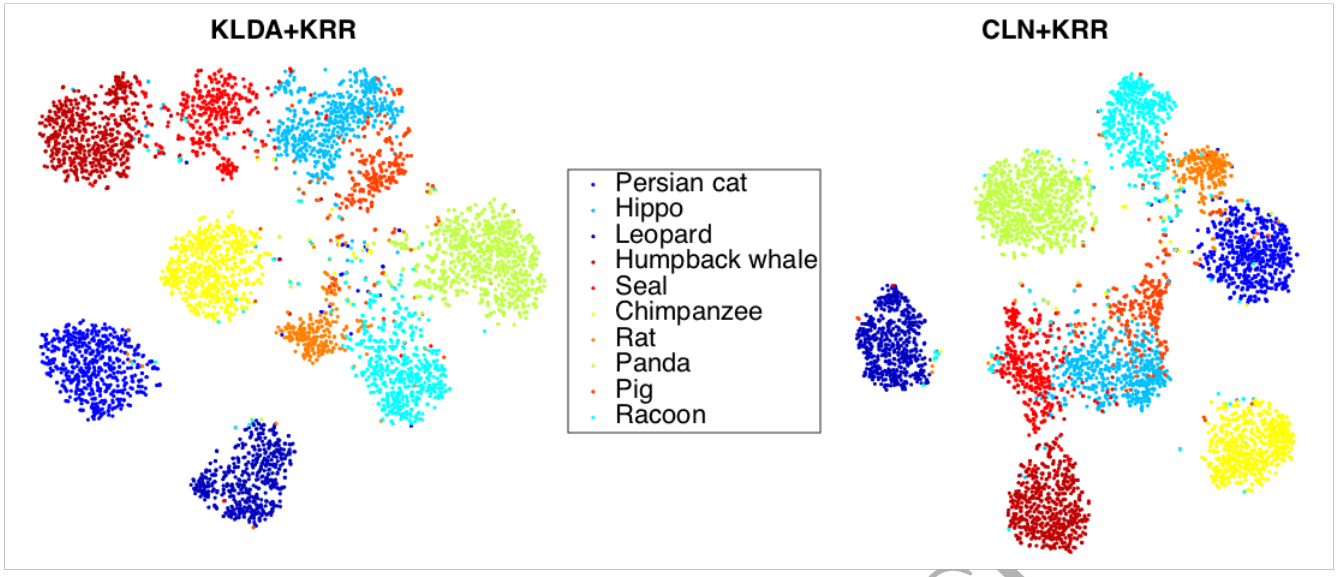

Fig. 5. t-SNE visualization for our discriminative representations for unseen classes in AwA dataset.

number, samples in mini-batch may not enough to update template correctly. Therefore, batch size may affect the center updating performance.

We report the averaged performance results over 10 runs of experiment with the optimal parameters chosen on each dataset. All the experiments are conducted on a PC equipped with 8-core 3.3GHz CPUs and 64GB RAM. Furthermore, two 1080-Ti GPUs was used as we accelerating CLN training with tensorflow (Abadi et al., 2016).

We conduct experiments on three tasks: zero shot recognition under ST-1 setting, zero shot retrieval under ST-1 setting and zero shot recognition under ST-2 setting. Zero shot retrieval under ST-2 setting was not studied due to the lack of related research.

4.1. Performance comparison

Our evaluation under ST-1 setting compared our approach with 10 existing ZSL approaches. We not only refer to the published results but also re-implemented SSE, JSLE, SynC, MFMR with provided implementation codes. Details were shown in Table 2. Our KLDA+KRR and CLN+KRR perform best among all datasets except for aP&Y dataset. JSLE outperform our methods as JSLE uses transductive experiment setting which exploits the distribution information from unseen classes. Note that our results of JSLE are lower than reported in the original paper as we use GoogLeNet(Szegedy et al., 2015) features instead of VGG-19(Simonyan, Karen and Zisserman, Andrew, 2014) features to reproduce results. CLN perform best on average, centralized representation forbid samples to distribute all over the representation space. Instead, CLN representations cluster tightly around its "template", therefore yields good performance.

4.2. Detail Analysis

To better understand the performance of our model for recognition, we also conduct class level accuracy comparison in Fig. 4. Here (and in the following experiments) we only consider SSE as the baseline approach because it achieves the state-of-the-art over the four datasets on average. In general, KLDA and CLN help improve the performance on individual classes as they maximized inter-class separation.

In few cases, however, we observe samples that are misclassified, such as class 11 , "bag" class in aPY dataset and class 1, "Persian cat" in AwA dataset. Most misclassified "bag" images were mis-classified to "monkey" and "statue". Have a close view of the images, one can see that misclassified "bag" images are mostly persons with bags and our model recognize them as something humanlike (in our experiment,"statue" and "monkey") instead of "bag".

AwA dataset did not provide origin images due to intellectual property reason. In order to gain some insight of this dataset, we analyze our model using t-SNE(Maaten and Hinton, 2008) visualization. We compared KLDA+KRR with CLN+KRR in Figure 5. Corresponding to recognition results, CLN based model perform better due to it learned more aggregated features. CLN based representations entangled less between classes than KLDA based representations.

In order to gain some insight of zero shot recognition, A sample wise analysis on aPY dataset also conducted. We analysis True Positive (TP) and False Positive (FP) prediction for each unseen class, the results were shown in Fig. 6.

Animal categories are quite confusing, "goat", "donkey" and "zebra" are usually confused with each other. Monkeys resemble human a lot, therefore many "monkey" images are recognized as "statue". On the other hand, almost no images are misclassified as "buildings" and "jetski" due to this two classes are quite discriminative comparing with other categories. Note that similar categories can always be confusing while distinct categories can be recognized easily, ensembling fine-grained classifier with coarse-grained classifier might improve ZSL performance.

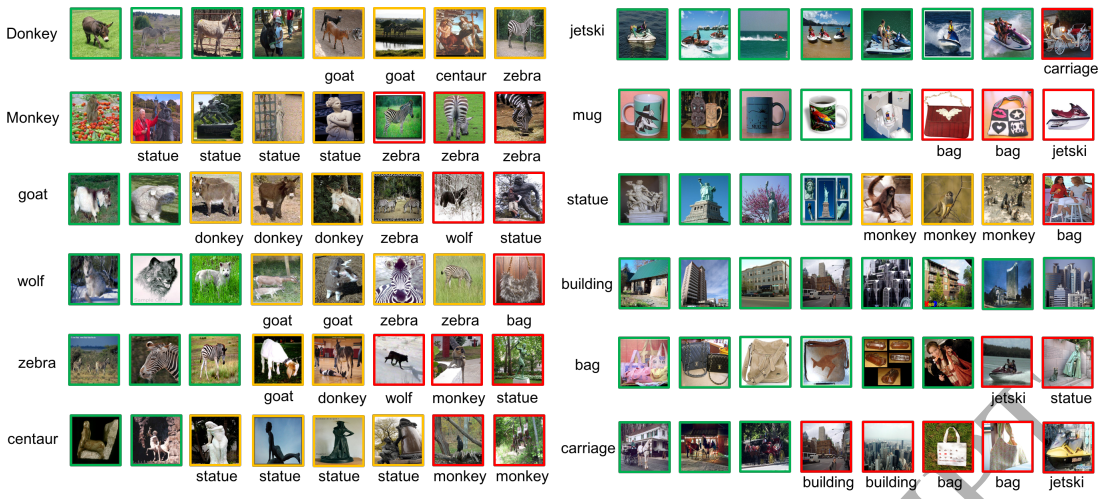

Fig. 6. Recognition result of images. Each row corresponds to one prediction category. Images with wrong predicted label was given correct label under its image. Green boundbox means correct. Yellow boundbox means similar but wrong. Red boundbox means totally wrong.

Table 3. Retrieval performance comparison (%) in terms of mAP, Best results were highlighted in bold fonts. Within each column, the best is in red and the 2nd best is in blue.

Method	AWA	APY	CUB	SUN	Ave.
SSE-INT	46.3	15.4	4.7	58.9	31.3
SSE-ReLU	42.6	14.1	3.7	44.6	26.2
JSLE	66.5	32.7	23.9	**76.5**	49.9
SynC-ova	64.3	29.6	30.4	72.1	49.1
SynC-struct	65.4	30.5	34.3	74.3	51.1
KLDA-KRR	**68.30**	**35.54**	**35.59**	74.21	**53.36**
CLN-KRR	**71.36**	**45.29**	**47.29**	**82.33**	**61.56**

Table 4. Zero shot recognition result on ST-2 experiment setting. Within each column, the best is in red and the 2nd best is in blue. We measure top-1 accuracy in %. references are not offered here due to limited space, please refer to Related Work.

Method	AWA	APY	CUB	SUN
DAP	44.1	33.8	40.0	39.9
CONSE	45.6	26.9	34.3	38.8
CMT	39.5	28.0	34.6	39.9
SSE	60.1	34.0	43.9	51.5
LATEM	55.1	35.2	49.3	55.3
ALE	59.9	39.7	54.9	**58.1**
DEVISE	54.2	**39.8**	52.0	56.5
SJE	**65.6**	32.9	53.9	53.7
ESZSL	58.2	38.3	53.9	54.5
SYNC	54.0	23.9	**55.6**	56.3
CLN+KRR	**68.2**	**44.8**	**58.1**	**60.0**

4.3. Retrieval Analysis

Zero shot retrieval is another important task which is not well studied. We use a semantic embedding vector as a query to retrieve test images. Retrieval task uses mean average precision as the performance index. Our approaches compared with the state of the art zero shot retrieval approaches including SSE, JSLE, SynC. Table 3 lists retrieval results in terms of mAP for All datasets using VGG-19 features.

We can see that our approach achieved 53.36% and 61.56% on average compared with the best counterpart of SynC-struct which has a mAP of 51.1%. This again validates that our approach learned more effective representations. CLN based approach obtained even better retrieval performance. The superior performance is due to that **Aggregated** character was successfully transferred on test samples.

4.4. Recognition Results on ST-2 Setting

As discussed in (Xian et al.), standard experiment setting can be flawed. For example, in aPY dataset, 7 test classes monkey, wolf, zebra, mug, building, bag, carriage are among ImageNet 1K classes. ST-2 exchange theses flawed "pre-trained" unseen classes with train classes that are not "pre-trained". We follow this setting when we conduct ST-2 recognition experiment.

For comparison purpose, we follow the feature-extraction process in (Xian et al.), which utilize 2048-d ResNet features for recognition. Our method performs best on all datasets. Among second best approaches, SJE, ALE and DeVISE used large margin mechanism to improve performance. This experiment again proved that learning discriminative representations can be advantageous in zero shot learning.

5. Conclusion

In this paper, we described a simple yet effective representation learning approach for ZSL. Our approach outperforms the state-of-the-art approaches on 4 standard datasets. The main idea of our approach was to leverage supervised information on train data to learn a discriminative representation to achieve class separation. Both Kernerlized Discriminant Analysis and Central-loss neural network based approach was developed to exploit the supervised information that can be transferred to unseen data. Extensive evaluations validated the efficiency of our framework on the conventional ZSL problem. In the future, we plan to improve this work by utilizing the hashing techniques Shen et al. (2016, 2017a,b); Yang et al. (2015); Luo et al. (2017)

to address the large-scale zero-shot recognition problem.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Project 61502081, Project 61673299, Project 61572108 and Project 61602089, and the Open Fund Project of Fujian Provincial Key Laboratory of Information Processing and Intelligent Control (Minjiang University) (No. MJUKF201708).

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al., 2016. Tensorflow: A system for large-scale machine learning, in: Proc. OSDI, pp. 265–283.

Akata, Z., Perronnin, F., Harchaoui, Z., Schmid, C., 2013. Label-embedding for attribute-based classification, in: Proc. CVPR, pp. 819–826.

Akata, Z., Reed, S., Walter, D., Lee, H., Schiele, B., 2015. Evaluation of output embeddings for fine-grained image classification, in: Proc. CVPR, pp. 2927–2936.

Al-Halah, Z., Rybok, L., Stiefelhagen, R., 2016. Transfer metric learning for action similarity using high-level semantics. Pattern Recognition Letters 72, 82–90.

Baudat, G., Anouar, F., 2000. Generalized Discriminant Analysis Using a Kernel Approach. Neural Computation 12, 2385–2404.

Changpinyo, S., Chao, W.L., Gong, B., Sha, F., 2016a. Synthesized classifiers for zero-shot learning, in: Proc. CVPR, pp. 5327–5336.

Changpinyo, S., Chao, W.L., Sha, F., 2016b. Predicting visual exemplars of unseen classes for zero-shot learning. arXiv preprint arXiv:1605.08151 .

Chao, W.L., Changpinyo, S., Gong, B., Sha, F., 2016. An Empirical Study and Analysis of Generalized Zero-Shot Learning for Object Recognition in the Wild, in: Proc. ECCV. Springer, Cham, Cham, pp. 52–68.

De Boom, C., Van Canneyt, S., Demeester, T., Dhoedt, B., 2016. Representation learning for very short texts using weighted word embedding aggregation. Pattern Recognition Letters 80, 150–156.

Deng, J., Berg, A.C., Li, K., Fei-Fei, L., 2010. What Does Classifying More Than 10,000 Image Categories Tell Us?, in: Proc. ECCV, pp. 71–84.

Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L., 2009. ImageNet: A large-scale hierarchical image database, in: Proc. CVPR, pp. 248–255.

Deng, J., Krause, J., Fei-Fei, L., 2013. Fine-grained crowdsourcing for fine-grained recognition. Proc. CVPR , 580–587.

Dinu, G., Lazaridou, A., Baroni, M., 2014. Improving zero-shot learning by mitigating the hubness problem .

Farhadi, A., Endres, I., Hoiem, D., Forsyth, D., 2009. Describing objects by their attributes, in: Proc. CVPRW), pp. 1778–1785.

Frome, A., Corrado, G.S., Shlens, J., Bengio, S., Dean, J., Mikolov, T., et al., 2013. Devise: A deep visual-semantic embedding model, in: Proc. NIPS, pp. 2121–2129.

Hardoon, D.R., Szedmak, S., Shawe-Taylor, J., 2006. Canonical Correlation Analysis: An Overview with Application to Learning Methods. dx.doi.org 16, 2639–2664.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep Residual Learning for Image Recognition , 770–778.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T., 2014. Caffe: Convolutional architecture for fast feature embedding, in: Proc. ACM Multimedia, ACM. pp. 675–678.

Lampert, C.H., Nickisch, H., Harmeling, S., 2009. Learning to detect unseen object classes by between-class attribute transfer, in: Proc. CVPRW, pp. 951–958.

Lampert, C.H., Nickisch, H., Harmeling, S., 2014. Attribute-Based Classification for Zero-Shot Visual Object Categorization. IEEE Trans. Pattern Analysis and Machine Intelligence 36, 453–465.

Larochelle, H., Erhan, D., Bengio, Y., 2008. Zero-data Learning of New Tasks. Proc. AAAI , 646–651.

Luo, Y., Yang, Y., Shen, F., Huang, Z., Zhou, P., Shen, H.T., 2017. Robust discrete code modeling for supervised hashing. Pattern Recognition doi:10.1016/j.patcog.2017.02.034. doi:10.1016/j.patcog.2017.02.034.

Maaten, L.v.d., Hinton, G., 2008. Visualizing data using t-sne. Journal of Machine Learning Research 9, 2579–2605.

Mika, S., Ratsch, G., Weston, J., Scholkopf, B., Mullers, K.R., 1999. Fisher discriminant analysis with kernels, in: Neural Networks for Signal Processing IX: 1999 IEEE Signal Processing Society Workshop, pp. 41–48.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., 2013. Distributed representations of words and phrases and their compositionality, in: Proc. NIPS, pp. 3111–3119.

Mikolov, Tomas, Chen, Kai, Corrado, Greg, Dean, Jeffrey, 2013. Efficient Estimation of Word Representations in Vector Space .

Norouzi, Mohammad, Mikolov, Tomas, Bengio, Samy, Singer, Yoram, Shlens, Jonathon, Frome, Andrea, Corrado, Greg S, Dean, Jeffrey, 2014. Zero-Shot Learning by Convex Combination of Semantic Embeddings, in: Proc. ICLR.

Parikh, D., Grauman, K., 2011. Relative attributes, in: Proc. ICCV, pp. 503–510.

Patterson, G., Xu, C., Su, H., Hays, J., 2014. The SUN Attribute Database: Beyond Categories for Deeper Scene Understanding. International Journal of Computer Vision 108, 59–81.

Qiao, R., Liu, L., Shen, C., Hengel, A.v.d., 2016. Less is More: Zero-Shot Learning from Online Textual Documents with Noise Suppression, in: Proc. CVPR, pp. 2249–2257.

Romera-Paredes, B., Torr, P., 2015. An embarrassingly simple approach to zero-shot learning, in: Proc. ICML, pp. 2152–2161.

Saunders, C., Gammerman, A., Vovk, V., 1998. Ridge regression learning algorithm in dual variables, in: ICML, pp. 515–521.

Shen, F., Mu, Y., Yang, Y., Liu, W., Liu, L., Song, J., Shen, H.T., 2017a. Classification by retrieval: Binarizing data and classifiers, in: Proc. SIGIR, pp. 595–604.

Shen, F., Yang, Y., Liu, L., Liu, W., Dacheng Tao, H.T.S., 2017b. Asymmetric binary coding for image search. IEEE Trans. Multimedia .

Shen, F., Zhou, X., Yang, Y., Song, J., Shen, H.T., Tao, D., 2016. A fast optimization method for general binary code learning. IEEE Trans. Image Processing 25, 5610–5621.

Simonyan, Karen, Zisserman, Andrew, 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition .

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2015. Going deeper with convolutions, in: Proc. CVPR, pp. 1–9.

Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S., 2011. The caltech-ucsd birds-200-2011 dataset .

Wen, Y., Zhang, K., Li, Z., Qiao, Y., 2016. A Discriminative Feature Learning Approach for Deep Face Recognition, in: Proc. ECCV. Springer, Cham, Cham, pp. 499–515.

Xian, Y., Akata, Z., Sharma, G., Nguyen, Q., Hein, M., Schiele, B., 2016. Latent Embeddings for Zero-Shot Classification, in: Proc. CVPR, pp. 69–77.

Xian, Y., Schiele, B., Akata, Z., . Zero-shot learning - the good, the bad and the ugly, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4582–4591.

Xu, X., Shen, F., Yang, Y., Zhang, D., Shen, T., Song, J., 2017. Matrix tri-factorization with manifold regularizations for zeroshot learning, in: Proc. CVPR, pp. 3798–3807.

Yang, Y., Shen, F., Shen, H.T., Li, H., Li, X., 2015. Robust discrete spectral hashing for large-scale image semantic indexing. IEEE Trans. Big Data 1, 162–171.

Zhang, L., Xiang, T., Gong, S., 2017. Learning a Deep Embedding Model for Zero-Shot Learning , 2021–2030.

Zhang, Z., Saligrama, V., 2015. Zero-shot learning via semantic similarity embedding, in: Proc. ICCV, pp. 4166–4174.

Zhang, Z., Saligrama, V., 2016. Zero-Shot Learning via Joint Latent Similarity Embedding, in: Proc. CVPR, pp. 6034–6042.

Zhou, Z., Wang, Y., Wu, Q.J., Yang, C.N., Sun, X., 2017. Effective and efficient global context verification for image copy detection. IEEE Trans. Information Forensics and Security 12, 48–63.

Zhou, Z., Yang, C.N., Chen, B., Sun, X., Liu, Q., QM, J., 2016. Effective and efficient image copy detection with resistance to arbitrary rotation. IEICE Trans. information and systems 99, 1531–1540.