Zero-Shot Recognition via Structured Prediction

Ziming Zhang & Venkatesh Saligrama
ECE, Boston University

Boston University College of Engineering
Department of Electrical & Computer Engineering

Zero-Shot Problem

- **Training**: Learn with annotated *source* and *target* domain data for a subset of classes (**seen** classes).
- **Test-time:** recognize target-domain images for **unseen** classes based on matching to source-domain attributes or descriptors

Traditional Zero-Shot Learning (ZSL) & Zero-Shot Recognition (ZSR)

- **Training:** learn a similarity function, $\kappa(\phi_s(A), \phi_t(I))$, to score similarities between arbitrary source-target domain tuples (A, I), where ϕ_s, ϕ_t denote the (latent) embedding functions for both source and target domains

Learning Methods in Literature:

- ❑ Zhang & Saligrama [CVPR'16] (i.e. [13] in tables): posterior probability of a match given a tuple (A, I)
  - Posterior is a sufficient statistic

$$\kappa(\phi_s(A), \phi_t(I)) = \log\big(\text{Prob}(\text{match}|A, I)\big)$$

  - Learning latent embeddings of both source and target domain data for similarity measure

$$\text{Prob}(\text{match}|A, I) = \sum_{z_A}\sum_{z_I} p(z_A|A)p(z_I|I)p(\text{match}|z_A, z_I)$$

where z_A, z_I denote the source & target domain latent embeddings

- ❑ Akata et al. [CVPR'15] (i.e. [29] in tables): explicitly learn the correlations between source and target features

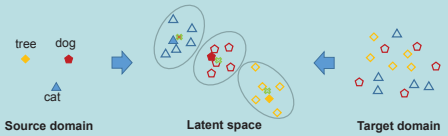
$$\kappa(\phi_s(A), \phi_t(I)) = A^T \boldsymbol{W} I \rightarrow \text{Learn } \boldsymbol{W}$$

- **Test-time:** For each image feature I, identify max scoring source descriptor from a collection of (unseen) classes, and label the test image using the label of the source descriptor.

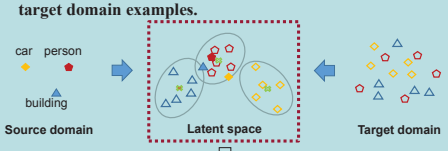
$$y_A = \underset{A}{\text{argmax}}\, \kappa(\phi_s(A), \phi_t(I))$$

Previous Work: Key Insight

- **Training: Source domain is mapped to cluster center in target domain (or in embedded latent domain)** (e.g. Akata et al. [29], Zhang & Saligrama [13, 34])



- **Test-time: Hope that this generalizes to unseen source and target domain examples.**



- **Problem: *Projection Domain Shift***

Test-time unseen source domain data does not map to cluster-centers. This leads to significant misclassification error

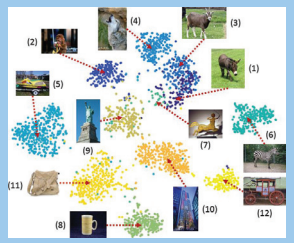
This Paper : Transductive Zero-Shot Recognition

- **Problem:** ***projection domain shift*** between source and target data leads to ZSR degradation.
- **Solution:** Model domain shift as small distortion in ***predicted cluster centers*** based on seen classes. Then solve a ***structured matching problem*** to improve alignment between source and target domain.

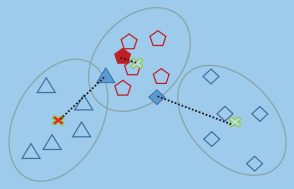
Motivation

a) Observation: CNN features are quite reliable for supervised recognition, as they are clustered quite well for different classes

b) Assumption: There exist strong correlations (e.g. in terms of distance or similarity) between source domain data and cluster centers of target data distributions with different (seen and unseen) classes in latent spaces



(a) t-SNE visualization of CNN feature distributions for 12 unseen classes on aP&Y

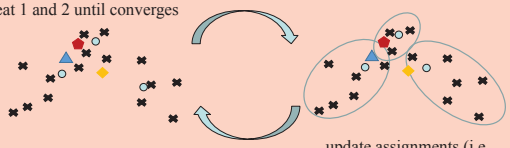


(b) Illustration of distance relationship between source data and their corresponding target cluster centers, which could be used to recover their one-to-one mappings between the two domain data

c) Question: How to Account for Projection Domain Shift in test-time based on -
(1) the learned zero-shot model
(2) Unannotated target instances from unseen classes (i.e., transductive setting)

Our Approach

- **Idea:**
  1. For each unseen test class, estimate its cluster center among unlabeled target data
  2. Then based on the cluster centers, update the assignments of target data instances as the predicted class labels with the help of unseen source domain data
  3. Repeat 1 and 2 until converges



estimate cluster centers in target data

update assignments (i.e. predicted class labels) between target data and cluster centers

- **Structured prediction** **Total source-target similarity** **Total data-class distance**

$$\min_{\boldsymbol{H},\{\boldsymbol{\mu}_{c'}^{(t)}=\boldsymbol{\Phi}_t\boldsymbol{H}_{c'}^T\}_{c'\in\{1,\cdots,C'\}}} \frac{1}{2}\|\boldsymbol{H}\|_F^2 - \lambda_s \sum_{c'=1}^{C'} \boldsymbol{S}_{c'}\boldsymbol{H}_{c'}^T + \lambda_t \sum_{i'=1}^{N'}\sum_{c'=1}^{C'} H_{c',i'} \left\|\phi_t\left(x_{i'}^{(t)}\right) - \boldsymbol{\mu}_{c'}^{(t)}\right\|_2^2$$

$$s.t.\, H_{c',i'} \geq 0, \sum_{c'=1}^{C'} H_{c',i'} \neq 0, \sum_{i'=1}^{N'} H_{c',i'} = 1, \forall i', \forall c', \sum_{i'=1}^{N'} H_{c'_m,i'} H_{c'_n,i'} = 0, \forall c'_m \neq c'_n$$

one instance to one cluster (i.e. unseen class) constraints

where

- ❑ $\boldsymbol{H}$: assignment matrix (i.e. predicted class labels) between unseen target data and classes
- ❑ $\boldsymbol{S}$: (target) data-class similarity matrix computed based on pre-learned zero-shot models
- ❑ $\boldsymbol{\Phi}_t = \left[\phi_t\left(x_{i'}^{(t)}\right)\right]_{\forall i'}$: a target (latent) embedding matrix
- ❑ $\boldsymbol{\mu}_{c'}^{(t)}, \forall c'$: cluster center vector for class c' in target data

Experiments

Exp I: Zero-Shot Recognition

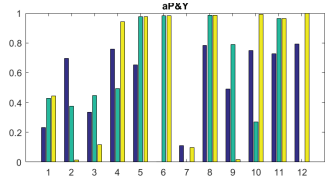
Table 1. Zero-shot recognition accuracy comparison (%) using CNN features in the form of "mean±standard deviation". Here numbers for the comparative methods are cited from the original papers, and "-" means no repeated result available yet.

Method	aP&Y	AwA	CUB	SUN	Ave.
Akata *et al.* [29]	-	61.9	40.3	-	-
Lampert *et al.* [4]	38.16	57.23	-	72.00	-
Fu *et al.* [15]	-	80.5	47.9	-	-
Kodirov *et al.* [14]	-	75.6	40.2	-	-
Zhang & Saligrama [34]	46.23±0.53	76.33±0.83	30.41±0.20	82.50±1.32	58.87
Romera-Paredes & Torr [27]	24.22±2.89	75.32±2.28	-	82.10±0.32	-
ZS similarities [27] + This paper	37.5	84.3	-	***89.5***	-
BL-ZSL (*i.e.* Denoising version of [29])	39.45	70.45	39.58	84.00	58.37
BL-ZSL similarities + This paper	***69.74±3.47***	92.06±0.18	53.26±1.04	86.01±1.32	***75.27***
Zhang & Saligrama [13]	50.35±2.97	79.12±0.53	41.78±0.52	83.83±0.29	63.77
ZS similarities [13] + Label Propagation [16]	58.7	82.6	50.2	84.0	68.9
ZS similarities [13] + This paper	62.19±4.65	***92.08±0.14***	***55.34±0.77***	86.12±0.99	73.93

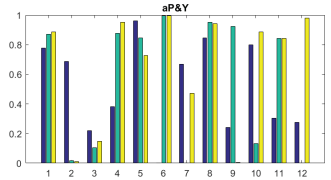
Table 2. Average precision and recall comparison (%) for recognition.

Precision	aP&Y	AwA	CUB	SUN	Ave.
Zhang & Saligrama [13]	52.70±27.33	81.70±14.67	54.06±24.13	82.51±12.24	67.74
ZS similarities [13] + This paper	55.96±35.72	***91.37±14.75***	***57.09±27.91***	85.96±10.15	72.59
BL-ZSL similarities + This paper	***62.80±42.67***	91.37±14.83	51.10±29.66	***86.12±9.78***	***72.84***
Recall (*i.e.* class accuracy)					
Zhang & Saligrama [13]	51.34±29.69	72.14±26.29	45.05±26.16	82.00±16.31	62.63
ZS similarities [13] + This paper	54.66±42.27	***90.28±8.08***	***55.73±31.80***	***86.00±13.19***	71.67
BL-ZSL similarities + This paper	***65.36±37.29***	90.25±8.09	53.30±33.39	86.00±14.97	***73.73***

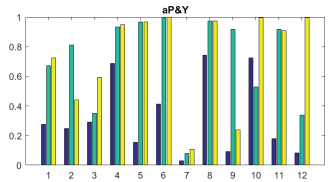
- Class-level **precision** comparison



- Class-level **recall** comparison



- Class-level **AP** comparison

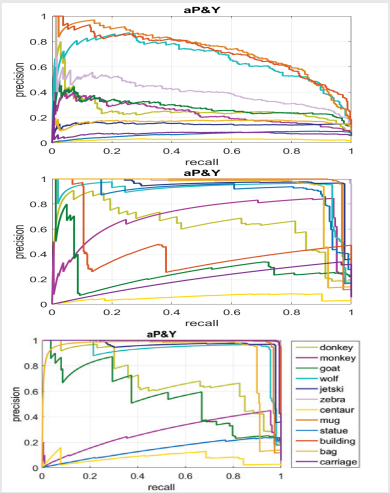


(**blue**) [13] (**green**) [13]+SP-ZSR (**yellow**) BL-ZSL+SP-ZSR

Exp II: Zero-Shot Retrieval

Table 3. mAP comparison (%) for zero-shot retrieval.

Method	aP&Y	AwA	CUB	SUN	Ave.
Zhang & Saligrama [13]	32.69	66.56	23.93	76.48	49.92
ZS similarities [13] + This paper	70.70	***94.03***	***63.25***	***92.17***	***80.04***
BL-ZSL similarities + This paper	***74.11***	92.05	58.76	91.68	79.15



(**top**) [13] (**middle**) [13]+SP-ZSR (**bottom**) BL-ZSL+SP-ZSR

Reference:

[29] Akata et al. "Evaluation of output embed-dings for fine-grained image classification", in CVPR, 2015.
[34] Zhang and Saligrama. "Zero-shot Learning via Semantic Similarity Embedding", in ICCV, 2015.
[13] Zhang and Saligrama, "Zero-Shot Learning via Joint Latent Similarity Embedding", in CVPR, 2016.

Code Available: https://zimingzhang.wordpress.com/