What Visual Attributes Characterize an Object Class ?

Jianlong Fu[1⋆], Jinqiao Wang[1], Xin-Jing Wang[2], Yong Rui[2], Hanqing Lu[1]

[1]National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, No.95, Zhongguancun East Road, Beijing, 100190, China
[2]Microsoft Research, No.5, Dan Ling Street, Haidian District, Beijing 10080, China
[1]{jlfu, jqwang, luhq}@nlpr.ia.ac.cn, [2]{xjwang, yongrui}@microsoft.com

Abstract. Visual attribute-based learning has shown a big impact on many computer vision problems in recent years. Albeit its usefulness, most of works only focus on predicting either the presence or the strength of pre-defined attributes. In this paper, we discuss how to automatically learn visual attributes that characterize an object class. Starting from the images of an object class that are collected from the Web, we first mine visual prototypes of attributes (i.e., a clean intermediate representation for learning attributes) by clustering with Gaussian mixtures from multi-scale salient areas in noisy Web images. Second, a joint optimization model is proposed to fulfill the attribute learning with feature selection. As sparse approximation is adopted for feature selection during the joint optimization, the learned attributes tend to present a more representative visual property, e.g., stripe pattern (when texture features are selected), yellow-color (when color features are selected). Finally, to quantify the confidence of attributes and restrain the noisy attributes learned from the Web, a ranking-based method is proposed to refine the learned attributes. Our approach ensures the learned visual attributes to be visually recognizable and representative, in contrast to manually constructed attributes [1] that contain properties difficult to be visualized, e.g., "smelly," "smart." We evaluated our approach on two benchmark datasets, and compared with state-of-the-art approaches in two aspects: the quality of the learned visual attributes and their effectiveness in object categorization.

1 Introduction

A visual attribute presents a certain type of property (e.g., striped, yellow, long-neck) that can describe an object class [2]. Recent research on visual attributes has shown a big impact on both research achievements and practical applications, e.g., face verification [3], image retrieval [4][5], object recognition [6][7], and adopting attributes such as size, color to refine search results by commercial search engines.

However, existing approaches on attribute learning and attribute-based object recognition generally work on the pre-defined vocabulary of attributes, and the task is to predict the presence or relative strength of an attribute in an image or an object class [8][9]. Few works were done to automatically generate or discover attributes so that images of this class can be discriminated from images of other object classes when projected into a more specific and representative attribute space. Moreover, few works were done which

⋆ This work was conducted when Jianlong Fu was a research intern at Microsoft Research.

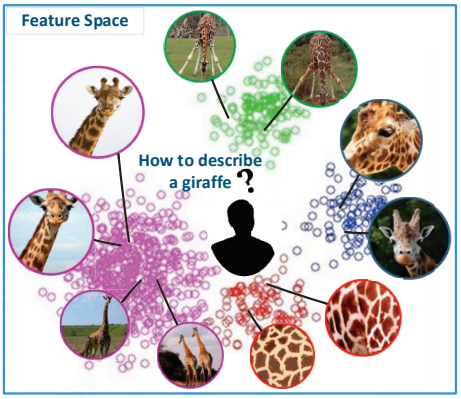

Fig. 1. An illustration of the attribute learning of "giraffe." We start with a large collection of Flickr images (small circles) and produce attributes with example images (large circles) indicating that a giraffe has a long neck, deer-like face, stripes and four legs.

considered the "visualness" of an attribute[1], and generated only *visual attributes* that could be effectively modeled with low-level visual features. Osherson and Wilkie collected 85 attributes of 48 animal classes via manual judgments [1], but not all of the attributes are visual, e.g., "smelly."

In this paper, we propose an unsupervised approach to learn the visual attributes that characterize an object class. That is, given an object class with its associated Web images (e.g., Flickr images), our approach outputs a ranked list of visual attributes that capture the key properties of the object class, where a rank score suggests the confidence for each attribute. Fig. 1 shows a few attributes our approach learned from $5,000$ Flickr images of "giraffe." It is clear that some attributes are visually recognizable and can present the property of long-neck, skin patterns, deer-like face and four legs, though we don't focus on assigning semantics in this work.

Learning from the Web has demonstrated great success due to the huge quantities of images and unlimited vocabulary [11]. However, the challenge for learning attributes from noisy Web images can derive from two aspects. First, Web images often consist of both main objects and complex background. Second, the text to image association is far less controlled. For example, an image may be irrelevant to its user-contributed tag in Flickr. To solve the two problems, we propose an approach with three steps: 1) A Gaussian mixture model (GMM) [12] is first applied onto the multi-scale salient areas of a certain class images from Flickr, which generates visual prototypes of attributes, with each Gaussian one prototype. The intuition of building the visual prototypes is to reduce the background noises from Flickr images, which ensures a good intermediate representation for the attribute learning of a targeted object class. 2) Visual prototypes with specific properties are further learned and represented as attributes, where each attribute is an ensemble of Gaussian mixtures on the selected features. 3) Each attribute is ranked according to a confidence score by accumulating the rank scores from its

[1] Visualness [10] is a quantitative measure of how likely a concept can be visualized with example images.

contained visual prototypes. Thus, noisy attributes learned from irrelevant images can be restrained by low scoring.

We conducted comprehensive evaluations of the approach on two standard datasets, Animal with Attributes [13] and PASCAL VOC 2007. The evaluations show the effectiveness of our approach, which not only learned clean and intuitive attributes of object classes (e.g., the attribute of "long neck" is ranked at the top for giraffe and "stripe" is ranked at the top for zebra), but also achieved higher accuracy in object categorization, compared to state-of-the-art approaches.

The **main contribution** of this work is the unsupervised data-driven approach which automatically learns visual attributes from noisy Web images. Specifically, 1) the proposed visual prototype and attribute ranking scheme can effectively reduce the impact of noises from Web images. 2) The design of spectral analysis with feature selection ensures that the learned attribute can reflect a more representative visual property against previous approaches. 3) This approach is highly efficient and scalable as the training data is directly collected from the Web without any human cleanup.

2 Related Work

In this section, we review some works related to ours in two categories, i.e., pre-defined attributes and data-driven attributes.

Learning pre-defined attributes: A large body of works on attribute learning are based on pre-defined attributes. The list of pre-defined attributes can be generally formed by human [13][14] or mining online text [15]. Li *et al.* [16] and Torresani *et al.* [17] both consider the output of many object class classifiers of pre-defined categories as attributes for high-level visual recognition. To utilize the rich data on the Web, Ferrari *et al.* [18] learn visual models of a list of given attributes by Web image search results. A similar work is done by Tamara *et al.* [15] who automatically mine both texts and images on the Web to recognize attributes, thus it can dramatically alleviate human efforts. However, the pre-defined attribute lists crawled from the Web or collected from existing classifiers are limited and cannot be discriminative to a new specific categorization task. Besides, some are even not predictable by visual features, e.g., "smell."

Learning data-driven attributes: As the pre-defined attributes cannot fully discover specific properties for an object class, data-driven attributes have been proposed to learn attributes from data itself. Yang *et al.* [19] propose an automatic event detection approach from a large collection of unconstrained videos using data-driven approaches. Jingen *et al.* [20] automatically infer data-driven attributes from training data using an information theoretic approach. Yu *et al.* [2] and Wang *et al.* [6] design discriminative attribute learning approaches to improve object recognition, where the large-margin framework and latent models are adopted, respectively.

Compared to previous data-driven approaches where the attributes are considered as bag-of-words representation in [19], latent variables in [6] or linear model in [2], our learned attributes are visually recognizable and tend to present a more representative visual property with an importance rank score since features are selected in optimization processes. Meanwhile, we specially design the generating of visual prototypes and the

scheme of attribute ranking to reduce the impact of noises from Web images, instead of using them directly as in [17] and [18].

3 Unsupervised Visual Attributes Learning

In this section, we present the details of the proposed unsupervised attributes learning approach for an object class. The framework is shown in Fig. 2. As we can observe from this figure, our approach takes as input a set of noisy Flickr images associated with an object class (shown in (a)) and returns as output a series of ranked attributes (shown in (e)). To achieve this goal, we first generate the visual prototypes from multi-scale salient areas (shown in (b) and (c)). Second, a joint optimization model is conducted to do the attribute learning with feature selection (shown in (d)). Finally, we compute a confident score for each attribute by accumulating the scores from its contained visual prototypes in the selected feature space.

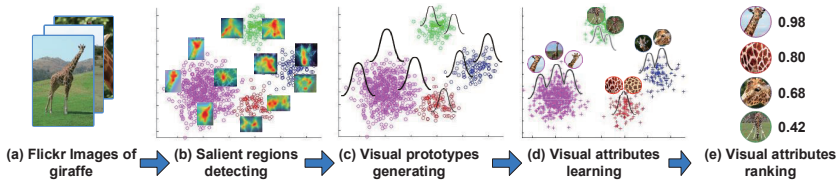

Fig. 2. The proposed unsupervised visual attributes learning approach. In (b), the detected regions are projected into the low-level feature space. In (c), visual prototypes are first learned from multi-scale salient regions in the original feature space. In (d), the attributes are further discovered from visual prototypes in the selected feature space which guarantees to present a certain property, e.g., long-neck (when shape features are selected), striped (when texture features are selected).

3.1 Visual Prototypes Generating

To reduce the background interference and ensure that the object-located region of an image can be selected for attribute learning, we propose a method of generating visual prototypes. A visual prototype is defined as a clean intermediate representation for attribute learning. First, we adopt a saliency detection approach [21] with multiple parameters, which generates a series of salient regions with different scales in an image. As it is hard to determine which scale the object can locate in, we cluster multi-scale salient areas by Gaussian mixture models and use the Gaussian mean with covariance matrix as the visual prototypes. This idea has an intuitive explanation that the object-located region can be determined by a soft voting from the multi-scale salient areas. Considering there can be similar images in the set of Flickr images associated with an object class, this clustering is conducted on the salient regions of all training images.

The learning model is given in Eq.1:

$$\begin{aligned} G(x|\boldsymbol{\omega}) &= \sum_{i=1}^{K} \pi_i G_i(x|\omega_i) \\ &= \sum_{i=1}^{K} \pi_i \frac{1}{(2\pi)^{D/2}|\Sigma_i|^{1/2}} exp\{-\frac{1}{2}(x-\mu_i)^T \Sigma_i^{-1}(x-\mu_i)\} \end{aligned} \tag{1}$$

where x is the visual feature of a salient region and $G_i(x|\omega_i)$ is a visual prototype. $\boldsymbol{\omega} = \{\omega_1, ..., \omega_K\} = \{(\pi_1, \mu_1, \Sigma_1), ..., (\pi_K, \mu_K, \Sigma_K)\}$ denotes the parameter set of G. D is the dimension of x. π_i is the weight of each component, $\pi_i >= 0$ and $\sum_{i=1}^{K} \pi_i = 1$. $\boldsymbol{\mu} = \{\mu_1, ..., \mu_K\}$ and $\boldsymbol{\Sigma} = \{\Sigma_1, ..., \Sigma_K\}$. μ_i is the mean vector and Σ_i is the covariance matrix of the i^{th} component.

The two parameters in Eq.1, $\boldsymbol{\omega}$ and K, need to be optimized. First, we use the Expectation Maximization (EM) algorithm [22] to estimate $\boldsymbol{\omega}$. The EM algorithm is a general method of finding the maximum-likelihood estimate of the parameters of an underlying distribution from a given data set. Second, as K affects the descriptive ability of visual prototypes, we apply an n-fold cross-validation approach to determine the best K, rather than setting it empirically. We separate training data into n pieces and pick $n-1$ pieces of data to estimate $\boldsymbol{\omega}$. Then we calculate the loglikelihood on the rest one piece of data. The procedure is performed n times and produces an average loglikelihood. Previous work [12] has demonstrated that with the increasing number of K, the average loglikelihood increases but seems to be converging to some upper bound. It also shows that a small number of K is indeed insufficient to achieve good performance. Therefore, we increase K starting from 50 to $+\infty$ and stop when the difference of loglikelihoods between two successive calculations is smaller than a threshold (denoted as T_1).

3.2 Visual Attributes Learning

Once we have obtained the reliable intermediate representations, i.e., visual prototypes, we further learn visual attributes by a joint optimization with feature selection. The visual attributes are learned from a set of similar prototypes in a selected feature space, which ensures that the learned attributes can describe a certain visual property for an object class, e.g., round (when shape features are selected), striped (when texture features are selected). Note that whether presenting a specific property is the key difference between attributes and prototypes, while prototypes just represent the appearances of main objects in images.

For a set of visual prototypes $G = \{G_1, G_2, ..., G_K\}$, each prototype G_i is represented by (μ_i, Σ_i). A full-connected graph is constructed between any two prototypes to reflect a global structure on Gaussian space. Specifically, a label matrix $Y \in R^{K \times C}$ is defined as $y_{i,j} = 1$ if the i^{th} prototype can be grouped into the j^{th} attribute, otherwise 0, where C denotes the number of attributes. Furthermore, to find the specific visual property in attribute representations, a feature selection matrix $W \in R^{D \times C}$ is leveraged.

A Joint Objective Function Given a spectral clustering term $\mathcal{F}(Y)$ and a feature selection term $\mathcal{L}(Y, W)$, the joint objective function is proposed as:

$$\begin{aligned} &\min_{Y,W} \mathcal{F}(Y) + \mathcal{L}(Y, W) \\ =&\min_{Y,W} \mathrm{Tr}[Y^T L Y] + \alpha(||\boldsymbol{\mu}^T W - Y||_F^2 + \beta||W||_{2,1}) \\ &s.t. \quad Y^T Y = I_C, Y \geq 0 \end{aligned} \tag{2}$$

where α and β are two nonnegative parameters.

In the spectral clustering term, an effective affinity matrix is obviously beneficial to reflect the relationship among different visual prototypes. As each prototype is a Gaussian, we use KL divergence to depict this relationship. To construct the affinity matrix $S \in R^{K \times K}$, we define:

$$S_{i,j} = exp\{-\frac{KL(i,j)^2}{\sigma^2}\} \tag{3}$$

where σ is a free parameter to control the decay rate and KL divergence between two prototypes has a closed formed expression:

$$KL(i,j) = \frac{1}{2}[\log\frac{|\Sigma_j|}{|\Sigma_i|} + \mathrm{Tr}[\Sigma_j^{-1}\Sigma_i] - D + (\mu_i - \mu_j)^T \Sigma_j^{-1}(\mu_i - \mu_j)] \tag{4}$$

Then the spectral clustering term is defined as minimizing the following formula:

$$\mathcal{F}(Y) = \frac{1}{2}\sum_{i,j=1}^{K} S_{ij}||\frac{y_i}{\sqrt{A_{ii}}} - \frac{y_j}{\sqrt{A_{jj}}}||_2^2 = \mathrm{Tr}[Y^T L Y] \tag{5}$$

where A is a degree matrix defined as the diagonal matrix with the degrees $a_1, ..., a_K$ on the diagonal. $a_i = \sum_{j=1}^{K} S_{i,j}$ and $L = I - A^{-1/2} S A^{-1/2}$.

In the feature selection term, mean vector μ_i is used to describe the visual appearance of the i^{th} prototype. The l_{21}-norm is defined as $||W||_{2,1} = \sum_{i=1}^{D}\sqrt{\sum_{j=1}^{C} W_{i,j}^2}$, which is viewed as a regularization term to ensure the sparsity of W in row. We constrain one prototype can be grouped to one visual attribute, therefore an orthogonal constraint is imposed. Besides, to make Y more accurate and discriminative, a nonnegative constraint is also introduced. Both the orthogonal and nonnegative constraints guarantee that there is only one element in each row of Y that is much larger than zero and the others tend to be zeroes.

In the optimization process, on one hand, the spectral clustering learns the pseudo cluster labels. On the other, to minimize the overall loss, the algorithm automatically searches the most discriminative features to pseudo cluster labels and learns feature selection matrix W.

Optimization Note that the l_{21}-norm is non-smooth and the objective function is not convex for W and Y simultaneously, then an efficient iterative optimization strategy is applied. First, we relax the orthogonal term and rewrite the optimization problem as:

$$\begin{aligned}
&\min_{Y,W} \mathcal{F}(Y) + \mathcal{L}(Y,W) \\
=&\min_{Y,W} \mathrm{Tr}[Y^T L Y] + \alpha(||\boldsymbol{\mu}^T W - Y||_F^2 + \beta||W||_{2,1}) \\
&+ \frac{\gamma}{2}||Y^T Y - I_C||_F^2 \\
&s.t. \quad Y \geq 0
\end{aligned} \tag{6}$$

where $\gamma \geq 0$ is a parameter to control the orthogonal constraint. It can be set large enough to ensure the constraint satisfied as in [23]. Following [23] and [24], we define $\mathscr{F}(Y,W) = \mathcal{F}(Y) + \mathcal{L}(Y,W)$. Setting $\frac{\partial \mathscr{F}(Y,W)}{\partial W} = 0$, we have:

$$\begin{aligned}
\frac{\partial \mathscr{F}(Y,W)}{\partial W} &= 2\alpha(\boldsymbol{\mu}(\boldsymbol{\mu}^T W - Y) + \beta B W) = 0 \\
&\Rightarrow W = (\boldsymbol{\mu}\boldsymbol{\mu}^T + \beta B)^{-1}\boldsymbol{\mu} Y
\end{aligned} \tag{7}$$

Here B is a diagonal matrix with $B_{ii} = \frac{1}{2||w_i||_2}$. Representing W by Eqn. 7, Eqn. 6 is induced as:

$$\min_{Y} \mathrm{Tr}[Y^T Z Y] + \frac{\gamma}{2}||Y^T Y - I_C||_F^2 \quad s.t. \quad Y \geq 0 \tag{8}$$

where $Z = L + \alpha[I_K - \boldsymbol{\mu}^T(\boldsymbol{\mu}\boldsymbol{\mu}^T + \beta B)^{-1}\boldsymbol{\mu}]$ and $I_K \in R^{K \times K}$ is an identity matrix. Then we introduce multiplicative updating rules. Letting $\phi_{i,j}$ be the Lagrange multiplier for constraint $Y_{ij} \geq 0$ and $\Phi = [\phi_{ij}]$, the lagrange function is:

$$\mathrm{Tr}[Y^T Z Y] + \frac{\gamma}{2}||Y^T Y - I_C||_F^2 + \mathrm{Tr}(\Phi Y^T) \tag{9}$$

Setting its derivative of Y to zero and using the KKT condition where $\phi_{ij} Y_{ij} = 0$, Y can be updated according to the following rules:

$$Y_{ij} \leftarrow Y_{ij} \frac{(\gamma Y)_{ij}}{(ZY + \gamma Y Y^T Y)_{ij}} \tag{10}$$

Then Y is normalized by $(Y^T Y)_{ii} = 1, i = 1, ..., K$. Convergence of the iterative algorithm can be proven in [23].

3.3 Visual Attributes Ranking

The resultant visual attribute models are the clusters of visual prototypes with selected features. Each cluster represents one attribute representation. We first describe the visual prototypes in the selected feature space, then the generating of visual attributes with ranking scheme is further proposed.

After the above optimization, the position of zero rows in W indicates the position of the feature dimensions which are not discriminative and can be abandoned. Therefore, we delete the related rows and columns of μ_i and Σ_i in G_i to obtain the new prototype G_i' on the selected features, where these related rows and columns correspond to the abandoned feature dimensions. Then we recalculate the KL divergence as $KL(i,j)'$ and the affinity matrix as S'.

Let the columns of S' be a standard simplex [25], then S' has the largest eigenvalue equal to one and a real eigenvector r^*. The ranking process can be achieved according to spectral analysis by solving the following objective function:

$$r^* = \arg_r \min ||S'r - r||_2^2 \tag{11}$$

here r^* contains all the rank scores for each new prototype G'_j with selected features and the optimization can be solved by iterative method [26]. Then the rank score for each attribute is defined as:

$$R(M_i) = \sum_{j=1}^{|N_i|} r_j^* \tag{12}$$

where M_i denotes the i^{th} attribute and r_j^* denotes the rank score of G'_j which is selected from r^*. This sum runs over the scores of all the prototypes grouped to M_i. $|N_i|$ measures the size of M_i by its contained prototypes. Attributes are ranked by $R(M_i)$ with decreasing order. Each produced attribute is a Gaussian mixture, which is presented as the ensemble of visual prototypes with their ranks:

$$M_i = \sum_{j=1}^{|N_i|} r_j^* * G'_j \tag{13}$$

where $i = 1, ..., C$. The complete unsupervised attributes learning algorithm is summarized in Algorithm 1.

4 Experiments

In this section, we evaluated the proposed approach on two aspects: the quality of the learned visual attributes for each object class and their effectiveness for object categorization tasks.

4.1 Datasets

For attribute learning, we collected $5,000$ images from Flickr for each object class by searching user-contributed tags. We extracted features of color (RGB color histogram), texture [27], shape (PHOG [28] and self-similarity histograms [29]). Different features were normalized and concatenated into a feature vector with the dimension of 1073.

For object categorization, we trained classification models and evaluated them on two datasets. One is Animal with Attributes (AwA) [13] which contains $30,475$ images of 50 animal object classes. The other is PASCAL VOC 2007 which consists of $9,963$ images of 20 different object classes.

4.2 Experiment Settings

Parameter Settings for Attribute Learning There are two key parameters in saliency detection [21], i.e., "sigma" and "level. " The former one controls the spatial spread of weights between different image locations. The latter one controls the resolution of the

Algorithm 1 Unsupervised Visual Attributes Learning

Input: Noisy Flickr images given an object class
parameters $K, \boldsymbol{\omega}, \alpha, \beta, \gamma$
1.Saliency detection and feature extraction
2.Visual prototypes generating by Eqn. 1
$\boldsymbol{\omega}$ is determined by EM algorithm
K is determined by cross-validation
3.Visual attributes learning by solving Eqn. 2
The iteration step $t = 1$
Initialize $Y \in R^{K \times C}$ and $W \in R^{D \times C}$
Set $B^{(t)} \in R^{D \times D}$ as an identity matrix
Repeat:
$Z^{(t)} = L + \alpha[I_K - \boldsymbol{\mu}^T(\boldsymbol{\mu}\boldsymbol{\mu}^T + \beta B^{(t)})^{-1}\boldsymbol{\mu}]$
$Y_{ij}^{(t+1)} = Y_{ij}^{(t)} \frac{(\gamma Y^{(t)})_{ij}}{(Z^{(t)}Y^{(t)} + \gamma Y^{(t)}(Y^{(t)})^T Y^{(t)})_{ij}}$
$W^{(t+1)} = (\boldsymbol{\mu}\boldsymbol{\mu}^T + \beta B^{(t)})^{-1}\boldsymbol{\mu}Y^{(t+1)}$
update $B^{(t+1)}$ with $B_{ii}^{(t+1)} = \frac{1}{2||w_i^{t+1}||_2}$
$t = t + 1$
Until Convergence or $t = 500$
4. Visual attributes ranking
calculate S'
$r^* = \arg_r \min ||S'r - r||_2^2$
$R(M_i) = \sum_{j=1}^{|N_i|} r_j^*$
$M_i = \sum_{j=1}^{|N_i|} r_j^* * G'_j$
Output: Attributes for the object class

feature map. To produce multi-scale salient regions, the range of "sigma" is set from 0.1 to 1.0 with the step of 0.1, and the "level" is set as $[2, 3, 4]$ and $[5, 6, 7]$. Hence, there are totally 20 groups of parameters that can generate 20 salient regions with different scales for an image in attribute learning.

In the visual prototype generating, we set the threshold T_1 as 0.01 and our experiments showed that K varied from 1,000 to 2,000 for most object classes. In the visual attribute learning, σ in Eqn. 3 is set to 2 empirically. The three parameters α, β, γ should be determined in Eqn. 6. γ is set to be 10^8 to ensure the orthogonal constraint as used in [23]. To evaluate the effect of α and β, a ratio is defined between intra-attribute similarity and inter-attribute similarity as:

$$Ratio = \frac{S(intra\_attribute)}{S(inter\_attribute)} \tag{14}$$

where $S(intra\_attribute)$, $S(inter\_attribute)$ can be obtained by calculating the sum of similarity of any two prototypes within any attribute (i.e., KL distance between Gaussians) and the sum of the similarity of any two attributes (i.e., KL distance between Gaussian mixtures), respectively. α and β are set to $\{1, 10, 10^2, 10^3, 10^4, 10^5, 10^6\}$. The results on the classes of AwA are shown in Fig. 3. $\alpha = 10^2$ and $\beta = 10^4$ were chosen when the ratio reached the highest value.

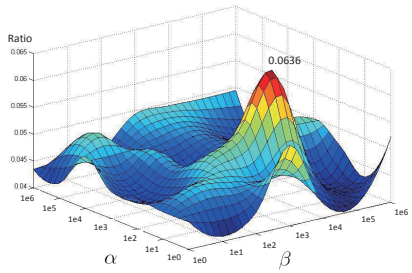

Fig. 3. Parameters setting for α and β for 50 object classes in AwA.

Compared Approaches for Object Categorization The following approaches are compared for performance evaluation of object categorization tasks.

1. low-level feature: a typical image representation approach with the low-level features described in Section 4.1.
2. Classemes [17]: an approach using the output of existing object class classifiers of pre-defined categories as attributes.
3. category-level attribute designing approach (CLA) [2]: an automatic attribute learning approach with large-margin framework.
4. LDA-based [30] attribute learning approach: an automatic attribute learning approach using latent dirichlet allocation to generate attributes for each object class.

Note that attribute-based approaches leverage the outputs on different attributes as features, e.g., the output of classifiers in Classemes [17] and CLA [2], the response of topics in LDA [30] and the response of attributes (i.e., Gaussian mixtures) in our approach. Classemes was implemented using the author-released code. We implemented CLA as in [2]. The LDA-based attribute learning approach was trained on noisy Flickr images as ours, with each topic one attribute. We used a non-linear SVM (χ^2 kernel) as the classification model. The training, testing and validation images were selected according to [2] and [31] for AwA dataset and PASCAL VOC 2007 dataset, respectively.

4.3 Attribute Learning Results

We first showed the attribute learning results of our unsupervised approach by fixing the number of attributes (i.e., C in Eqn. 13) to 30, as the performance cannot increase with larger numbers examined in the following sections. For different object classes, we visualized the attributes ranked in No.1, No.5 and No.30 by showing the top five salient regions from Flickr images with the highest responses to each attribute. The results are shown in Fig. 4 and Fig. 5. For an attribute, the higher the score, the more representative it is. As we can observe from Fig. 4, the most representative attribute for "Deer" is the salient "antler," which can discriminate deer from other animals such as sheep or horse. And their body postures are ranked in the fifth place. As we expected, "long neck" and "stripe texture" are the two most representative attributes for "giraffe" and "zebra," respectively. As an interesting discovery, the last image of the top attribute of zebra is actually a pedestrian crossing due to its similar visual appearance to the skin of zebras. For "Tiger," "Persian cat" and "Chihuahua," the results indicate that their facial cues are the most representative attributes, followed by body postures or textures.

It is reasonable because it is often difficult for humans to separate cat and dog only by their furry body, but we can easily recognize them by their faces.

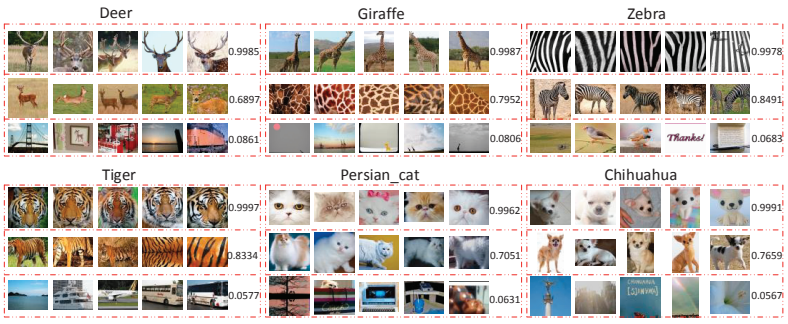

Fig. 4. Attribute learning results for six object classes of AwA dataset. For each object class, we visualize attributes ranked in No.1, No.5, No.30 within the 30 attributes by showing the top five salient regions from Flickr images with their rank scores.

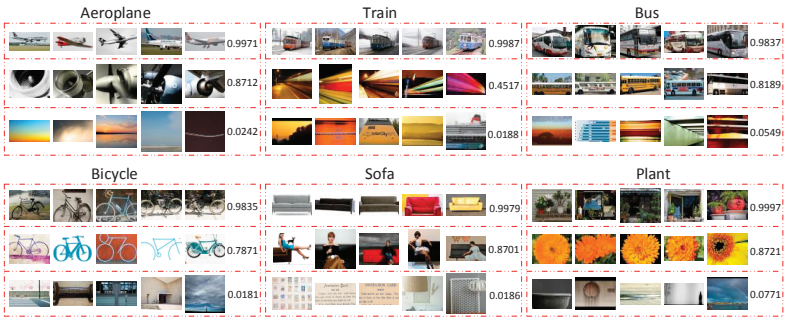

Fig. 5. Attribute learning results for six object classes of PASCAL VOC 2007 dataset (The illustration is same as Fig. 4).

From Fig. 5, we can observe some implicit or even social attributes. For example, the learned attribute located in the second row of "Train" can be interpreted as "fast." For "Sofa," the second-row attribute reveals that the sofa is a kind of furniture with its specific function that human can comfortably sit on it. In addition, our approach can greatly weaken the noisy attributes which correspond to irrelevant images for an object class. For instance, the third-row attributes ranked in No.30 for all classes have the lowest rank scores, which reflect those appearances of irrelevant images. The role of these attributes can be neglected as the responses on them can approach to zero.

Moreover, we examined the most discriminative visual features for different object classes and showed the results in Tab. 1 and Tab. 2. These numbers are obtained by calculating the percentage of non-zero rows of each feature type in the feature selection matrix W. The higher percentages, the more important role the feature type plays. Taking Tab. 1 as an example, the result shows that texture is the most discriminative feature type for "Zebra." While for "Dear," "Giraffe," "Persian cat" and "Chihuahua," shape is

the most discriminative one, which is consistent with our observation from Fig. 4. It is reasonable that the texture feature can well reflect the "stripe pattern" of zebra, the shape feature can reveal the salient contour of "antler" and "long neck" for dear and giraffe, as well as the facial cues for persian cat and chihuahua.

Table 1. The analysis of feature selection for six object classes in AwA.

	color	texture	shape
Deer	0.55	0.47	**0.97**
Giraffe	0.6	0.47	**0.97**
Zebra	0.55	**0.98**	0.33
Tiger	**0.65**	0.44	0.12
Persian_cat	0.45	0.14	**0.93**
Chihuahua	0.55	0.12	**0.96**

Table 2. The analysis of feature selection for six object classes in PASCAL VOC 2007.

	color	texture	shape
Aeroplane	0.52	0.44	**0.85**
Train	0.45	0.16	**0.81**
Bus	0.49	0.42	**0.82**
Bicycle	0.45	0.34	**0.87**
Sofa	**0.81**	0.47	0.75
Plant	**0.83**	0.45	0.73

4.4 Object Categorization on AwA

In this part, we showed the superiority of the learned attributes by applying them to the task of object categorization on AwA dataset. We first conducted an experiment on 40 known classes. Each object class produced attributes with the number of C, and thus there were totally attribute features of $40 * C$ dimensions. We examined the influence of the number of attributes C of an object class. We can observe from Fig. 6(b), C increases from 10 to 40. There is no significant accuracy improvement when the C reaches 30, compared to the performance of $C = 40$. Therefore, we consider 30 attributes can effectively cover the properties of an object class and we keep this number for all object classes in the following experiments.

To compare the performance for object categorization, different compared approaches were constrained to produce the attribute features of the same dimensions. For example, Classemes can generate attribute features of $2,659$ dimensions. We used PCA to reduce this feature representation to the dimension of $1,200$. For LDA-based attribute learning approach, we produced 30 attributes for each object class and generated $1,200$ attributes for the 40 classes as ours.

Fig. 6(a) shows the comparison result with different attribute learning approaches. We can observe the following conclusions. First, attribute-based approaches can achieve higher accuracy against the low-level-feature-based approach when we have enough training samples, e.g., 30 or 50. Second, our approach surpasses Classemes with predefined attributes, which demonstrates the superiority of the data-driven attribute learning approach that can detect specific attributes from data itself. Third, compared to CLA and LDA, the proposed approach consistently achieves better performance. The reason can be concluded in two folds. On one hand, the feature selection scheme ensures to present a certain type of properties, which enhance the discrimination ability in

object categorization against the CLA approach. On the other, our approach can effectively weaken the noisy attributes learned from irrelevant images and provide a cleaner attribute representations against the LDA-based approach, which directly learns the attribute from noisy Web images.

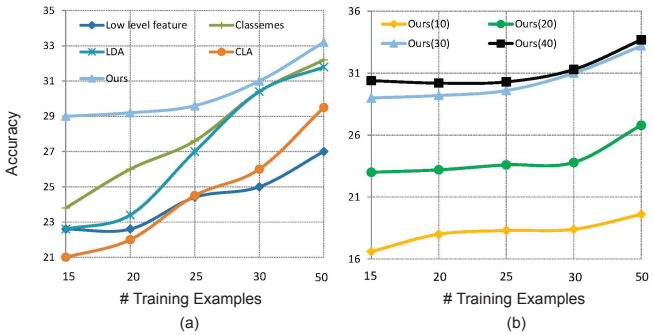

Fig. 6. Multi-class classification on 40 known object classes. (a) shows the accuracy of various approaches with the increasing of training examples. (b) shows the influence of different numbers of attributes in an object class to the classification results (the numbers are in brackets).

To show the performance for 10 novel classes defined in [13], we conducted an interesting experiment which projected images of a novel object class onto the learned attributes of known classes. As we can observe from Fig. 7, our approach achieves the best performance compared with both the low-level-feature-based and the attribute-based approaches. We achieve the accuracy gain of 4.5% against the second-best approach (Classemes) when using half training samples. We also find that the accuracy seems to reach an upper bound with the increasing of training samples, which reveals the limitation of the shared attributes between different object classes.

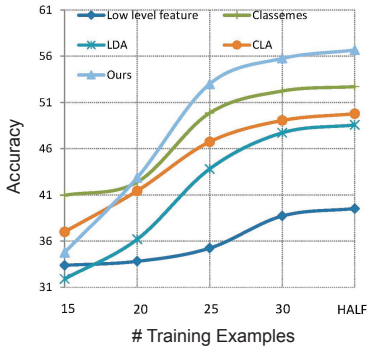

Fig. 7. Multi-class classification on 10 novel object classes.

4.5 Object Categorization on PASCAL VOC 2007

We conducted another object categorization comparison on PASCAL VOC 2007 dataset. We kept 30 attributes for each object class. As there were 20 classes in the dataset, we obtained an attribute space of $30 * 20$ dimensions. The images of training, testing and validation were projected into the attribute space, with each response of an attribute as one attribute feature. Tab. 3 shows the comparison with the four baselines and the best result reported in [31]. Our proposed attribute-based approach improves the best result on 8 out of the 20 classes and boosts the mean average precision (mAP) with 2.8%.

5 Conclusion

In this paper, we have studied the problem of automatic visual attributes learning. To achieve this goal, we proposed an approach with three steps, i.e., prototypes generating, attributes learning and attributes ranking, which can effectively reduce the impact of noises in Web images and ensure that the learned attributes can present a certain property. Extensive experiments showed the good quality of the learned visual attributes and their effectiveness in object categorization. Note that this paper focuses only on visual attribute learning rather than assigning semantic meanings to each learned attribute. We will study the semantic association in our future work.

Table 3. Classification result on PASCAL VOC 2007

	aeroplane	bicycle	bird	boat	bottle	bus	car	cat	chair	cow	table
Low level feature	43.1	36.5	42.6	48.7	27.5	40.6	43.9	38.3	42.5	45.1	46.3
BestResult [31]	**77.5**	63.6	56.1	**71.9**	33.1	60.6	**78.0**	58.8	53.5	42.6	54.9
Classemes(600)	70.5	48.5	**66.7**	68.3	44.7	**66.8**	64.4	**68.0**	**65.6**	60.7	71.7
CLA(600)	65.8	49.0	47.8	57.2	40.0	56.3	60.0	62.4	61.2	46.3	52.6
LDA(600)	61.2	62.8	53.6	56.3	37.5	45.3	46.5	61.3	53.9	60.2	64.8
Ours(600)	67.5	**74.6**	64.8	62.4	**44.8**	59.0	50.2	60.0	63.2	**70.9**	**73.4**

	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
Low level feature	39.4	47.5	40.2	35.3	21.4	31.4	41.6	40.3	18.5	38.5
BestResult [31]	45.8	**77.5**	64.0	**85.9**	36.3	44.7	50.6	**79.2**	**53.2**	59.4
Classemes(600)	58.3	59.3	52.2	46.1	25.4	68.1	**67.5**	69.1	35.5	58.9
CLA(600)	47.8	56.2	62.4	44.1	27.9	41.4	45.3	53.9	25.7	50.1
LDA(600)	52.6	64.5	63.6	67.3	34.0	67.7	48.7	68.5	28.3	54.9
Ours(600)	**61.2**	66.0	**69.7**	74.9	**40.0**	**73.4**	59.5	73.7	35.5	**62.2**

Acknowledgement. This work was supported by 863 Program (2014AA015104), and National Natural Science Foundation of China (61273034, and 61332016).

References

1. Osherson, D.N., Stern, J., Wilkie, O., Stob, M., E.Smith., E.: Default probability. In: Cognitive Science. Volume 15. (1991)

2. F.X.Yu, L.L.Cao, R.S.Feris, J.R.Smith, Chang, S.F.: Designing category-level attributes for discriminative visual recognition. In: CVPR. (2013)
3. Kumar, N., Berg, A.C., Belhumeur, P.N., Nayar, S.K.: Attribute and simile classifiers for face verification. In: ICCV. (2009)
4. Siddiquie, B., Feris, R.S., Davis, L.S.: Image ranking and retrieval based on multi-attribute queries. In: CVPR. (2011) 801–808
5. F.X.Yu, Ji, R., Tsai, M.H., Ye, G., Chang, S.F.: Weak attributes for large-scale image retrieval. In: CVPR. (2012) 2949–2956
6. Wang, Y., Mori, G.: A discriminative latent model of object classes and attributes. In: ECCV. (2010)
7. Branson, S., Wah, C., Schroff, F., Babenko, B., Welinder, P., Perona, P., Belongie, S.: Visual recognition with humans in the loop. In: ECCV. (2010) 438–451
8. Wang, G., Forsyth, D.A.: Joint learning of visual attributes, object classes and visual saliency. In: ICCV. (2009) 537–544
9. Parikh, D., Grauman, K.: Relative attributes. In: ICCV. (2011) 503–510
10. Xu, Z., Wang, X.J., Chen, C.W.: Mining visualness. In: ICME. (2013) 1–6
11. Wang, X.J., Zhang, L., Ma, W.Y.: Duplicate-search-based image annotation using web-scale data. Proceedings of the IEEE **100** (2012) 2705–2721
12. Zoran, D., Weiss, Y.: Natural images, gaussian mixtures and dead leaves. In: NIPS. (2012) 1745–1753
13. Lampert, C.H., Nickisch, H., Harmeling, S.: Learning to detect unseen object classes by between-class attribute transfer. In: CVPR. (2009) 951–958
14. Farhadi, A., Endres, I., Hoiem, D., Forsyth, D.: Describing objects by their attributes. In: CVPR. (2009)
15. Berg, T.L., Berg, E.C., Shih, J.: Automatic attribute discovery and characterization from noisy web data. In: ECCV. (2010)
16. Li-Jia Li, Hao Su, E.P.X., Fei-Fei, L.: Object bank: A high-level image representation for scene classification and semantic feature sparsification. In: NIPS. (2010)
17. Torresani, L., Szummer, M., Fitzgibbon, A.: Efficient object category recognition using classemes. In: ECCV. (2010) 776–789
18. Ferrari, V., Zisserman, A.: Learning visual attributes. In: NIPS. (2007)
19. Yang, Y., Shah, M.: Complex events detection using data-driven concepts. In: ECCV. (2012) 722–735
20. Liu, J., Kuipers, B., Savarese, S.: Recognizing human actions by attributes. In: CVPR. (2011) 3337–3344
21. Harel, J., Koch, C., Perona, P.: Graph-based visual saliency. In: NIPS. (2006) 545–552
22. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society: Series B **39** (1977) 1–38
23. Yang, Y., Shen, H.T., Ma, Z., Huang, Z., Zhou, X.: $l_{2,1}$-norm regularized discriminative feature selection for unsupervised learning. In: IJCAI. (2011) 1589–1594
24. Nie, F., Huang, H., Cai, X., Ding, C.H.Q.: Efficient and robust feature selection via joint ;2, 1-norms minimization. In: NIPS. (2010) 1813–1821
25. Luxburg, U.: A tutorial on spectral clustering. Statistics and Computing **17** (2007) 395–416
26. Golub, G.H., van der Vorst, H.A.: Eigenvalue computation in the 20th century. JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS **123** (2000) 35–65
27. Lazebnik, S., Schmid, C., Ponce, J.: A discriminative framework for texture and object recognition using local image features. In: Toward Category-Level Object Recognition. Volume 4170., Springer (2006) 423–442
28. Bosch, A., Zisserman, A., Munoz, X.: Representing shape with a spatial pyramid kernel. In: CIVR. (2007) 401–408

29. Shechtman, E., Irani, M.: Matching local self-similarities across images and videos. In: CVPR. (2007)
30. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. **3** (2003) 993–1022
31. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. (http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html)