Image and Vision Computing 78 (2018) 1–13

Contents lists available at ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

Attribute annotation on large-scale image database by active knowledge transfer☆

Huajie Jiang[a,b,c,d], Ruiping Wang[b,d,*], Yan Li[b,d], Haomiao Liu[b,d], Shiguang Shan[b,d], Xilin Chen[b,d]

[a]Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, Shanghai 200050, China
[b]Key Laboratory of Intelligent Information Processing of Chinese Academy of Sciences, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China
[c]School of Information Science and Technology, ShanghaiTech University, Shanghai 201210, China
[d]University of Chinese Academy of Sciences, Beijing 100049, China

ARTICLE INFO

Article history:
Received 5 April 2017
Received in revised form 7 May 2018
Accepted 29 June 2018
Available online 20 July 2018

Keywords:
Attribute
Annotation
Relationship
Active learning
Transfer learning

ABSTRACT

Attributes are widely used in different vision tasks. However, existing attribute resources are quite limited and most of them are not in large scale. Current attribute annotation process is generally done by human, which is expensive and time-consuming. In this paper, we propose a novel framework to perform effective attribute annotations. Based on the common knowledge that attributes can be shared among different classes, we leverage the benefits of transfer learning and active learning together to transfer knowledge from some existing small attribute databases to large-scale target databases. In order to learn more robust attribute models, attribute relationships are incorporated to assist the learning process. Using the proposed framework, we conduct extensive experiments on two large-scale image databases, i.e. ImageNet and SUN Attribute, where high quality automatic attribute annotations are obtained.

1. Introduction

Traditional machine learning methods for object recognition require large numbers of training samples to optimize the parameters of an image classifier. There have been continued efforts in collecting large image corpora with a broader coverage of object categories [1], thereby enabling image classification with many classes. While annotating more images of specific categories will make contributions to more accurate classifiers, creating high quality labeled images in large scales is challenging, expensive and time-consuming. Moreover, as images of new categories appear, the annotations should be revised and the classifiers should be re-trained, which wastes much time. Therefore, general models are essential to tackle such problems.

Attributes, which can be shared among different classes, are enjoying increasing popularity recently. It is common knowledge that humans learn new objects from their characteristics which can be described by attributes. Recent research explores a variety of applications for attributes, including object recognition [2-4], image retrieval [5-7], scene understanding [8] and action recognition [9,10]. While attributes play an important role in various vision tasks, image databases with attribute annotations are very scarce and most existing databases are not in large scale. It is thus essential to annotate attributes for more images. However, current attribute annotation is usually done by human labor, which is a heavy burden when the database is in large scale.

In this paper, we focus on the problem of how to perform effective attribute annotation. The motivations for our approach are in three aspects. First, attributes can be shared among different categories [2,3], thus some existing attribute databases can be leveraged to make knowledge transfer, which will save much human labor for annotation. Owing to the fact that the source and target databases may contain totally different categories and the attributes are domain specific, domain shift problem will be incurred [11], so transfer learning methods are needed to tackle such problems. Second, not all samples in the target database are equally important to create discriminative attribute classifiers. In order to save the most of human labor, active learning approaches can be utilized to select the most informative ones to annotate. Third, it is well known that attribute relationships are helpful for the attribute prediction task. For example,

☆ This paper has been recommended for acceptance by Jakob Verbeek.

* Corresponding author at: Key Laboratory of Intelligent Information Processing of Chinese Academy of Sciences, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China.
E-mail address: wangruiping@ict.ac.cn (R. Wang).

https://doi.org/10.1016/j.imavis.2018.06.012
0262-8856/

open area and *closed area* cannot exist in one scene simultaneously. In order to utilize such relationships to improve the attribute models, we incorporate the attribute relationships into the attribute learning process.

In order to solve the problems in the large-scale attribute annotation task, we propose a framework to learn all attributes simultaneously. The general framework of our approach contains three parts, as is shown in Fig. 1. Transfer learning is utilized to borrow prior knowledge from existing source datasets. Active learning is used to annotate the most informative target samples to update the attribute models borrowed from source datasets, aiming for reducing the number of samples to annotate. Moreover, attribute relationships are incorporated to boost the learning process. By unifying transfer learning and active learning in our framework, the workload of human annotation task on the target database can be reduced by a large margin.

The main contributions of this paper lie in three aspects. First, the benefits of transfer learning and active learning are combined aiming to reduce the human labor for the attribute annotation task, where transfer learning borrows knowledge from existing sources and active learning selects the most informative samples to annotate. Both of them reduce the workload of annotation. Second, we incorporate the attribute correlations into the attribute learning process and thus more accurate attribute models can be obtained. Third, we explore the relationships of class-attribute, class-class and attribute-attribute by statistical approaches using the large numbers of labeled samples.

2. Related work

In this section, we give a brief overview of previous works which are closely related to our work. In Section 2.1, we review the related works on attributes. In Sections 2.2 and 2.3, we will introduce some relevant transfer learning methods and active learning methods respectively.

2.1. Attributes

Attributes are general descriptions of images and have drawn much attention in different computer vision tasks such as image classification [2,3,8,9], image retrieval [5,12] and image captioning [13,14]. As mid-level representations, attributes are widely studied in the past few years [15-23]. Traditional approaches usually define attributes beforehand by human, which needs expert knowledge to classify different categories. Recently, [24] proposes an approach to automatically discover discriminative attributes from large text corpus and obtain the class-attribute associations. With the popularity of deep learning in recent years, some works applied deep models to attribute detection [25-29]. All these works need a large number of attribute annotations on images to learn good attribute models. However, the scale of existing attribute databases is limited and almost all existing attribute annotations are performed by human labor which is a huge engineering job. Lampert et al. [2] create an animal database for object recognition which contains 50 classes of animals with a total of 30,475 images. There are 85 attributes annotated for the database, but the attribute annotation is based on classes. It is well known that there are large variations within each category and the class-based annotation could not cover the variations of each individual sample. In order to make more efficient scene recognition, Patterson and Hays [8] use ATM workers to annotate 102 attributes for more than 700 scene categories, with a total number of 14,000 images. Based on the novel idea that attributes can build up relationships among different objects, Russakovsky and Fei-Fei [30] annotate attributes on ImageNet [1], where 25 attributes for about 400 classes with 25 images per class were manually annotated. The annotation task will cost much time and money with the increasing number of attributes and images. Recently [31] proposes to use the class-attribute priors to reduce the attribute annotation task, where a few labeled samples are needed beforehand. While attributes are important in different computer vision tasks, there are few works focus on the basic attribute annotation task and the whole process is done by human. In order to make more efficient attribute annotations, we propose the problem of how to utilize existing attribute databases to annotate attributes for other large-scale image databases.

2.2. Transfer learning

Transferring knowledge among different object classes is an important topic due to its potential to enable efficient learning of object models from a small number of training examples [32]. It provides the basis for scalability to a large number of classes. The basic idea lying in this method is to leverage previously learned object categories when training a new category in which few labeled images are available [33]. Other work trains new category models with no labeled samples in the beginning [34], where class relationships are

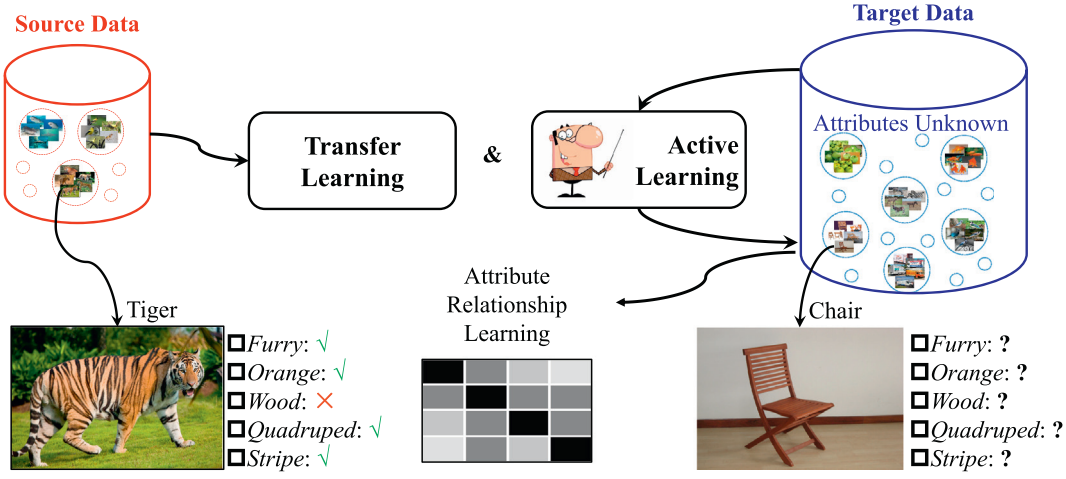

Fig. 1. The general objective of our work. The red circles in the source database represent classes with attribute annotations and the blue ones in the target database represent classes without attribute annotations. Transfer learning brings knowledge of attributes from the source database to the target one and active learning is used to iteratively select informative samples for human annotation and then feed these online labeled samples to update attribute models. Moreover, attribute relationships are taken into consideration to improve the attribute prediction results. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

utilized to form zero-shot new category models from existing categories. In contrast, we directly utilize the transferring property of attributes. For the attributes may be different between new categories and existing categories, domain adaptation methods should be utilized to deal with such gaps [35-37]. While recent researches have utilized the adaption method to tackle the real problems in attributes [27,38-40], most of the previous works consider attributes within the training classes where the domain gap is small. In this paper, we study more general settings where the source and target database may have totally different classes. Our framework has some difference with traditional transfer learning and domain adaptation methods. Traditional transfer learning deals with different source and target classes and domain adaptation deals with the same source and target class but in different domains. Our approach deals with different source and target classes, so it relates to transfer learning approach. However, attributes can be shared among different classes and the attribute classifiers for the source and target datasets are the same. From the view of attributes, it also relates to the domain adaptation approach.

2.3. Active learning

Active learning deals with the problem of finding the most crucial data in a set of unlabeled examples in order to get the maximum information gain [41]. It has yielded a variety of heuristics based on the variant of prediction [42], version space of SVMs [43], disagreement on classifiers [44] and expected informativeness [45]. Recently active learning approaches have also been applied to attributes to tackle some practical problems to reduce human labor [6,7,46-48]. However, most of the existing method deals with each attribute independently, which ignores the attribute correlations. It is well known that the relationships among attributes can improve the attribute prediction task [15,18-20]. In order to achieve more accurate attribute models, we learn multiple attributes simultaneously, where efficient classifier updating method is designed to perform our task.

2.4. Difference with similar works

The most related works to ours are [30] and [31]. [30] annotates 25 attributes for about 10,000 images in ImageNet by human and uses traditional supervised learning approach to learn attribute classifiers. It focuses on the attribute learning process and all the attributes are annotated by human. In contrast, we focus on attribute annotation itself and combine active learning and transfer learning to do incremental attribute learning, where only small numbers of samples need to be annotated by human. [31] is mainly focusing on human labeling. It aims to find discriminative attributes to classify different objects where the class-attribute relationship is utilized as a prior. The statistical results are used as a guideline for human annotation and no learning process is performed. In contrast, we focus on the transferring property of attributes and use the incrementally updated attribute models to guide human annotation.

3. Approach

We tackle the problem of utilizing some existing attribute databases to annotate attributes for a target large-scale database, where transfer learning and active learning approaches are unified to perform such task. The technical process of our approach is shown in Fig. 2. The target database for attribute annotation is divided into two parts: one is labeled offline by human, which serves as **validation set** to evaluate the performance of attribute models, and the other forms the **active set** for online attribute transfer. We first train initial attribute models (*i.e.* classifiers) from the existing small labeled source database. These are the source models, in transfer learning term. Then we evaluate the performance of such models on the validation set. If they are good enough (performance does not improve or reaches a threshold defined beforehand), the learning process will be stopped and the current models are used to predict the attributes of the unlabeled images in the active set, *i.e.* the **AP** set in Fig. 2. Otherwise, a round of active learning process will be triggered to select the most informative samples from the active set for human to annotate, *i.e.* the **AH** set in Fig. 2. Then, they are added to the labeled data pool to update the attribute classifiers, where the attribute relationships are taken into consideration and they will be updated at the same time. The performance of the updated attribute models is again evaluated by the validation set to decide whether to stop or go to a new round of active learning process.

3.1. Problem formulation

As mentioned above, we use active learning approaches to iteratively adapt the attribute models to the target database. As the process of active learning goes on, the number of labeled samples will become larger and larger, which makes it inefficient to retrain attribute models, so an online method is needed to speed up the updating speed.

Notations. Suppose we have N training samples $X = [x_1, x_2, \ldots, x_N] \in \mathbb{R}^{d\times N}$ in the tth iteration and their corresponding attribute labels are denoted as $A = [a_1, \ldots, a_N] \in \mathbb{R}^{T\times N}$, where d and T denote the dimension of the feature vector and the number of attributes respectively. The goal is to learn T attribute classifiers, which are represented by $W = [w_1, w_2, \ldots, w_T] \in \mathbb{R}^{d\times T}$. We denote $W_{pre} \in \mathbb{R}^{d\times T}$ as the attribute models of previous round.

Before the active learning process, some prior knowledge can be obtained from the source databases. Specifically, we train initial attribute models by the labeled images in the source database and it formulates the initial W_1. Then such attribute knowledge can be transferred to the target database, where informative target samples are selected to update the existing attribute models. The objective is to learn attribute classifiers which can not only perform well on the new target samples but also keep the good property of original models. This can be formulated as

$$f(W) = \arg\min_W \frac{1}{N} \left\| A - W^T X \right\|_F^2 + \beta \left\| W - W_{pre} \right\|_F^2 \quad (1)$$

The first term minimizes the classification error to guarantee that the updated attribute models to perform well on the new target samples, where $L2$ loss is utilized to get the closed-form solution, which can be very quick. The second term restricts the updated models to be similar to the original models, which aims to keep the good property of original models. As the number of labeled target samples in each iteration is small, the second term plays an important role to prevent the models from changing by a large margin.

It is common knowledge that attributes often correlated with each other. For example, if a scene picture contains trees, it has a large probability to be taken outdoor. Incorporating such relationships will benefit the attribute learning process. An intuitive way to obtain the attribute relationships is using statistical approaches to make statistics on a large number of labeled samples. However, it is not suitable for our task. Inspired by [49] and [19] where the class relationships are represented by the model correlations, we use the attribute classifiers to model the relationship between the attributes. Thus the attribute relationships can be updated in real-time without having to obtain the attribute labels of all the samples to make statistics. The updated attribute models should keep the attribute relationships so that these models can be updated in a stable way. After incorporating

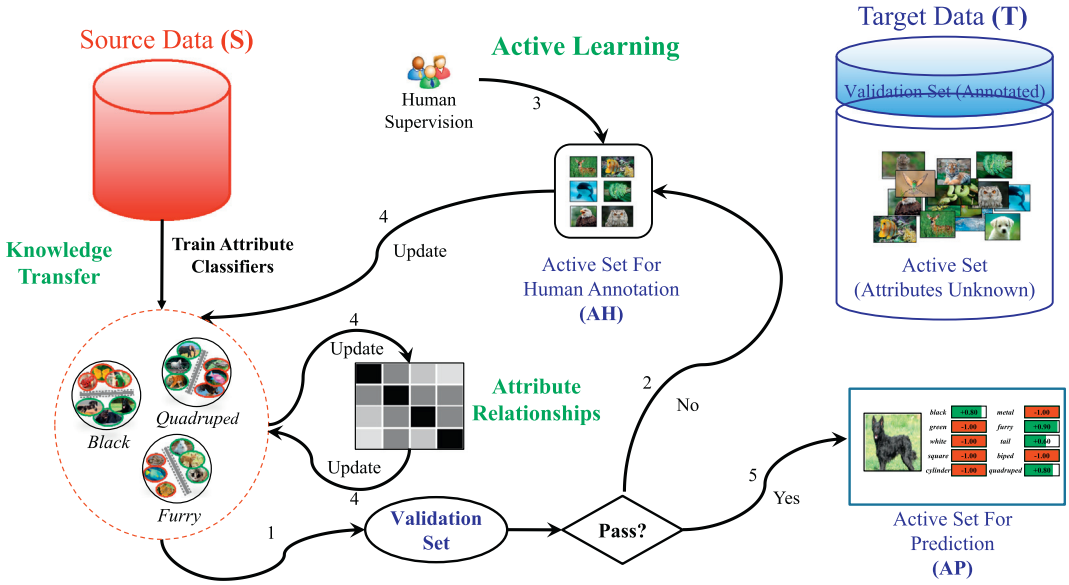

Fig. 2. The technical framework of our approach. The source database is an existing resource which has attribute annotations. The target one is used for annotation. It is divided into two parts: one (**Validation Set**) is annotated beforehand to test the performance of attribute classifiers and the other serves as **Active Set** from which the informative samples are selected to update the attribute classifiers.

the attribute relationships into the active adaption framework, the objective can be formulated as

$$\begin{aligned} f(W,C) = \arg\min_{W,C} & \frac{1}{N}\left\|A - W^T X\right\|_F^2 \\ & + \alpha\, tr(WC^{-1}W^T) + \beta\left\|W - W_{pre}\right\|_F^2 \\ s.t. \quad & C \succeq 0, \quad tr(C) = 1. \end{aligned} \tag{2}$$

where $C \in \mathbb{R}^{T\times T}$ is the column covariance matrix of the weight matrix W, which reflects the attribute-attribute relationships. The first constraint $C \succeq 0$ restricts that C is positive semi-definite because it denotes a task covariance matrix. The second constraint $tr(C) = 1$ serves to restrict the complexity of C.

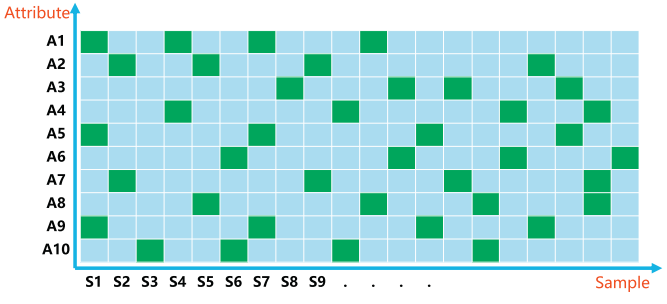

Fig. 3. Procedure for image selection and annotation. For each attribute (each row), the most uncertain samples are selected for human to annotate (green sample-attribute pairs) so each sample is partially labeled (each column). Then the missing attributes (blue ones) are filled with the attributes predicted by previous models. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3.2. Optimization

As the variables in Eq. (2) are jointly convex, thus the problem can be solved by the alternating optimization method.

(1) Fix W, update C.
We can initialize W by W_{pre}, then the sub-problem can be converted to

$$\begin{aligned} C^* = \arg\min_{C} \ & \mathrm{tr}(WC^{-1}W^T) \\ \text{s.t.} \quad & C \succeq 0, \quad \mathrm{tr}(C) = 1. \end{aligned} \tag{3}$$

As is proposed by [49], the optimal C that minimizes Eq. (3) has the following close-form solution:

$$C^* = \frac{(W^TW)^{\frac{1}{2}}}{\mathrm{tr}\left((W^TW)^{\frac{1}{2}}\right)} \tag{4}$$

(2) Fix C, update W.
The sub-problem can be formulated as

$$\begin{aligned} W^* = \arg\min_{W} & \frac{1}{N}\left\|A - W^T X\right\|_F^2 \\ & + \alpha\,\mathrm{tr}\left(WC^{-1}W^T\right) + \beta\left\|W - W_{pre}\right\|_F^2 \end{aligned} \tag{5}$$

Table 1
Attributes annotated for 1000 categories of ImageNet.

Type	Attributes
Color	*Black, blue, brown, gray, green, orange, pink, red, purple, white, yellow, multicolor*
Texture	*Spot, stripe*
Shape	*Square, rounded, cylinder, sharp*
Material	*Metal, wood, furry*
Structure	*Tail, horn, biped, quadruped*

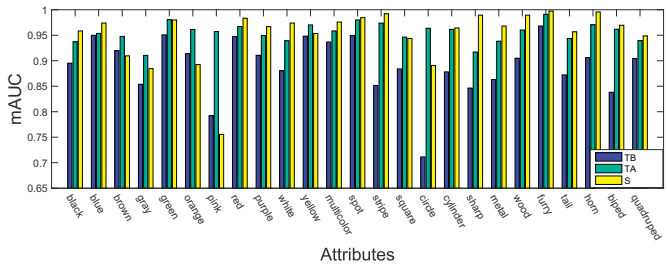

Fig. 4. Performance of the proposed framework on ImageNet. 'TB' shows the performance of initial attribute models on the **target** validation set **before** transfer. 'TA' shows the performance on the **target** validation set **after** attribute transfer. 'S' shows the performance of initial attribute models on the **source** validation dataset.

This is a convex problem which has closed-form solution. We can obtain its gradient as follows:

$$\frac{\partial f(W,C)}{\partial W} = \frac{2}{N}\left(XX^TW - XA^T\right) + \alpha W\left(C^{-1} + \left(C^{-1}\right)^T\right) + 2\beta\left(W - W_{pre}\right) \quad (6)$$

We denote $C^{-1} + (C^{-1})^T$ by R. By forcing $\frac{\partial f(W,C)}{\partial W}$ to be 0, we obtain that

$$\mathrm{vec}(W^*) = \left(I \otimes \left(\frac{2}{N}XX^T\right) + (\alpha R + 2\beta I)^T \otimes I\right)^{-1} * \mathrm{vec}\left(\frac{2}{N}XA^T + 2\beta W_{pre}\right) \quad (7)$$

where $\otimes$ represents Kronecker product and vec represents the vec operator. I represents the identity matrix. It is obvious that when the feature dimension and attribute numbers are large, the computation and memory cost will be extremely huge for Kronecker product operation. For example, the computation cost of Eq. (7) is $O(d^6)$ and the memory cost is $O(d^4)$, where d is the dimension of features. To solve such problem, we propose an algorithm based on Block Coordinate Descent (BCD) principles. In this approach, we introduce a slack variable W_r and force it to be similar to W, then the original problem may be solved by two alternating processes, focusing on a new cost function and the regularization term respectively. That is, we first convert the original problem into

$$g(W,W_r) = \arg\min_{W,W_r} \frac{1}{N}\left\|A - W^TX\right\|_F^2 + \alpha\, tr\left(W_rC^{-1}W_r^T\right) + \beta\left\|W - W_{pre}\right\|_F^2 + \gamma\left\|W - W_r\right\|_F^2 \quad (8)$$

in which the norm $\|W - W_r\|_F^2$ enforces a similar solution of W and W_r. First, we initialize W with W_{pre}. Then we obtain the attribute models by iteratively updating W_r and W in the following two problems:

Optimization of W_r: For fixed W, the optimization of W_r can be solved by

$$W_r^* = \arg\min_{W_r} \alpha\, Tr\left(W_rC^{-1}W_r^T\right) + \gamma\left\|W - W_r\right\|_F^2 \quad (9)$$

and there is a closed-form solution for W_r

$$W_r^* = 2\gamma W\left(\alpha\left(\left(C^{-1}\right)^T + C^{-1}\right) + 2\gamma I\right)^{-1} \quad (10)$$

Optimization of W: For fixed W_r, the optimal attribute models W can be obtained via solving

$$W^* = \arg\min_{W} \frac{1}{N}\left\|A - W^TX\right\|_F^2 + \beta\left\|W - W_{pre}\right\|_F^2 + \gamma\left\|W - W_r\right\|_F^2 \quad (11)$$

This optimization problem is convex and has closed-form solution

$$W^* = \left(\frac{1}{N}XX^T + \beta I + \gamma I\right)^{-1}\left(\frac{1}{N}XA^T + \beta W_{pre} + \gamma W_r\right) \quad (12)$$

By transforming Eqs. (7) to (8), the computation and memory costs can be reduced to $O(d^3)$ and $O(d^2)$ respectively.

3.3. Image selection strategy

In this paper, we consider a pool-based active learning which appears to be the most popular scenario for applied research in active learning. Existing active learning approaches are mainly based on uncertainty sampling and expected loss reduction. Due to the fact that the computation of entropy reduction will cost much time in a large-scale image database, we adopt the more simple uncertainty sampling method. In uncertainty sampling approaches, the most informative samples are considered to locate near the current classifier hyper planes, so using these annotated samples to update the attribute models will gradually improve the models. However, traditional uncertainty sampling methods are mainly designed for single attribute, while the proposed framework deals with all attributes simultaneously. An intuitive approach is to combine the uncertainty of all attributes and select the samples which have largest overall uncertainty and annotate all attributes of these selected samples. However, this approach may select samples not informative for each attribute although they have largest overall uncertainty.

In order to select informative samples for each attribute, we modify the traditional uncertainty sampling method to fit for the task of simultaneously updating all attribute classifiers. Specifically, we select the most informative attribute-sample pairs, as is shown in

Table 2
The performance of initial attribute classifiers on the source and target databases on SUN Attribute. 'S' represents the performance of original models on the **source** database. 'TB' shows the performance of initial attribute models on the **target** validation set **before** transfer. 'TA' shows the performance on the **target** validation set **after** attribute transfer.

Attribute	S	TB	TA	Attribute	S	TB	TA
Rock/stone	0.94	0.87	0.89	*Direct sun/sunny*	0.85	0.81	0.81
Still water	0.94	0.83	0.86	*Natural light*	0.88	0.85	0.86
Warm	0.72	0.68	0.71	*Far-away horizon*	0.96	0.94	0.95
Shrubbery	0.92	0.92	0.92	*Clouds*	0.91	0.88	0.89
Ocean	0.97	0.94	0.95	*Open area*	0.94	0.92	0.93

Fig. 5. Examples of source images and failed target images on 'cylinder' and 'metal' for ImageNet.

Fig. 3. Specifically, for each attribute A_i, we select the most informative samples for human to annotate, as is shown by the green block of Fig. 3 in each row. For simplicity, the distance to the classification hyperplane is exploited as the uncertainty measure and it can be computed by

$$D_i^m = |W_m * x_i - \sigma_m|, \tag{13}$$

where D_i^m denotes the distance of sample x_i to the classification hyperplane for attribute m and σ_m is the classification threshold. Samples with smallest D_i^m are selected for human to annotate. This process is the same as that in traditional uncertainty sampling approaches for single attribute [6,7,46,47]. In this way, we obtain some partially labeled samples, *i.e.* samples with one or few attributes annotated (each column of Fig. 3), and we call these annotations attribute-sample pair annotations. However, in order to update all the attribute models simultaneously, our framework needs to know all the attributes of each sample, so we fill the missing attribute labels (blue blocks in Fig. 3) of the selected samples with the attributes predicted by the current attribute classifiers. Thus we can obtain all the attributes of the selected samples. Then, we use these samples to update the attribute classifiers. There are two benefits for the proposed method. First, the most informative samples for each attribute are obtained and they are annotated by human (green blocks in Fig. 3). These accurate annotations are helpful to improve the current attribute models. Second, we use the attributes predicted by previous models to fill up the missing attribute labels (blue blocks in Fig. 3). These predicted attributes are partial representations of previous models and using these attributes to update the models will keep the relatively good property of previously learned models, which is complementary to the third term of Eq. (2).

4. Experiments

4.1. Experiment settings

We perform experiments on two large-scale image datasets: ImageNet [1] and SUN Attribute Database [8]. ImageNet is a benchmark database for large-scale visual recognition which is organized by the hierarchical structure. We choose the 1000 categories which are used for the classification task of the ImageNet large-scale visual recognition challenge [50] to perform our experiment, where 50 images for each class are selected. To perform the proposed framework, we randomly select 100 categories to form the source image database and the rest 900 categories are used as the target database. Then we divide the target database into two parts, *i.e.*, 10% of the images in each class are selected to serve as validation set and the others form the active set from which the informative samples are selected. We extract 4096-dimension AlexNet features of all images by Caffe [51] with the released model pre-trained on ImageNet, where no additional pre-training is done. It is worth noting that the source database and the target database contain totally different categories, which makes the gap within the attributes larger and be more difficult than traditional transfer learning problem. In our experiment, we set the parameters in Eq. (8) as $\alpha = 0.1, \beta = 10$, and $\gamma = 1$. SUN Attribute Dataset is built based on SUN categorical

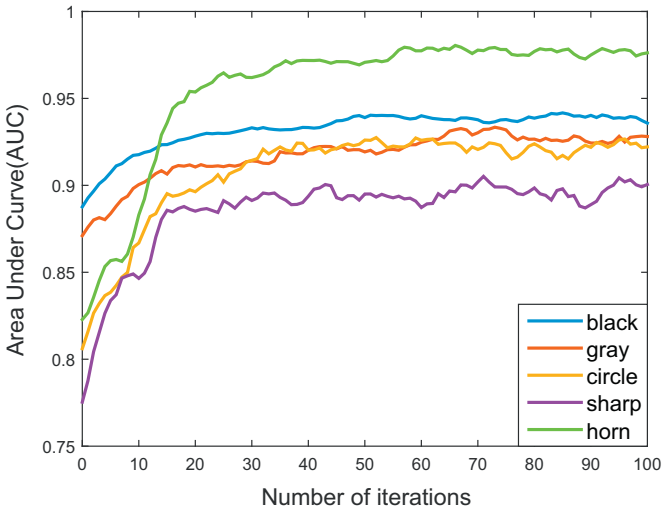

Fig. 6. Improving performance of 5 attribute models in ImageNet during 100 iterations.

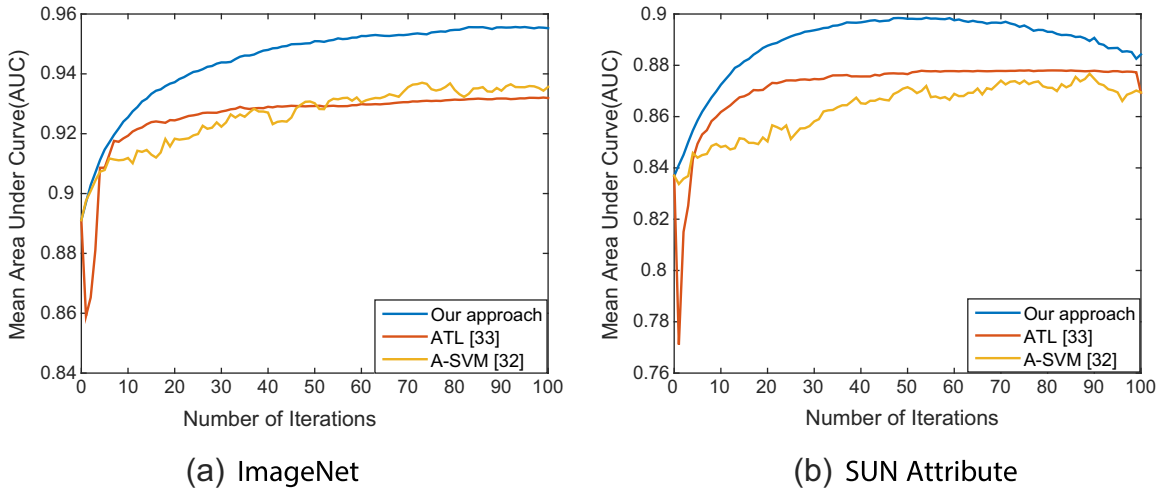

Fig. 7. Comparison results with state-of-art methods.

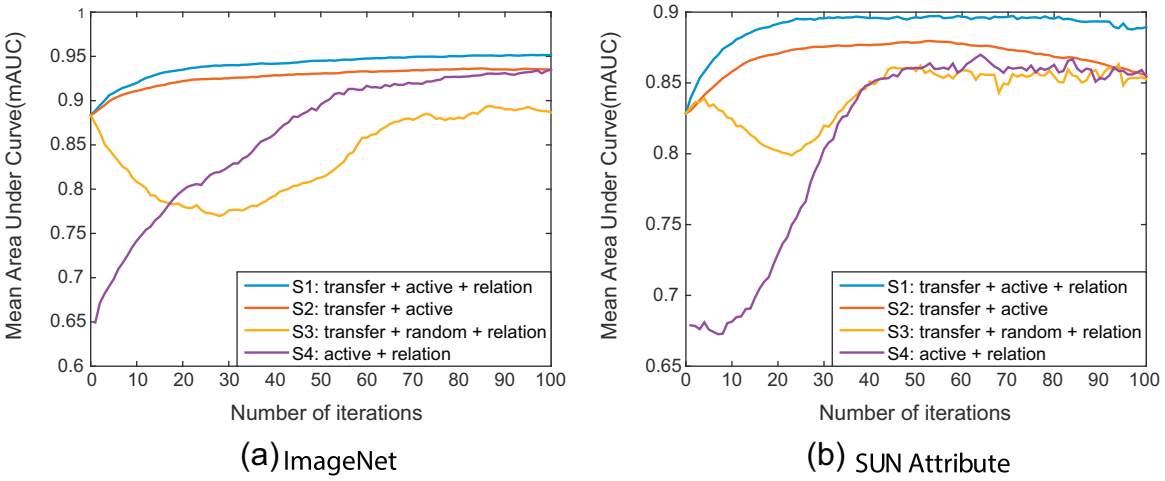

Fig. 8. The performance of different comparing methods by mean Area Under Curve (mAUC): **S1** (transfer + active + relation) is the proposed method; **S2** (transfer + active) removes the relationship constraint (second term) in the proposed framework; **S3** (transfer + random + relation) replaces the active selection by random selection; and **S4** (active + relation) is the proposed framework without transfer (no source models are utilized).

database [52] for high-level scene understanding and fine-grained scene recognition. It spans more than 700 categories and 14,000 images with 102 attributes annotations. In our experiment, 100 categories are picked out as the source database and the others are used as the target database to annotate. In our experiment, we set the parameters in Eq. (8) as $\alpha = 0.1, \beta = 10$, and $\gamma = 10$. We define the stopping mAUC as 0.96, which is a little higher than that on the source validation set. However, it mostly cannot be reached. We also define the maximum iterations to be 100 to control the learning process. In the real applications, the updating process can be stopped when the performance has little changes in successive iterations.

4.2. Attribute annotation

It is well known that large numbers of attributes are needed to define an object for the classification task. Since our purpose is focusing on the attribute annotation task, which is general to all attributes, we define 25 common attributes by reference to [30], which include color, texture, shape, material and structure, to demonstrate the effectiveness of our framework. In order to perform our experiment automatically, we annotate 25 attributes for the images in ImageNet beforehand, with a total number of 50,000 images from the 1000 categories used in the classification task[1]. Table 1 shows the attributes annotated in our work. There are 19 attributes in common with [30]. We removed the attributes which are difficult to annotate or cannot be seen by appearance, such as *rough* and *vegetation*, and replace them with some common visual attributes.

We follow the annotation strategy of [30], where each image is annotated by three people for each attribute. Specifically, we hire 25 college students to annotate the images for three rounds. In each round, we randomly assign an attribute to one student with guarantee that each student annotates different attributes in three rounds. In this way, each image is annotated three times for one attribute by different people. Given an image, annotators are required to

[1] The annotation will be released at http://vipl.ict.ac.cn/database.php.

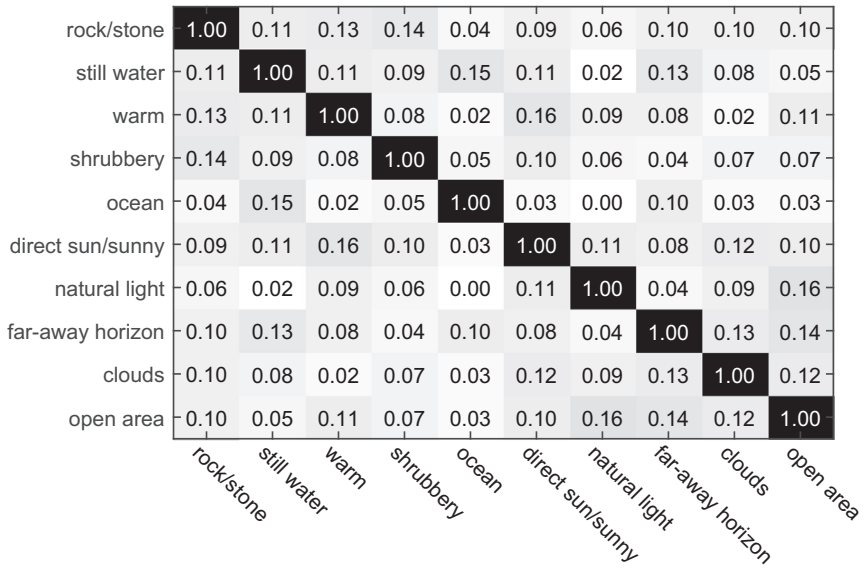

Fig. 9. Attribute relationships learned on SUN Attribute.

decide whether a specific attribute exists or not. The final results are produced by a voting strategy.

In order to check the annotation quality, we select 50 positive images and 50 negative images for each attribute which are used to check the annotation results. The mean accuracy of all attributes is 0.92 and the accuracy of 19 attributes exceeds 0.9. The attributes difficult for annotation include *brown*, *gray*, *orange*, *purple*, *spot*, and *sharp*. Some of these attributes may be difficult to annotate due to the outer factors. For example, it is difficult to decide whether an image contains *gray* or not in low light conditions and the variation of *sharp* is very large.

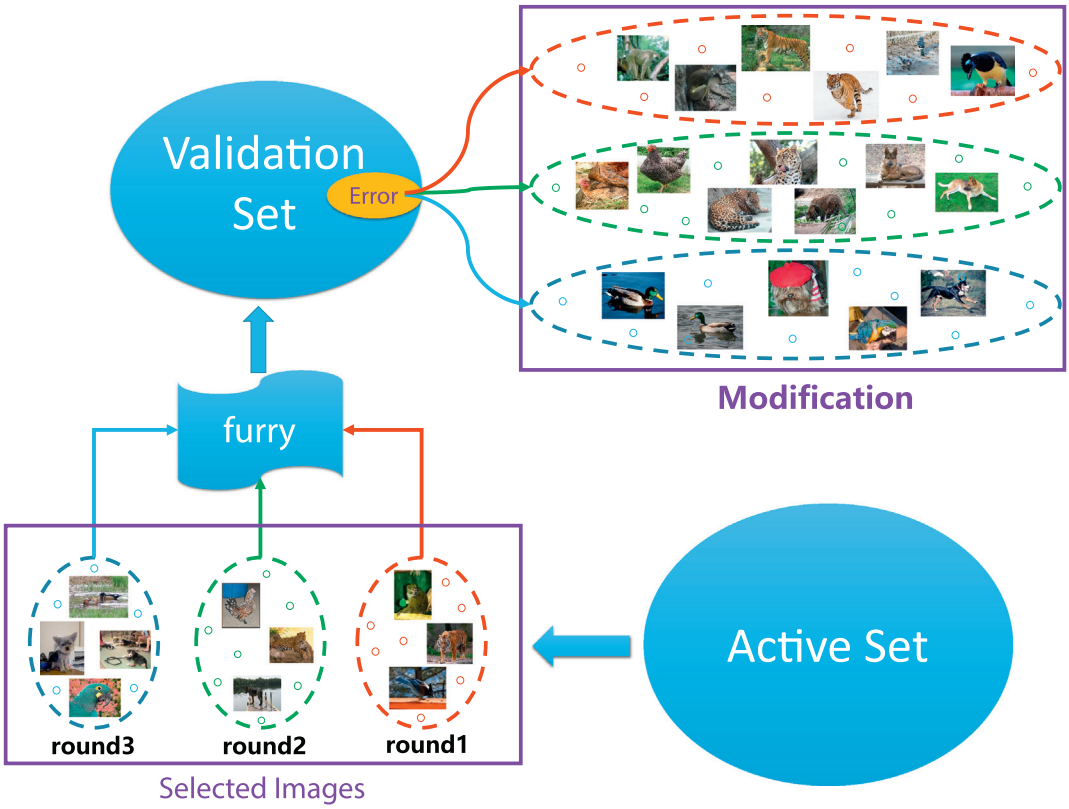

Fig. 10. The learning process of *furry* in the active learning rounds on ImageNet. 'Error' in the validation set denotes images which are wrongly predicted by the original attribute model. The images in the modification frame are the images correctly predicted after the model updating by the selected samples in the active set.

Fig. 11. Attribute retrieval results on **AP** for ImageNet. Images with green border (left 3 columns) are positive images with highest confidence and images in red border (right 3 columns) are negative images with highest confidence. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

4.3. Transfer property of attributes

Although it is well known that attributes can be shared among categories, few works give quantitative analysis of such property. In this part, we explore the transfer property of attributes. Specifically, we train initial attribute models on the source database and directly measure their performance on the target databases, where leave-one-out scheme is used to test the performance on the source database and the target **validation set** is used to measure the performance on the target database. For the source and target databases contain totally different categories, the difference in the performance is a good measure for the transfer property of attributes. Considering the imbalance of positive and negative samples, we use the Area Under Curve (AUC) to report the performance. In the initial process, we leave out a source validation set to test the original attribute models, where the performance will indicate the difficulty of learning each attribute. Fig. 4 and Table 2 show the performance on ImageNet and SUN Attribute. We can figure out that after active knowledge transfer, the performance improves a lot and nearly reaches the results on the source validation set. For ImageNet, the mean AUC before attribute transfer is 0.89 and after attribute transfer, the mean AUC improves to 0.95. It can be seen that some attribute classifiers trained on the source database can also have relatively good performance on the target database, such as *furry*, which indicates that attributes can be shared among different categories. However, the performance of some attribute models degrades a lot for the target database, such as *cylinder* and *metal*. This may be caused by the large variations between the two databases. Some examples of source images and failed target images are shown in Fig. 5. In such cases, directly using the original attribute models obtained from the source images to predict the attributes of the target images is not a satisfactory choice, where domain shift problems need to be tackled to adapt to the target database.

4.4. Effectiveness of the proposed framework

To deal with the gap between the source and target database, we perform experiments on ImageNet. Specifically, we use active learning method to select the most informative target images for annotation and use these samples to update the original attribute models trained on the source images. In our experiment, 50 images for each attribute are selected per iteration. The performance of 5 attributes is shown in Fig. 6. It can be inferred that the attribute models are improving as the learning process going on and after about 30 iterations, the improvement is small and the learning process can be stopped. To evaluate the theoretical upper bound of performance, we use all the active set (a total of 40,500 images) to train the attribute models and the mAUC is 0.97. The performance of source attribute model is 0.89 and we achieve 0.95. However, we only perform 50 iterations and labeled $50 * 50 * 25$ attributes, while the upper-bound models need to annotate $40,500 * 25$ attributes.

To demonstrate the effectiveness of the proposed framework, we compare our method with two approaches: active transfer learning (ATL) [34] and adaptive SVM (A-SVM) [33]. [34] proposes a method for zero-shot learning by reusing the past datasets. This work utilizes class relationships to form the new category classifiers in the initial process, while we perform attribute transfer directly. To make it suitable for our task, we modified the process of initializing the models, where attribute models learned from the source dataset are directly used. Then we perform the attribute prediction task using the models proposed by [34]. [33] uses traditional adaptive SVM (A-SVM) approach to do the transfer learning task. In order to make it comparable to our approach, we use traditional distance-based uncertainty sampling approach to select images for each attribute and update

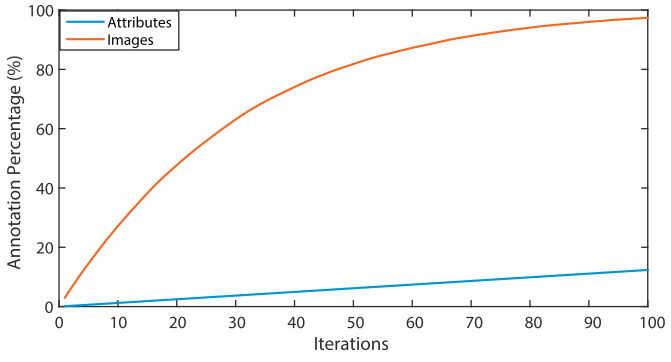

Fig. 12. The percentage of annotated attributes and images in ImageNet with the iterations going on.

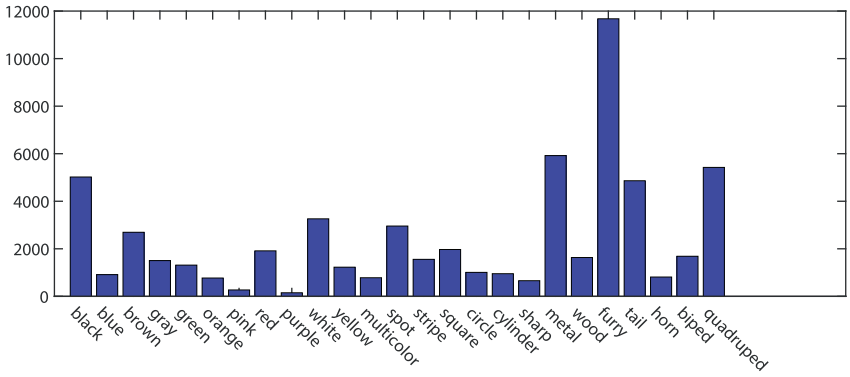

Fig. 13. The number of positive images for each attribute on ImageNet.

each attribute model using A-SVM. The comparison results are shown in Fig. 7. We can figure out that our model achieves higher performance and our approach is more stable. The performance of [34] drops in the first iteration. This may be caused by the learning procedure. It learns target classifier individually and adopts a weighted summarization of source classifier and target classifier to form the final classifier. The target classifiers learned by small numbers of samples in the first iteration is not good enough, so the performance may drop. While we use global optimization approach to learn the final classifier, which is more stable. We can figure out that the performance of our approach on SUN Attribute begins to drop after 60 iterations. We think that the models have reached their limits. To demonstrate such suppose, we use all the active set samples to train the attribute models and the mAUC is 0.909. Our model reaches about 0.9 after 50 iterations, the performance may drop if we continue to update the models using uncertain samples. Moreover, our approach takes about 15 min for 25 attributes on ImageNet in 100 iterations and [34] takes about 130 min, which demonstrates that our approach is very fast.

To evaluate the effectiveness of each part, we compare the performance of four different approaches in Fig. 8. S1 is the proposed method; S2 removes the relationship constraint (second term) in the proposed framework; S3 replaces the active selection by random selection; S4 is the proposed framework without transfer (no source models are utilized).

4.4.1. Effectiveness of active learning

By comparing the performance of **S1** and **S3**, we can figure out that active learning plays an important role in the improvement of attribute models. Using the most informative samples to modify current attribute models will gradually adapt the attribute models to the target database. The performance decreases at the beginning of the random selection, this may be caused by the less informative samples selected, which pull the classification hyperplane away from the original models by a large margin.

4.4.2. Effectiveness of transfer learning

By comparing the performance of **S1** and **S4**, we can see that with the help of transfer learning, the attribute models have a good start in the active learning process and converge with fewer labeled samples. It can be inferred that with the help of transfer learning fewer labeled images are needed to reach a suitable performance.

4.4.3. Effectiveness of the attribute relationship

By comparing the performance of **S1** and **S2**, we can figure out that modeling the relationships between attributes in the learning process will make some benefits. In order to have a concrete impression of what these relationships are, we visualize the relationships between 10 attributes in Fig. 9. Specifically, we normalize C by

$$C_{ij} = \frac{C_{ij}}{(C_{ii} * C_{jj})^{\frac{1}{2}}} \tag{14}$$

and choose ten attributes to show their relationships, which are reflected by corresponding values in C. For C is not derived from statistics but the correlations of attribute model, so its values only show the relative strength of the relationships. From this figure, we can discover relative strong relationship between *natural light* and *open area*.

4.4.4. Active learning process

In order to explore the working manner of active learning, we select some representative images in ImageNet to show the improving process of attribute models in Fig. 10. In each round, the most informative images in the active set are selected to be annotated and then be used to update the attribute models. It is interesting that similar images which have been wrongly predicted in previous rounds are modified. This demonstrates that it is possible to make knowledge transfer by active learning techniques.

4.4.5. Attribute retrieval

To see what the attribute models have learned, we use these attribute models to perform image retrieval on the **AP** set in

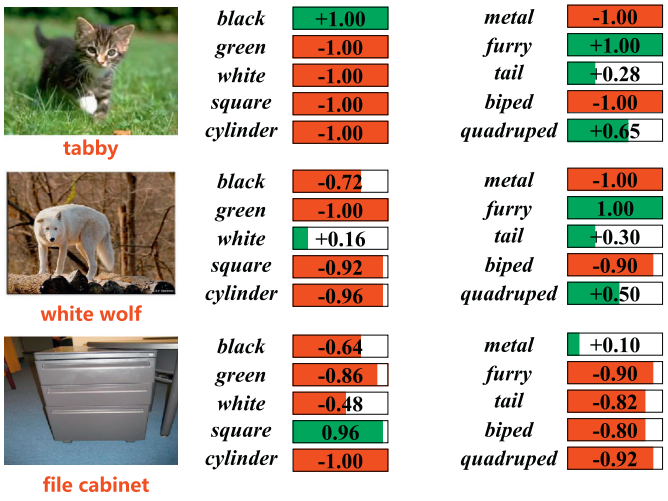

Fig. 14. Attribute descriptions of 3 classes on ImageNet, which are learned automatically by statistics.

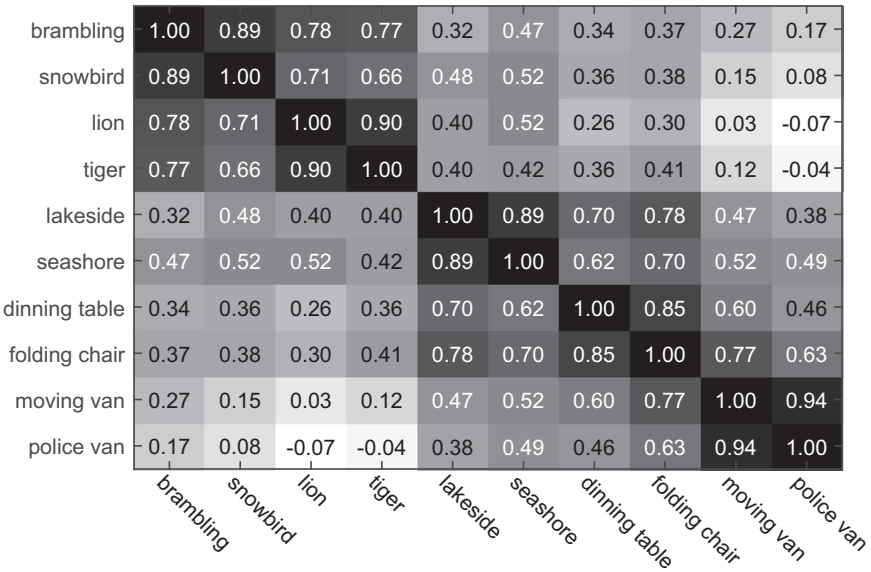

Fig. 15. Class relations on ImageNet, which are learned by their attribute descriptions.

ImageNet. The results of five attributes are shown in Fig. 11. It is obvious that the attribute models have definitely learned their corresponding semantic meanings. We have also made quantitative analysis of our models on the target validation set. The attribute retrieval mAP (mean average precision) on all the images is 0.90.

4.4.6. Statistical results

Fig. 12 shows the percentage of labeled attributes and images with the iterations going on. It can be figured out that the number of annotated attributes is growing stably and the percentage of annotated images reaches nearly 1 in 100 iterations, which indicates the diversity of annotated samples. We also make statistics about the final annotation results by combining the human labeled attributes (**AH**) and the attributes predicted by the final models (**AP**). Fig. 13 shows the number of positive images for each attribute. We can figure out that some attributes are rare in the 1000 classes, such as *pink* and *purple*. The average number of attributes per image is 1.50.

4.5. Further analysis

Attributes are helpful for describing objects. Meanwhile, they can build up relations of different classes. To make a direct impression on whether the learned attribute models are good or not, we make a qualitative analysis of the learned attribute models, where knowledge discovery process is performed based on statistics.

The first kind of knowledge we want to explore is the associations between classes and attributes. Using these attribute prediction results, we can learn these associations automatically by statistics. Specifically, we make statistics for each class whether a specific attribute exists or not based on the automatic annotation results. Fig. 14 shows some class descriptions using ten attributes, where the scores are normalized to $[-1, 1]$. Intuitively they are in accordance with human knowledge, which reflects the attribute models learned by our framework are relatively good.

The second knowledge we want to learn is the relationships between classes. It is obvious that different classes can be connected by the same attributes. Using the descriptions learned above,

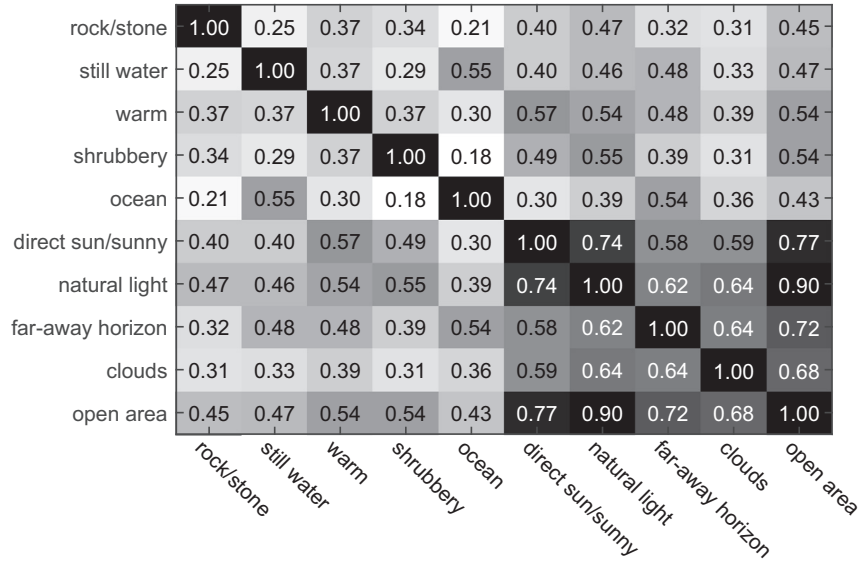

Fig. 16. Attribute relationships on SUN Attribute, which are learned by the attribute prediction results based on statistics.

we can automatically decide the similarity between two classes by their cosine distance. Fig. 15 shows the similarities between classes learned by the class-attribute association. It can be figured out that similar classes are grouped together and dissimilar classes have low correlations. For example, in ImageNet, *brambling* is most similar to *snowbird* and *tiger* is most similar to *lion*. The class relationships shown in Fig. 15 automatically groups these categories into two large parts, i.e. animal or non-animal. Moreover, within each large part, several small parts are formed. We can use these attribute descriptions to perform automatic clustering task.

The third knowledge we want to learn is the relationships between attributes. Using the attribute prediction results, we can learn the relationships between attributes based on the statistical method. Specifically, we compute the conditional probability of one attribute given another attribute and regard it as the relationships between these two attributes. We average the symmetric terms to form the final attribute relationships. Fig. 16 shows the attribute relationships learned by statistics. It can be seen that the relative strengths of these relationships are mostly in accordance with Fig. 9 that learned in our framework. For example, the relationship between *natural light* and *open area* is strong.

4.6. Evaluations on time saving

In this part, we make statistics about the human labeling time. It takes about 2 s to annotate one attribute for one image in the annotation task. In our experiment, there are 40,000 images in the target database. The proposed framework needs to annotate 4000 images as the validation set. It can be seen from Fig. 8 that the proposed framework converges in about 50 iterations, with 2500 images for each attribute annotated. In total, we will reduce the annotation task to about 1/6 of its original task. After 50 iterations of active learning, the final mAUC obtained by our framework is 0.95 and the final accuracy is 0.97.

5. Conclusion

This paper proposes a framework for large-scale attribute annotation, which leverages existing attribute databases to reduce the human labor. The proposed framework combines the benefits of transfer learning and active learning to reduce the workload in the annotation task. Meanwhile, the attribute relationships are also utilized to assist the learning process. In order to speed up the training process, an online updating approach is designed. Extensive experiments show the effectiveness of each part of the proposed framework. Furthermore, we annotate a large-scale attribute database based on ImageNet, which will be released to the public in the near future.

Conflict of interest statement

There is no conflict of interest.

Acknowledgments

This work is partially supported by 973 Program under contract No. 2015CB351802, Natural Science Foundation of China under contract Nos. 61390511 and 61772500, Frontier Science Key Research Project CAS No. QYZDJ-SSW-JSC009, and Youth Innovation Promotion Association CAS No. 2015085.

References

[1] J. Deng, W. Dong, R. Socher, L.J. Li, K. Li, F.F. Li, ImageNet: a large-scale hierarchical image database, Computer Vision and Pattern Recognition, 2009. pp. 248–255.
[2] C.H. Lampert, H. Nickisch, S. Harmeling, Learning to detect unseen object classes by between-class attribute transfer, Computer Vision and Pattern Recognition, 2009. pp. 951–958.
[3] A. Farhadi, I. Endres, D. Hoiem, D. Forsyth, Describing objects by their attributes, Computer Vision and Pattern Recognition, 2009. pp. 1778–1785.
[4] C. Huang, C. Change Loy, X. Tang, Unsupervised learning of discriminative attributes and visual representations, Computer Vision and Pattern Recognition, 2016. pp. 5175–5184.
[5] B. Siddiquie, R.S. Feris, L.S. Davis, Image ranking and retrieval based on multi-attribute queries, Computer Vision and Pattern Recognition, 2011. pp. 801–808.
[6] A. Kovashka, D. Parikh, K. Grauman, Whittlesearch: image search with relative attribute feedback, Computer Vision and Pattern Recognition, 2012. pp. 2973–2980.
[7] A. Kovashka, K. Grauman, Attribute pivots for guiding relevance feedback in image search, International Conference on Computer Vision, 2013. pp. 297–304.
[8] G. Patterson, J. Hays, Sun attribute database: discovering, annotating, and recognizing scene attributes, Computer Vision and Pattern Recognition, 2012. pp. 2751–2758.
[9] J. Liu, B. Kuipers, S. Savarese, Recognizing human actions by attributes, Computer Vision and Pattern Recognition, 2011. pp. 3337–3344.
[10] A. Diba, A.M. Pazandeh, H. Pirsiavash, L.V. Gool, DeepCAMP: deep convolutional action & attribute mid-level patterns, Computer Vision and Pattern Recognition, 2016. pp. 3557–3565.
[11] Y. Fu, T.M. Hospedales, T. Xiang, Z. Fu, S. Gong, Transductive multi-view embedding for zero-shot recognition and annotation, European Conference on Computer Vision, 2014. pp. 584–599.
[12] M. Douze, A. Ramisa, C. Schmid, Combining attributes and fisher vectors for efficient image retrieval, Computer Vision and Pattern Recognition, 2011. pp. 745–752.
[13] G. Kulkarni, V. Premraj, V. Ordonez, S. Dhar, S. Li, Y. Choi, A.C. Berg, T.L. Berg, Babytalk: understanding and generating simple image descriptions, IEEE Trans. Pattern Anal. Mach. Intell. 35 (12) (2013) 2891.
[14] V. Ordonez, G. Kulkarni, T.L. Berg, Im2Text: describing images using 1 million captioned photographs, 2011. pp. 1143–1151.
[15] Y. Wang, G. Mori, A discriminative latent model of object classes and attributes, European Conference on Computer Vision, 2010. pp. 155–168.
[16] D. Mahajan, S. Sellamanickam, V. Nair, A joint learning framework for attribute models and object descriptions, International Conference on Computer Vision, 2011. pp. 1227–1234.
[17] Y. Fu, T.M. Hospedales, T. Xiang, S. Gong, Attribute learning for understanding unstructured social activity, European Conference on Computer Vision, 2012. pp. 530–543.
[18] L. Chen, Q. Zhang, B. Li, Predicting multiple attributes via relative multi-task learning, Computer Vision and Pattern Recognition, 2014. pp. 1027–1034.
[19] M. Liu, D. Zhang, S. Chen, Attribute relation learning for zero-shot classification, Neurocomputing 139 (2014) 34–46.
[20] D. Jayaraman, F. Sha, K. Grauman, Decorrelating semantic visual attributes by resisting the urge to share, Computer Vision and Pattern Recognition, 2014. pp. 1629–1636.
[21] W. Kusakunniran, Attribute-based learning for gait recognition using spatio-temporal interest points, Image Vis. Comput. 32 (12) (2014) 1117–1126.
[22] S. Huang, M. Elhoseiny, A. Elgammal, D. Yang, Learning hypergraph-regularized attribute predictors, Computer Vision and Pattern Recognition, 2015. pp. 409–417.
[23] P. Samangouei, V.M. Patel, R. Chellappa, Facial attributes for active authentication on mobile devices, Image Vis. Comput. 58 (2017) 181–192.
[24] Z. Al-Halah, R. Stiefelhagen, Automatic discovery, association estimation and learning of semantic attributes for a thousand categories, Computer Vision and Pattern Recognition, 2017. pp. 5112–5121.
[25] P. Luo, X. Wang, X. Tang, A deep sum-product architecture for robust facial attributes analysis, International Conference on Computer Vision, 2013. pp. 2864–2871.
[26] N. Zhang, M. Paluri, M. Ranzato, T. Darrell, L. Bourdev, PANDA: pose aligned networks for deep attribute modeling, Computer Vision and Pattern Recognition, 2014. pp. 1637–1644.
[27] Q. Chen, J. Huang, R. Feris, L.M. Brown, J. Dong, S. Yan, Deep domain adaptation for describing people based on fine-grained clothing attributes, Computer Vision and Pattern Recognition, 2015. pp. 5315–5324.
[28] V. Escorcia, J.C. Niebles, B. Ghanem, On the relationship between visual attributes and convolutional networks, Computer Vision and Pattern Recognition, 2015. pp. 1256–1264.
[29] J. Zhu, S. Liao, Z. Lei, S.Z. Li, Multi-label convolutional neural network based pedestrian attribute classification, Image Vis. Comput. 58 (2017) 224–229.
[30] O. Russakovsky, L. Fei-Fei, Attribute learning in large-scale datasets, European Conference on Trends and Topics in Computer Vision, 2010. pp. 1–14.
[31] G. Patterson, J. Hays, COCO attributes: attributes for people, animals, and objects, European Conference on Computer Vision, 2016. pp. 85–100.
[32] S.J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22 (10) (2010) 1345–1359.
[33] Y. Aytar, A. Zisserman, Tabula rasa: model transfer for object category detection, International Conference on Computer Vision, 2011. pp. 2252–2259.
[34] E. Gavves, T. Mensink, T. Tommasi, C.G.M. Snoek, T. Tuytelaars, Active transfer learning with zero-shot priors: reusing past datasets for future tasks, International Conference on Computer Vision, 2015. pp. 2731–2739.

[35] K. Saenko, B. Kulis, M. Fritz, T. Darrell, Adapting visual category models to new domains, European Conference on Computer Vision, 2010. pp. 213–226.
[36] R. Gopalan, R. Li, R. Chellappa, Domain adaptation for object recognition: an unsupervised approach, International Conference on Computer Vision, 2011. pp. 999–1006.
[37] B. Gong, Y. Shi, F. Sha, K. Grauman, Geodesic flow kernel for unsupervised domain adaptation, Computer Vision and Pattern Recognition, 2012. pp. 2066–2073.
[38] A. Kovashka, K. Grauman, Attribute adaptation for personalized image search, International Conference on Computer Vision, 2013. pp. 3432–3439.
[39] Y. Han, Y. Yang, Z. Ma, H. Shen, N. Sebe, X. Zhou, Image attribute adaptation, Multimedia 16 (4) (2014) 1115–1126.
[40] C. Gan, T. Yang, B. Gong, Learning attributes equals multi-source domain generalization, Computer Vision and Pattern Recognition, 2016. pp. 87–97.
[41] B. Settles, Active Learning Literature Survey, 39 (2). University of Wisconsin–Madison. 2009, 127–131.
[42] A. Kapoor, K. Grauman, R. Urtasun, T. Darrell, Gaussian processes for object categorization, IJCV 88 (2) (2010) 169–188.
[43] S. Tong, D. Koller, Support vector machine active learning with applications to text classification, J. Mach. Learn. Res. 2 (1) (2002) 45–66.
[44] Y. Freund, H.S. Seung, E. Shamir, N. Tishby, Selective sampling using the query by committee algorithm, Mach. Learn. 28 (2) (1997) 133–168.
[45] D.J.C. Mackay, Information-based objective functions for active data selection, Neural Comput. 4 (4) (1989) 590–604.
[46] A. Biswas, D. Parikh, Simultaneous active learning of classifiers & attributes via relative feedback, Computer Vision and Pattern Recognition, 2013. pp. 644–651.
[47] L. Liang, K. Grauman, Beyond comparing image pairs: setwise active learning for relative attributes, Computer Vision and Pattern Recognition, 2014. pp. 208–215.
[48] T. Mensink, J. Verbeek, G. Csurka, Learning structured prediction models for interactive image labeling, Computer Vision and Pattern Recognition, 2011. pp. 833–840.
[49] Y. Zhang, D.-Y. Yeung, Multi-task boosting by exploiting task relationships, European Conference on Machine Learning and Knowledge Discovery in Databases, 2012. pp. 697–710.
[50] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, ImageNet large scale visual recognition challenge, Int. J. Comput. Vis. 115 (3) (2015) 211–252.
[51] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R.B. Girshick, S. Guadarrama, T. Darrell, Caffe: convolutional architecture for fast feature embedding, Multimedia, 2014. pp. 675–678.
[52] J. Xiao, J. Hays, K.A. Ehinger, A. Oliva, A. Torralba, Sun database: large-scale scene recognition from abbey to zoo, Computer Vision and Pattern Recognition, 2010. pp. 3485–3492.