Deep Relative Attributes

Xiaoshan Yang, Tianzhu Zhang, *Member, IEEE,* Changsheng Xu, *Fellow, IEEE,*
Shuicheng Yan, *Senior Member, IEEE,* M. Shamim Hossain, *Senior Member, IEEE,*
Ahmed Ghoneim, *Member, IEEE*

***Abstract*—Relative attribute learning aims to learn the ranking function describing the relative strength of the attribute. Most of current learning approaches learn linear ranking function for each attribute by use of the hand-crafted visual features. Different from the existing work, in this paper, we propose a novel deep relative attributes (DRA) algorithm to learn visual features and the effective nonlinear ranking function to describe the relative attribute of image pairs in a unified framework. Here, visual features and the ranking function are learned jointly, and they can benefit each other. The proposed DRA model is comprised of 5 convolutional neural layers, 5 fully connected layers, and a relative loss function which contains the contrastive constraint and the similar constraint corresponding to the ordered image pairs and the un-ordered image pairs, respectively. To train the DRA model effectively, we make use of the transferred knowledge from the large scale visual recognition on ImageNet [1] to the relative attribute learning task. We evaluate the proposed DRA model on three widely used datasets. Extensive experimental results demonstrate that the proposed DRA model consistently and significantly outperforms the state-of-the-art relative attribute learning methods. On the public OSR, PubFig and Shoes datasets, compared with the previous relative attribute learning results [2], the average ranking accuracies have been significantly improved by about 8%, 9%, and 14%, respectively.**

***Index Terms*—Relative attributes, Deep learning.**

I. INTRODUCTION

Visual attributes are intrinsic properties in images with human-designed names (e.g., ‘natural’, ‘smiling’), and they are valuable as higher semantic cues than low level visual features in many interesting scenarios. For example, researchers have shown that visual attributes are valuable for facial verification [3], object recognition [4], [5], [6], image retrieval/search [7], [8], [9], [10], [11], [12], [13], [14], video retrieval and recommendation [15], [16], generating descriptions of unfamiliar objects [17] and transfer learning [18], [19], [20], [21]. Many attributes mining and learning methods have also been proposed [22], [23], [24]. In these methods, the attributes are binary, which indicates the presence (or absence) of a certain property in an image. Compared with the binary attributes, using relative attributes is a much richer way for humans to describe objects semantically with relative visual properties. The consecutive relative values of the attributes can reflect not only whether the attribute appears in an image, but also the strength of the attribute. As a richer language of visual description than the commonly used binary attributes, relative attribute learning has gained much attention and can be used in many applications especially social event analysis [25], [26], [27], [28], and zero-shot learning [2], [29], [30], [31].

Most of the existing relative attribute learning algorithms are based on the ranking SVM framework [2], [29], [30] to learn a ranking function for each attribute. Here, the value of the ranking score denotes the strength of the attribute in an image with respect to other images. Despite remarkable progress in this field, there exists significant room for improvement, especially in the following three aspects: **(1)** Existing relative attribute methods rely on traditional hand-crafted features, such as gist descriptor [2], [31] and color histogram [2], [31], which may not optimally capture the most appropriate visual features to describe relative attributes. **(2)** Most of the relative attribute learning methods [2], [29], [30], [31] only learn a linear or shallow ranking function to obtain the relative score of image pair for a specific attribute. The linear or shallow models are simple, and may not best represent the mapping from visual features of images pair to the relative score of attributes. **(3)** Existing relative attribute learning methods [2], [29], [30], [31] perform feature extraction and ranking function learning separately, which cannot capture the most useful features for describing visual attributes of images.

To deal with the above issues, we propose a novel deep relative attributes (DRA) algorithm to learn visual features and the more effective nonlinear ranking function to describe the relative attribute of image pair in a unified framework. In this paper, with the same pipeline as in [2], we learn the ranking function for each relative attribute independently. As shown in Figure 1, the proposed DRA model is comprised of 5 convolutional neural layers, 5 fully connected layers, and a relative loss function. The convolutional neural layers are adopted to learn middle level visual features for attribute representation, and the fully connected layers are adopted to learn a nonlinear ranking function to map the learned visual features by the convolutional neural layers to the relative score of a specific attribute. The relative loss function contains the contrastive constraint of the ordered image pairs and the similar constraint of the un-ordered image pair. As a result,

X. Yang, T. Zhang and C. Xu are with the National Lab of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: xiaoshan.yang@nlpr.ia.ac.cn; tzzhang@nlpr.ia.ac.cn; csxu@nlpr.ia.ac.cn).

S. Yan is with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117576, Singapore (e-mail: eleyans@nus.edu.sg).

M. Shamim Hossain and A. Ghoneim are with the Department of Software Engineering, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia (e-mail: mshossain@ksu.edu.sa; ghoneim@ksu.edu.sa).

A. Ghoneim is also with the Department of Computer Science, College of Science, Menoufia University, Menoufia 32721, Egypt.

Manuscript received 2015; accepted 2016.

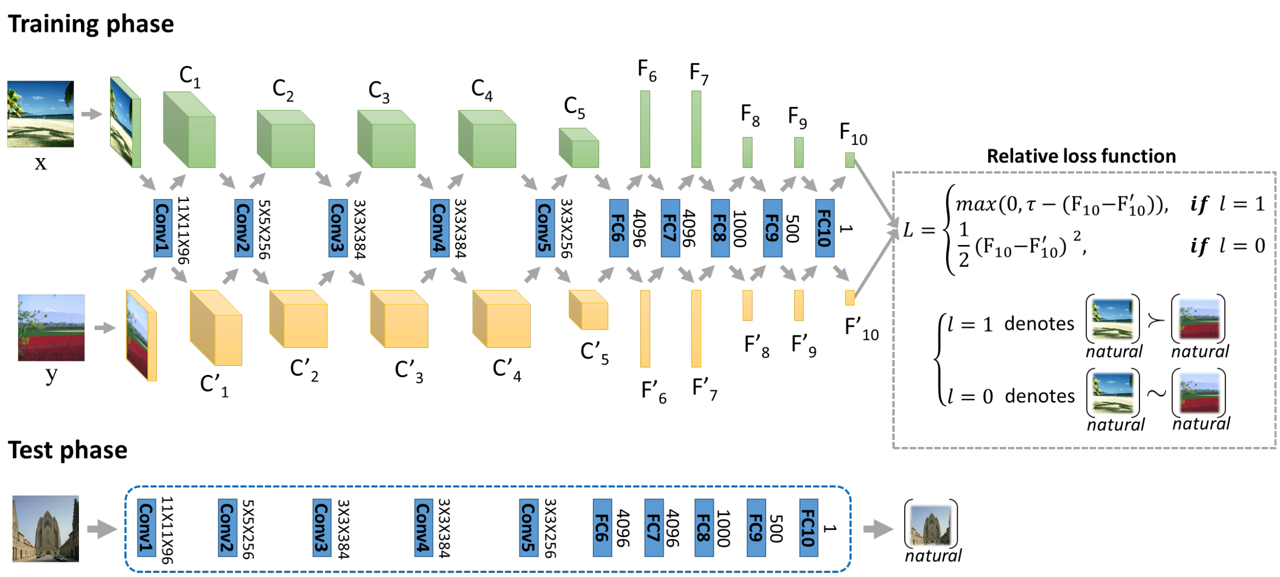


Fig. 1. The architecture of the proposed deep relative attributes model, which consists of 5 convolutional neural layers and 5 fully connected layers. The one dimensional output of the last fully connected layer denotes the relative strength of images with regard to the *natural* attribute. The relative loss function contains the contrastive constraint of the ordered image pair and the similar constraint of the un-ordered image pair.

the relative loss function can make the output of the last fully connected layer reflect the relative score of the attribute. In our DRA model, the visual features and the ranking function are learned jointly in a unified convolutional neural network framework, and they can benefit each other. More effective visual features can improve the ranking accuracy of the relative attribute, while a better ranking function can be used to guide the more appropriate visual features learning. In the proposed DRA model, there are million-scale parameters, such as the convolutional kernels in the convolutional layer, and the weights and the bias in the fully connected layer, which require large scale labeled data for training. However, the existing largest public dataset for relative attribute learning only contains about 10 thousand-scale labeled images. To overcome this issue, we make use of the transferred knowledge from the large scale visual recognition on ImageNet [1]. Thus, we adopt the trained image classification model [32], [33] to initialize the low level layers of the proposed DRA model. Then, the proposed DRA model is trained on the relative attributes dataset with the labeled image pairs.

The contributions of the proposed DRA are four-fold:

- To the best of our knowledge, the proposed DRA model is the first work to learn **relative attributes** directly using CNNs, though there are deep CNN based methods for **binary attributes**.
- Compared with conventional hand-crafted features and linear ranking SVM based relative attribute learning, we adopt convolutional neural networks to learn more effective nonlinear functions and map the original images to obtain their relative strength values of the attribute.
- In the proposed DRA model, the visual features and effective nonlinear ranking functions are learned jointly in a unified framework to benefit each other.
- Extensive experimental results demonstrate that the proposed DRA model consistently and significantly outperforms state-of-the-art relative attribute learning methods on three challenging benchmarks. On the public OSR, PubFig and Shoes datasets, compared with the previous relative attribute learning methods [2], the average ranking accuracies have been significantly improved by about **8%**, **9%**, and **14%**, respectively.

The rest of this paper is organized as follows. In Section II, we summarize the related work. Our method and the optimization are introduced in Section III. Experimental results are reported and analyzed in Section IV. Finally, we conclude the paper in Section V.

II. Related Work

In this section, we review the related work about binary attributes, relative attributes, and deep learning which are the three most related topics to the proposed method.

Binary attributes: Visual attribute learning allows prediction of color or texture types [34], and can also help obtain a mid-level cue for object or face recognition [35], [3], [5]. Attributes can also facilitate zero-shot learning [35], [4], [36] and part localization [17], [37], [38]. To avoid defining attribute vocabularies manually, some methods aim to explore attribute-related concepts on the Web [39], [40], extract them from existing knowledge sources [4], [6] or discover them interactively [41]. There are also some methods proposed for attribute mining. Zhang *et al.* [42] propose to automatically discover attribute from an arbitrary set of image and text pairs. To detect generic facial attribute by leveraging visual and contextual cues, Chen *et al.* propose to automatically acquire training images from publicly available community-contributed photos in an unsupervised manner [43]. To detect various facial attributes such as gender, age and more which consume more computation and storage resources, Lin *et al.*

propose a compression framework to find fewer significant latent human topics to approximate more facial attributes [44]. In contrast to the relative attribute learning approaches, all such methods restrict the attribute to be binary without considering the relative information through attributes.

Relative attributes: Using relative attributes is a semantically rich way to describe and compare objects in the world, and more powerful than existing binary attributes in uniquely identifying an image. In [2], relative attributes are first proposed based on the learning to rank framework, whose ranking function is learned for each attribute to denote the relative values or ranking scores. The ranking functions of all attributes are learned independently in [2], and ignore the correlations among multiple attributes. To improve this method, the multi-task learning is introduced in [31] to learn ranking functions of multiple attributes jointly. Relative attributes are also used for many other applications. In [30], the active learning framework is adopted to include feedback on not only the label but also the attributes. In [29], as human-interpretable mid-level visual concepts, the relative attributes are used for a supervisor to provide feedback to the classifier. In [45], a relative attribute feedback strategy is adopted for image search. Here, the ranking functions of attributes are learned iteratively according to the user feedback to make the images with top ranking scores close to the user's preference. These existing relative attributes methods are based on hand-crafted features to learn linear functions to map these features to the relative scores of the corresponding attributes. Different from these methods, the proposed deep relative attributes can learn image features and more effective nonlinear functions for attributes in a unified framework.

Deep learning: In recent years, deep models including deep belief networks (DBNs) [46], deep Boltzmann machines (DBMs) [47], stacked auto-encoders (SAEs) [48], [49] and convolutional neural networks (CNNs) [50], [32] have drawn much attention due to their encouraging performances. As effective feature learning methods, the deep models have been widely used in many applications, such as large scale object recognition [32], [1], [51], [52], [53], [54], human action recognition [55], face point detection [56], and social event analysis [57]. The most relevant methods to the proposed model are deep metric learning for face verification [58], [59] and deep ranking for fine-grained image similarity learning [60]. In the deep metric learning methods [58], [59], the trained Siamese networks aim to predict whether the two input images are the same person or not. The input of the networks is always an image pair. In contrast, the top fully connected layer of the proposed DRA maps the features of two images to two continuous strength values of the attribute. The great value denotes the strong strength of the attribute in the image while small value denotes weak strength. In the deep metric learning methods, the difference of two images is measured by Euclidean distance [58] and absolute difference [59], respectively. In contrast, we adopt the direct difference among two output values. Thus after training, the different output values for two images are able to not only represent whether two images are similar, but also give their strength order with regard to the attribute. In the test phase, the previous deep metric learning methods can only predict whether two images belong to the same person or not. Thus, the input must be an image pair. In contrast, the learned DRA model can predict the strength value of any individual image. The input is a single image. Moreover, the inputs to the networks [58] are hand-crafted low-level visual features including DSIFT, LBP and SSIFT. The deep ranking [60] learns a ranking function by a triplet-based network architecture, where each network is a combination of the convolutional neural networks and two low-resolution paths to extract low resolution visual features. Different from this method, the proposed deep relative attributes algorithm adopts a single CNN. In the forward propagation, different inputs and outputs of the two images are computed by the same parameters in each layer of the CNN. Though the single CNN model is adopted, the two images are propagated forward through the convolutional and fully connected layers, separatively.

III. The proposed Deep Relative Attributes

In this section, we firstly show the problem description of deep relative attribute learning. Then, we introduce the deep network structure of the proposed model and the detail of each layer. At last, we illustrate the forward and backward propagation schemes to optimize the proposed model.

A. Problem Description

The goal of relative attribute learning is to learn a ranking function for each attribute with a number of human labeled ordered or un-ordered image pairs. Given a test image, the score of the ranking function can be used to denote the strength of each attribute in the image [2]. In this paper, we focus on learning the ranking functions for the relative attributes independently. For simplicity, we adopt $f(\mathbf{x})$ to denote the ranking function corresponding to a specific attribute $\mathbf{a}$. For the attribute $\mathbf{a}$, we use $\mathcal{P}$ to denote a set of ordered image pairs and $\mathcal{Q}$ a set of un-ordered image pairs. If image pair $(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{P}$, it means that the image $\mathbf{x}_i$ has a higher relative value of attribute $\mathbf{a}$ than image $\mathbf{y}_i$. If image pair $(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{Q}$, the image $\mathbf{x}_i$ and image $\mathbf{y}_i$ have similar relative values of attribute $\mathbf{a}$. With these notations, the relative attribute learning for the attribute $\mathbf{a}$ can be formulated as learning $f(\mathbf{x})$ by satisfying the following constraints:

$$\forall(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{P} \quad f(\mathbf{x}_i) > f(\mathbf{y}_i) \tag{1}$$

$$\forall(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{Q} \quad f(\mathbf{x}_i) = f(\mathbf{y}_i) \tag{2}$$

In the traditional attribute learning methods [2], [31], the hand-crafted features are adopted, and the learned ranking function $f(\mathbf{x})$ is linear. Different from these methods, our aim is to learn visual features and nonlinear ranking function jointly in a unified framework to benefit each other under convolutional neural networks. The details are introduced in the next subsection.

B. Deep Network Structure

To achieve the above goal, we propose a novel deep relative attribute learning model as shown in Figure 1. Here, we show the training and testing process of the DRA model with image pairs for the attribute *natural*. The DRA model contains 5 convolutional layers (Conv1, Conv2, Conv3, Conv4, Conv5) and 5 fully connected layers (FC6, FC7, FC8, FC9, FC10). Different from the traditional CNNs [32], in the training phase, the input to the DRA model is an image pair $(\mathbf{x},\mathbf{y})$ with relative attribute assignment l which denotes the label of the image pair. The $l=1$ means that image $\mathbf{x}$ has larger attribute value than image $\mathbf{y}$ (ordered image pair) while the $l=0$ means the two images have similar attribute values (un-ordered image pair). In the forward propagation, different inputs and outputs of the two images are computed by the same parameters in each layer. Though the same CNN model is adopted, the two images are propagated forward through the convolutional and fully connected layers, separatively. The outputs (F_{10} and F'_{10}) in the last fully connected layer denote the relative values of the corresponding input images $\mathbf{x}$ and $\mathbf{y}$ with regard to the attribute *natural*. Following the fully connected layer, a relative loss function is adopted to constrain the relative output values of the image pair. By the contrastive constraint $max(0,\tau-(\mathrm{F}_{10}-\mathrm{F}'_{10}))$ and the similar constraint $\frac{1}{2}(\mathrm{F}_{10}-\mathrm{F}'_{10})^2$ for the *natural* attribute in the loss function, the ordered image pair ($l=1$) will be constrained to have the discrepant outputs while the un-ordered image pair ($l=0$) will be constrained to have the same or very close outputs. In the test phase, the single CNN with the learned parameters is used to predict the strength value of any individual image with regard to the *natural* attribute. It is worth noting that, as shown in Figure 1, for each convolutional layer, we show the size and number of the convolutional filters. For each fully connected layer, we show the dimension of the output feature vector. We do not show the pooling, normalization and dropout layers after the convolutional layers or the fully connected layer. For all the convolutional layers, we adopt the same manner of pooling or normalization as AlexNet [32], [33]. The dropout is only carried out after the F_6 and F_7 layers.

Convolutional Layers. For the m^{th} convolution layer, we denote its output as $\mathbf{h}_m(\mathbf{x}) = \mathbf{s}(\mathbf{W}_m * \mathbf{h}_{m-1}(\mathbf{x}) + \mathbf{b}_m)$, $m \in \{1,...,5\}$. Here, $*$ denotes the convolutional operation, $\mathbf{W}_m$ and $\mathbf{b}_m$ are the convolutional kernel and bias. $\mathbf{s}(\mathbf{x}) = max(0,\mathbf{x})$ denotes the non-saturating nonlinearity activation function which is also used as the rectified linear units (RELU) in [32].

Fully Connected Layers. For the m^{th} fully connected layer, we denote the output as $\mathbf{h}_m(\mathbf{x}) = \mathbf{s}(\mathbf{W}_m\mathbf{h}_{m-1}(\mathbf{x})+\mathbf{b}_m)$, $m \in \{6,...,10\}$. Here, $\mathbf{W}_m$ and $\mathbf{b}_m$ are the weight matrix and bias, respectively. For the activation function, the same rectified linear units $\mathbf{s}(\mathbf{x}) = max(0,\mathbf{x})$ as in the convolutional layer is adopted.

Relative Loss Function. For a specific attribute $\mathbf{a}$, the loss function for training the ranking function $f(\mathbf{x})$ is defined as the sum of the contrastive constraint, the similar constraint and the regularization item:

$$\mathcal{L} = \frac{1}{2|\mathcal{G}|}\sum_{(\mathbf{x}_i,\mathbf{y}_i)\in\mathcal{G}}\left[l_i\mathcal{L}_p(\mathbf{x}_i,\mathbf{y}_i)+(1-l_i)\mathcal{L}_q(\mathbf{x}_i,\mathbf{y}_i)\right]+\lambda\|\Theta\|_F^2. \tag{3}$$

Here, the $\mathcal{G} = \mathcal{P} \cup \mathcal{Q}$ contains all ordered and un-ordered image pairs annotated for a specific attribute. l_i denotes the label of the i^{th} image pair. $l_i = 1$ means that image $\mathbf{x}_i$ has larger attribute values than image $\mathbf{y}_i$ (ordered image pair) while $l_i = 0$ means the two images have similar attribute values (un-ordered image pair). $\mathcal{L}_p(\mathbf{x}_i,\mathbf{y}_i) = max(0,\tau-(f(\mathbf{x}_i)-f(\mathbf{y}_i)))$ and $\mathcal{L}_q(\mathbf{x}_i,\mathbf{y}_i) = (f(\mathbf{x}_i)-f(\mathbf{y}_i))^2$ denote the contrastive constraint for the ordered image pairs and the similar constraint for the un-ordered image pairs respectively. For the image $\mathbf{x}_i$ and the image $\mathbf{y}_i$, $f(\mathbf{x}_i)$ and $f(\mathbf{y}_i)$ denote the one dimensional output attribute strength values (F_{10} and F'_{10} in Figure 1) at the top fully connected layer. Θ contains all parameters of the proposed DRA model including the convolution kernels in the convolutional layers, the transformation matrices in the fully connected layers and the biases. In the training phase, the relative loss is used to learn the parameters of the CNN model. For the image $\mathbf{x}_i$ and the image $\mathbf{y}_i$, the learned CNN model can output values $f(\mathbf{x}_i)$ and $f(\mathbf{y}_i)$ which have the same ranking order with the labeled order of the two images with regard to the attribute. The τ controls the relative margin among the attribute values of ordered image pair. During training, $(f(\mathbf{x})-f(\mathbf{y}))$ can be larger or smaller than τ but will be constrained to be no less than τ. The λ is used to control the regularization item.

We give more detailed explanations of the loss function in two cases as follows. **(1)** If $l_i = 0$, which means image $\mathbf{x}_i$ has the same attribute value with image $\mathbf{y}_i$, the contrastive constraint will be zero while the minimization of the similar constraint $(f(\mathbf{x}_i)-f(\mathbf{y}_i))^2$ will make $f(x_i)$ and $f(y_i)$ have the same value. **(2)** If $l_i = 1$, which means image $\mathbf{x}_i$ has greater attribute value than image $\mathbf{y}_i$, the similar constraint will be zero while the minimization of the contrastive constraint $max(0,\tau-(f(\mathbf{x}_i)-f(\mathbf{y}_i)))$ will make $f(\mathbf{x}_i)$ have greater value than $f(\mathbf{y}_i)$. For this case, we illustrate it with two subcases. **(a)** If $f(\mathbf{x}_i) \geq f(\mathbf{y}_i)+\tau$, the loss will be zero which is just what we want. Thus the minimization will do nothing and have no any penalty. **(b)** If $f(\mathbf{x}_i) < f(\mathbf{y}_i)+\tau$, the loss will be a positive value $\tau-(f(\mathbf{x}_i)-f(\mathbf{y}_i))$. Thus the minimization will make it close to zero until $f(\mathbf{x}_i) > f(\mathbf{y}_i)+\tau$.

C. Optimization

The optimization of the proposed DRA model is similar to the conventional neural networks, where the stochastic gradient descent is adopted. The kernels in the convolutional layers and the weights in the fully connected layers are updated through the forward and backward propagations. In the forward propagation of the training phase, given two input images, different outputs of the two images in each layer are computed by the same operation. Though the same CNN model is adopted, the two images are propagated forward through the convolutional and fully connected layers separatively. In the backward propagation, the gradients with regard to the outputs in each layer are calculated for the two images separatively. Then, the gradients with regard to the parameters in each layer are calculated based on the gradients of the outputs for two images.

(1) Forward propagation. In the forward propagation, for an image pair $(\mathbf{x}_i, \mathbf{y}_i)$, the image $\mathbf{x}_i$ will be propagated through the convolutional layers and the fully connected layers: $\mathbf{x} \rightarrow \mathrm{C}_1 \rightarrow \mathrm{C}_2 \rightarrow \mathrm{C}_3 \rightarrow \mathrm{C}_4 \rightarrow \mathrm{C}_5 \rightarrow \mathrm{F}_6 \rightarrow \mathrm{F}_7 \rightarrow \mathrm{F}_8 \rightarrow \mathrm{F}_9 \rightarrow \mathrm{F}_{10}$. Meanwhile, the image $\mathbf{y}_i$ will also be propagated through the same convolutional layers and the same fully connected layers: $\mathbf{y} \rightarrow \mathrm{C}'_1 \rightarrow \mathrm{C}'_2 \rightarrow \mathrm{C}'_3 \rightarrow \mathrm{C}'_4 \rightarrow \mathrm{C}'_5 \rightarrow \mathrm{F}'_6 \rightarrow \mathrm{F}'_7 \rightarrow \mathrm{F}'_8 \rightarrow \mathrm{F}'_9 \rightarrow \mathrm{F}'_{10}$. The outputs F_{10} and F'_{10} of the last fully connected layer denote the relative strength values of the two images with regard to the attribute. Then the relative loss of the networks is computed based on the outputs of the fully connected layers F_{10} and F'_{10}.

(2) Backward propagation. In the backward propagation, the partial gradients of the loss function in Eq. (3) are firstly computed with regard to the outputs $f(\mathbf{x}_i) = \mathrm{F}_{10}$ and $f(\mathbf{y}_i) = \mathrm{F}'_{10}$ of the last connected layer. Then the errors computed in the loss function will be propagated backward to the remaining fully connected layers and the convolutional layers, the parameters Θ including the weights $\{\mathbf{W}_m\}_{m=1}^M$ and biases $\{\mathbf{b}_m\}_{m=1}^M$ of the fully connected layers and the kernels in the convolutional layers will be updated. In a specific layer, for an input image pair $(\mathbf{x}_i, \mathbf{y}_i)$, once the gradients of the outputs for image $\mathbf{x}_i$ and $\mathbf{y}_i$ are separatively computed, the gradients of the same parameters will be computed based on them. Then the parameters will be updated according to their partial gradients. More details of the partial gradients in the relative loss function and the fully connected layer are listed as follows. The partial gradients in the convolutional layers are computed in the similar way as the conventional convolutional neural networks.

Partial gradients in the relative loss function. We use $f(\mathbf{x})$ to denote the output of the last fully connected layer F_{10} for the attribute. If we denote $d_i = f(\mathbf{x}_i) - f(\mathbf{y}_i)$ and $sign(.)$ as a binary sign function, the partial gradients of the loss function in Eq. (3) with regard to $f(\mathbf{x})$ can be computed as:

$$\frac{\partial \mathcal{L}}{\partial f(\mathbf{x}_i)} = \begin{cases} \frac{1}{|\mathcal{G}|} d_i, & \text{if } (\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{Q} \\ -\frac{1}{2|\mathcal{G}|} sign(\tau - d_i), & \text{if } (\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{P} \end{cases} \tag{4}$$

$$\frac{\partial \mathcal{L}}{\partial f(\mathbf{y}_i)} = \begin{cases} -\frac{1}{|\mathcal{G}|} d_i, & \text{if } (\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{Q} \\ \frac{1}{2|\mathcal{G}|} sign(\tau - d_i), & \text{if } (\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{P} \end{cases} \tag{5}$$

Partial gradients in the fully connected layer. For simplicity, we denote $f(x) = \mathbf{h}_M(\mathbf{x}) = \mathbf{s}(\mathbf{W}_M \mathbf{h}_{M-1}(\mathbf{x}) + \mathbf{b}_M)$ as the output of the last fully connected layer. The partial gradients of the loss function in Eq. (3) with regard to the weights $\mathbf{W}_M$ of the M^{th} fully connected layer (F_{10}) can be computed as:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{W}_M} = \sum_{(i,j)\in\mathcal{P}} \Big(\frac{\partial \mathcal{L}(\mathbf{x}_i)}{\partial \mathbf{W}_M} + \frac{\partial \mathcal{L}(\mathbf{y}_i)}{\partial \mathbf{W}_M}\Big) + \sum_{(i,j)\in\mathcal{Q}} \Big(\frac{\partial \mathcal{L}(\mathbf{x}_i)}{\partial \mathbf{W}_M} + \frac{\partial \mathcal{L}(\mathbf{y}_i)}{\partial \mathbf{W}_M}\Big) + \lambda \mathbf{W}_M. \tag{6}$$

The partial gradients with regard to the bias is:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{b}_M} = \sum_{(i,j)\in\mathcal{P}} \Big(\frac{\partial \mathcal{L}(\mathbf{x}_i)}{\partial \mathbf{b}_M} + \frac{\partial \mathcal{L}(\mathbf{y}_i)}{\partial \mathbf{b}_M}\Big) + \sum_{(i,j)\in\mathcal{Q}} \Big(\frac{\partial \mathcal{L}(\mathbf{x}_i)}{\partial \mathbf{b}_M} + \frac{\partial \mathcal{L}(\mathbf{y}_i)}{\partial \mathbf{b}_M}\Big). \tag{7}$$

Here, $\frac{\partial \mathcal{L}(\mathbf{x}_i)}{\partial \mathbf{W}_M}$ denotes the contribution of image $\mathbf{x}_i$ to the partial gradients of the whole loss with regard to $\mathbf{W}_M$, and $\frac{\partial \mathcal{L}(\mathbf{y}_i)}{\partial \mathbf{W}_M}$ denotes the contribution of image $\mathbf{y}_i$.

$$\frac{\partial \mathcal{L}(\mathbf{x}_i)}{\partial \mathbf{W}_M} = \begin{cases} \frac{1}{|\mathcal{G}|} d_i \frac{\partial \mathbf{h}_M(\mathbf{x}_i)}{\partial \mathbf{W}_M}, & \text{if } (\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{Q} \\ \frac{-1}{2|\mathcal{G}|} sign(\tau - d_i) \frac{\partial \mathbf{h}_M(\mathbf{x}_i)}{\partial \mathbf{W}_M}, & \text{if } (\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{P} \end{cases} \tag{8}$$

$$\frac{\partial \mathcal{L}(\mathbf{y}_i)}{\partial \mathbf{W}_M} = \begin{cases} -\frac{1}{|\mathcal{G}|} d_i \frac{\partial \mathbf{h}_M(\mathbf{y}_i)}{\partial \mathbf{W}_M}, & \text{if } (\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{Q} \\ \frac{1}{2|\mathcal{G}|} sign(\tau - d_i) \frac{\partial \mathbf{h}_M(\mathbf{y}_i)}{\partial \mathbf{W}_M}, & \text{if } (\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{P} \end{cases} \tag{9}$$

Here, the partial gradients of $\mathbf{h}_M(\mathbf{x})$ are computed as

$$\frac{\partial \mathbf{h}_M(\mathbf{x})}{\partial \mathbf{W}_M} = \mathbf{s}'\big(\mathbf{W}_M \mathbf{h}_{M-1}(\mathbf{x}) + \mathbf{b}_M\big)\big(\mathbf{h}_{M-1}(\mathbf{x})\big)^\top. \tag{10}$$

Similarly, the contributions to the gradients with regard to $\mathbf{b}_M$ of images $\mathbf{x}_i$ and $\mathbf{y}_i$ can be computed as follows:

$$\frac{\partial \mathcal{L}(\mathbf{x}_i)}{\partial \mathbf{b}_M} = \begin{cases} \frac{1}{|\mathcal{G}|} d_i \frac{\partial \mathbf{h}_M(\mathbf{x}_i)}{\partial \mathbf{b}_M}, & \text{if } (\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{Q} \\ \frac{-1}{2|\mathcal{G}|} sign(\tau - d_i) \frac{\partial \mathbf{h}_M(\mathbf{x}_i)}{\partial \mathbf{b}_M}, & \text{if } (\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{P} \end{cases} \tag{11}$$

$$\frac{\partial \mathcal{L}(\mathbf{y}_i)}{\partial \mathbf{b}_M} = \begin{cases} -\frac{1}{|\mathcal{G}|} d_i \frac{\partial \mathbf{h}_M(\mathbf{y}_i)}{\partial \mathbf{b}_M}, & \text{if } (\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{Q} \\ \frac{1}{2|\mathcal{G}|} sign(\tau - d_i) \frac{\partial \mathbf{h}_M(\mathbf{y}_i)}{\partial \mathbf{b}_M}, & \text{if } (\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{P} \end{cases} \tag{12}$$

Here, the partial gradients of $\mathbf{h}_M(\mathbf{x})$ are computed as

$$\frac{\partial \mathbf{h}_M(\mathbf{x})}{\partial \mathbf{b}_M} = \mathbf{s}'\big(\mathbf{W}_M \mathbf{h}_{M-1}(\mathbf{x}) + \mathbf{b}_M\big). \tag{13}$$

IV. EXPERIMENTS

In this section, we present experimental results on evaluation of the proposed algorithm against several state-of-the-art methods for relative attribute learning on three benchmark datasets.

A. Datasets

We evaluate the proposed algorithm on three popularly used relative attribute learning datasets:

(1) Outdoor Scene Recognition (OSR) [2]. This dataset contains 2688 images from 8 categories including *tall-building*, *inside-city*, *street*, *highway*, *coast*, *open-country*, *mountain*, *forest*. All these 8 categories are assigned with relative values of 6 attributes: *natural*, *open*, *perspective*, *size-large*, *diagonal-plane*, *depth-close*. The strength values of the 6 relative attributes on 8 classes are shown in Table I.

(2) Public Figure Face (PubFig) [2]. This dataset contains 800 images from 8 random identities including *Alex-Rodriguez*, *Clive-Owen*, *Hugh-Laurie*, *Jared-Leto*, *Miley-Cyrus*, *Scarlett-Johansson*, *Viggo-Mortensen*, *Zac-Efron*. All these 8 identities are assigned with relative values of 11

Classes	T	I	S	H	C	O	M	F
Natural	1	2	2	3	4	4	4	4
Open	1	2	2	4	4	4	3	1
Perspective	7	5	6	4	2	1	3	3
Size-large	5	3	3	4	4	2	2	1
Diagonal-plane	6	4	4	5	3	2	2	1
Depth-close	4	4	4	4	1	3	2	4

TABLE I
RELATIVE ORDERING OF ATTRIBUTES ON OSR DATASET. T(TALL BUILDING), I(INSIDE CITY), S(STREET), H(HIGHWAY), C(COAST), O(OPEN COUNTRY), M(MOUNTAIN), F(FOREST)

Classes	A	C	H	J	M	S	V	Z
Male	6	8	7	5	2	1	4	3
White	1	2	3	5	7	6	8	4
Young	5	3	2	4	8	6	1	7
Smiling	4	4	3	1	6	5	2	5
Chubby	8	4	3	2	6	7	1	5
Visible-forehead	5	5	5	1	3	4	5	2
Bushy-eyebrows	6	7	5	8	1	2	4	3
Narrow-eyes	4	6	5	2	1	3	7	8
Pointy-nose	1	2	8	3	3	4	3	7
Big-lips	7	5	1	2	6	8	3	4
Round-face	6	4	1	3	8	7	2	5

TABLE II
RELATIVE ORDERING OF ATTRIBUTES ON PUBFIG DATASET. A(ALEX RODRIGUEZ), C(CLIVE OWEN), H(HUGH LAURIE), J(JARED LETO), M(MILEY CYRUS), S(SCARLETT JOHANSSON), V(VIGGO MORTENSEN), Z(ZACEFRON)

Classes	A	B	C	F	H	P	R	SN	ST	W
Pointy-at-the-front	2	6	3	5	10	9	4	1	8	7
Open	3	2	8	5	7	6	1	4	9	10
Bright-in-color	6	1	2	8	4	3	10	7	9	5
Covered-with-ornaments	4	9	6	5	8	7	1	3	10	2
Shiny	2	9	4	3	6	5	8	1	10	7
High-at-the-heel	4	6	5	1	9	8	3	2	10	7
Long-on-the-leg	7	9	2	3	6	5	10	8	4	1
Formal	3	6	4	7	9	8	1	2	5	10
Sporty	10	5	6	7	4	3	8	9	1	2
Feminine	1	6	4	5	10	9	3	2	8	7

TABLE III
RELATIVE ORDERING OF ATTRIBUTES ON SHOES DATASET. A(ATHLETIC SHOES), B(BOOTS), C(CLOGS), F(FLATS), H(HIGH HEELS), P(PUMPS), R(RAIN BOOTS), SN(SNEAKERS), ST(STILETTO), W(WEDDING SHOES)

attributes including *Male*, *White*, *Young*, *Smiling*, *Chubby*, *Visible-Forehead*, *Bushy-Eyebrows*, *Narrow-Eyes*, *Pointy-Nose*, *Big-Lips*, *Round-Face*. The strength values of the 11 relative attributes on 8 classes are shown in Table II.

(3) Shoes Dataset [2], [45]. This dataset includes 14658 images with 10 categories of shoes collected from like.com, *athletic-shoes*, *boots*, *clogs*, *flats*, *high-heels*, *pumps*, *rain-boots*, *sneakers*, *stiletto*, *wedding-shoes*. All these categories are assigned with relative values of 10 attributes including *pointy-at-the-front*, *open*, *bright-in-color*, *covered-with-ornaments*, *shiny*, *high-at-the-heel*, *long-on-the-leg*, *formal*, *sporty*, *feminine*. The strength values of the 10 relative attributes on 10 classes are shown in Table III.

TABLE IV
RANKING ACCURACIES OF THE 4 COMPARED RELATIVE ATTRIBUTE LEARNING METHODS FOR ALL 6 ATTRIBUTES ON THE OSR DATASET.

Method	RA [2]	MTL [31]	RAD	DRA
Natural	94.82	96.47	98.20	99.47
Open	91.01	92.88	94.79	97.81
Perspective	86.56	88.39	93.66	97.19
Size-large	86.37	88.50	93.84	96.88
Diagonal-plane	88.00	90.87	94.88	98.46
Depth-close	88.35	89.05	95.18	97.24
Avg	89.19	91.03	95.09	**97.84**

B. Evaluated Algorithms

To evaluate the effectiveness of the proposed deep relative attributes algorithm, we compare the following attributes learning methods: **(1) Relative Attributes (RA)** [2] algorithm learns a linear ranking function for each attribute independently with a learning to rank formulation. **(2) Multi-Task Learning (MTL)** [31] method learns the linear ranking functions for all attributes simultaneously in a multi-task learning framework. **(3) Relative Attributes with Deep features (RAD)** adopts the relative attribute learning method [2] with deep learning features extracted from the seventh fully connected layer (F_7) of the AlexNet [32], [33] which is pre-trained on ImageNet images for the LSVRC2012 [1]. **(4)** The proposed **Deep Relative Attributes (DRA)** model learns deep visual feature and nonlinear ranking function jointly.

C. Implementation Details

We implement the proposed DRA model based on the public deep learning library Caffe [33], and train a convolutional neural network model for each attribute. The parameter τ in the relative loss function is set to 1.0. λ is set to $5e$-5. More details are illustrated as follows.

Layer Structure: In the first convolutional layer Conv1, the input images are resized to 227×227 uniformly without cropping in our experiment. To facilitate the weight transfer from the pre-trained model, the number of maps and the output dimensions of the first 5 convolutional layers (C_1, C_2, C_3, C_4 and C_5), and the first two fully connected layers (F_6 and F_7) are set according to the AlexNet provided in Caffe [33]. The remaining layers are newly created. The dimensions of the outputs in the F_8 layer, the F_9 layer, and the F_{10} layer are set to 1000, 500 and 1, respectively. The outputs in the last fully connected layer F_{10} denote the relative values of the images with regard to the attribute.

Weights Initialization: The weights in the first 5 convolutional layers and the first 2 fully connected layers are initialized according to the BVLC AlexNet model [33] which wins the large scale visual recognition challenge (LSVRC2012) [32]. The reference model is pre-trained on about 1 million images with 1000 categories on ImageNet. The remaining 3 fully connected layers F_8, F_9, and F_{10} are initialized with Gaussian filter with standard deviation 0.005 and constant bias 0.

TABLE V
RANKING ACCURACIES OF THE 4 COMPARED RELATIVE ATTRIBUTE LEARNING METHODS FOR ALL 11 ATTRIBUTES ON THE PUBFIG DATASET.

Method	RA [2]	MTL [31]	RAD	DRA
Male	82.57	84.52	84.08	90.82
White	79.14	80.11	76.29	87.12
Young	82.52	83.91	84.62	91.49
Smiling	81.37	82.19	82.82	92.68
Chubby	77.80	79.16	80.91	89.30
Visible-forehead	88.75	89.86	87.46	94.39
Bushy-eyebrows	80.63	82.06	81.61	90.19
Narrow-eyes	81.68	81.48	81.87	90.60
Pointy-nose	79.01	79.86	79.66	91.03
Big-lips	80.38	81.20	83.91	90.35
Round-face	82.37	83.43	85.46	91.99
Avg	81.47	82.52	82.61	**90.91**

TABLE VI
RANKING ACCURACIES OF THE 4 COMPARED RELATIVE ATTRIBUTE LEARNING METHODS FOR ALL 10 ATTRIBUTES ON THE SHOES DATASET.

Method	RA [2]	MTL [31]	RAD	DRA
Pointy-at-the-front	79.32	84.66	84.06	88.34
Open	76.41	77.37	80.04	87.02
Bright-in-color	53.09	64.06	66.36	74.97
Covered-with-ornaments	57.96	71.20	71.44	79.86
Shiny	66.61	80.53	80.66	86.92
High-at-the-heel	78.38	80.92	81.58	87.50
Long-on-the-leg	68.35	73.61	76.53	84.30
Formal	73.93	74.16	76.35	81.76
Sporty	69.84	80.46	81.74	87.72
Feminine	77.84	84.06	83.46	87.98
Avg	70.17	77.10	78.22	**84.64**

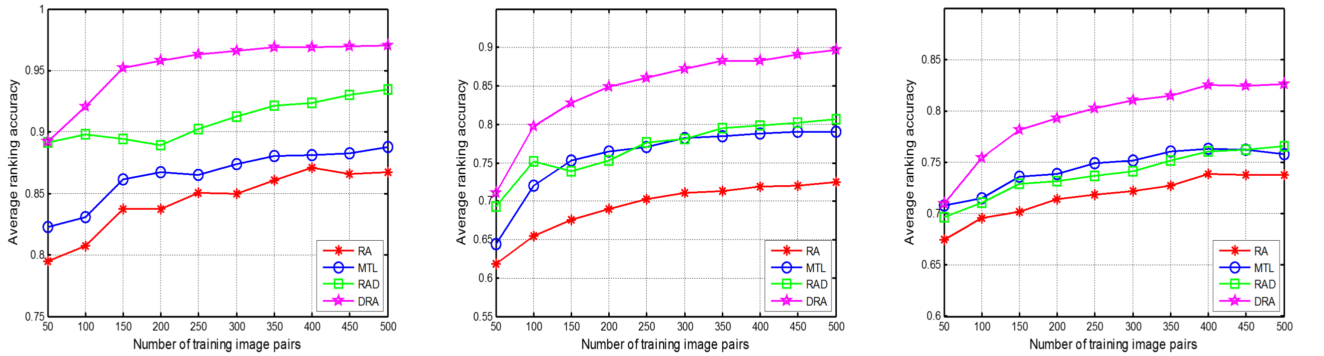
Learning Rate: Since there are only thousand-scale training images available for each attribute, we fix the learning rates in the first 4 convolutional layers (C_1, C_2, C_3, C_4) as zeros to avoid overfitting. The learning rates for the layers C_5, F_6 and F_7 are all set to 0.001, while the learning rates for the layers F_8, F_9 and F_{10} are all set to 0.01. The scheme for the learning rate setting is also consistent with the hierarchical nature of the features of different layers in the networks. As illustrated in [51], the output features of the bottom convolutional layers respond to corners and other edge/color conjunctions. The middle layers have more complex invariances, capturing similar textures (e.g. mesh patterns) while the top layers are more class-specific. Since the proposed convolutional networks for relative attribute learning are initialized with the weights of the model trained for the 1000-class classification task. The top layers need to be trained carefully due to the task difference. However, the bottom layers are more likely to share common features since they are the low level features, such as corners and other edge/color conjunctions. Thus, the learning rates of the bottom layers of the proposed DRA model can be assigned with small values or even fixed.

D. Learned Ranking Results

We train all the relative attribute learning methods, including RA [2], MTL [31], RAD, and the proposed DRA, using image pairs comprised of all the annotated training images in the dataset. On the OSR dataset, there are 240 training images which can generate about 10k image pairs for each attribute. On the PubFig dataset, it includes 241 training images, and can generate about 7k image pairs for each attribute. On the Shoes dataset, there are 240 training images which can generate about 6k image pairs for each attribute. We adopt the same evaluation scheme illustrated in [2]. Given a specific attribute, we predict the order of an image pair (i, j) in the test set by their relative values which are generated by the learnt relative attribute model for this attribute. The predictions are then compared to the ground-truth relative ordering. For all the evaluated algorithms, the ranking results on the OSR dataset, the PubFig dataset, and the Shoes dataset are shown in Table IV, Table V, and Table VI respectively. Note that, for the MTL [31] method, we use the public code provided by the authors. The time and memory costs are extremely large for the joint ranking learning. Therefore, we use about 3k image pairs for training due to the limitation of our computer hardware. But absolutely fair comparisons between the MTL model and the proposed DRA model can be found in Figure 2.

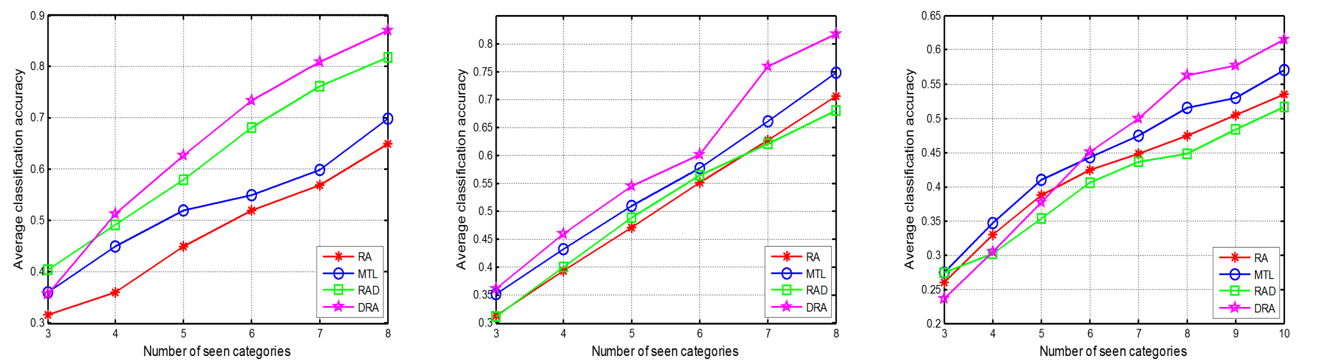
Based on the results in Tables IV, V, VI, it is clear that the proposed DRA model consistently and significantly outperforms the state-of-the-art methods on all datasets. Compared with the relative attribute learning (RA) method [2], which is based on linear ranking SVM, the average accuracy of our DRA approach is increased about **8%** on the OSR dataset, **9%** on the PubFig dataset, and **14%** on the Shoes dataset. The RAD method is based on the deep features extracted by the reference model trained for the large scale image classification task, and obtains better performance than the RA [2], which demonstrates the effectiveness of the deep visual features. However, the RAD method still cannot outperform the proposed DRA method. This is due to that, compared with the RAD method, 2 extra fully connected layers are added and trained in the proposed DRA for the relative attribute learning task. All these experimental results show that the proposed DRA method can learn not only much more effective task-specific visual features for image representation, but also much more effective nonlinear ranking functions to describe the relative scores of the attributes.

To show the effect of the number of training image pairs, we give the average accuracies of all attributes with different numbers of image pairs as shown in Figure 2. We can see that with a few number of image pairs, the proposed DRA model cannot show significantly better performances than other methods. This is because the proposed DRA algorithm needs more image pairs to learn the large number of model parameters. With more training image pairs, the gap between the proposed DRA model and other baseline methods is enlarged. It is worth noting that the proposed DRA can perform well with hundreds of training image pairs.



(a) OSR (b) PubFig (c) Shoes

Fig. 2. Average ranking accuracies of the compared 4 relative attribute learning methods (RA, MTL, RAD, and DRA) on the three datasets (OSR, PubFig, and Shoes) with different number of image pairs used for training.



(a) OSR (b) PubFig (c) Shoes

Fig. 3. Average classification accuracies of the zero shot learning on the three datasets (OSR, PubFig, and Shoes) with image pairs from different number of seen categories used for training the relative attribute models.

E. Zero-Shot Learning Results

The zero-shot learning is an application of relative attribute learning, and aims to classify N image categories where only S of them are provided with training images and no training images are provided for the other U categories [2]. Here, $N = S+U$ and the S categories are called "seen" categories while the U categories are called "unseen" during training.

With the same experimental setup as in [2], we adopt Gaussian distribution as the generative model and estimate the means $\{\mu_i\}_{i=1}^N$ and the covariance matrices $\{\Sigma_i\}_{i=1}^N$ for all the N categories. **(1)** For the S seen categories $\{\mathbf{c}_i^s\}_{i=1}^S$, we learn K predicting models $\{\mathbf{f}^k(\mathbf{x})\}_{k=1}^K$ using the proposed DRA method for all K relative attributes based on the category relationships with regard to each attribute. Then these K relative attributes models are used to predict the relative values of all K attributes for a given image. Thus each image $\mathbf{x}$ from the S seen categories can be represented as a K dimensional vector $\tilde{\mathbf{x}} \in \mathbb{R}^K$ indicating the relative values of all K attributes. The means $\{\mu_i\}_{i=1}^S$ and the covariance matrices $\{\Sigma_i\}_{i=1}^S$ of the S seen categories are estimated according to the relative values (or ranking-scores) of the training images. **(2)** For the U unseen categories $\{\mathbf{c}_i^u\}_{i=1}^U$, since there are no training images provided, the means $\{\mu_i\}_{i=1}^U$ and the covariance matrices $\{\Sigma_i\}_{i=1}^U$ of the generative Gaussian models are set based on the parameters of the seen categories and guided by the relative orders of the seen categories and the unseen categories with regard to all the K attributes. For example, for the k^{th} attribute $\mathbf{a}_k$, if the unseen category $\mathbf{c}_r^u$ is described as $\mathbf{c}_p^s \succ \mathbf{c}_r^u \succ \mathbf{c}_q^s$, then the k^{th} component of the mean μ_r^u is set to $(\mu_{pk}^s + \mu_{qk}^s)/2$. Here, $\mathbf{c}_p^s$ and $\mathbf{c}_q^s$ are the seen categories, μ_p^s and μ_q^s are their means of the Gaussian distributions. More details for generating the means $\{\mu_i\}_{i=1}^U$ and the covariance matrices $\{\Sigma_i\}_{i=1}^U$ for unseen categories could be found in [2].

During the testing, a new image $\mathbf{x}$ is assigned with a K dimensional vector $\tilde{\mathbf{x}}$ by the K relative attributes predicting models $\{\mathbf{f}^k(\mathbf{x})\}_{k=1}^K$. It is then assigned with a seen or unseen category based on the learned generative Gaussian models of the seen and unseen categories:

$$\mathbf{c}^* = \arg \max_{i \in 1,\ldots,N} P(\tilde{\mathbf{x}}|\mu_i, \Sigma_i) \tag{14}$$

The experimental results of the zero shot learning on the OSR, PubFig and Shoes datasets are shown in Figure 3. It is clear that, for the RA [2], MTL [31], RAD, and the proposed DRA methods, the zero shot image classification performances are improved significantly with the increase of the number of the seen categories. On all the three datasets, the MTL method consistently performs much better than the RA method due to the joint learning scheme. On the OSR datast, the proposed DRA model performs the best when the number of the seen categories is greater than 4. The RAD method shows better result than the proposed DRA when there are 3 seen categories. On the PubFig dataset, the DRA method performs better than all baselines especially when the seen categories is larger than 6. On the Shoes dataset, the MTL method performs the best when the number of seen categories is smaller than 6. However, the proposed DRA outperforms the MTL method when the number of seen categories is greater than 6.

F. Discussions

Convergence Analysis: To explore the convergence of the proposed relative convolutional neural networks, in Figure 4, we show the training losses of the DRA model trained with different numbers of iterations. Here, we show the convergency curves in the first 100 iterations of all the 6 attributes on the OSR dataset. For each attribute, 500 image pairs are used for training. We can see that the relative convolutional neural networks can converge quickly especially in the first 60 iterations. In Figure 5, we show the ranking accuracies of the proposed DRA model trained with different numbers of iterations. Here, we only show the accuracies of the 6 attributes in the first 100 iterations, which makes the same conclusion that the DRA can converge extremely fast at the beginning. As illustrated in Section IV-D, the Shoes dataset is the largest labeled image dataset for relative attribute learning, and only contains 240 training images which can generate about 6k image pairs for each attribute. With the scarce training data, it is important for the proposed DRA to have a fast convergence to obtain the prospective performance.

Layer Analysis: To show the effect of the layer structure in the proposed DRA, in Figure 6, we give the ranking accuracies on the OSR dataset with different layer settings. Here, the **DRA** is the proposed model, and the **DRA1** model has the DRA structure without the $\mathbf{F}_8$ layer. Compared with the DRA1 model, the proposed DRA has about **1%** accuracy improvements for all 6 attributes, which demonstrates the effectiveness of the fully connected layer. The **DRA2** denotes the DRA1 structure without dropout after the $\mathbf{F}_7$ layer. Compared with the DRA1, the performance degradation of the DRA2 shows that the dropout layer is indispensable for training deep neural networks. The **DRA3** denotes the DRA structure without the $\mathbf{F}_8$ and $\mathbf{F}_9$ layers. The large margin of the performance degradation shows that the $\mathbf{F}_8$ and $\mathbf{F}_9$ layers play an important role in improving the performance of the DRA for relative attribute learning.

Effect of τ: In the relative loss function 3, the τ controls the relative margin among the attribute values of the ordered image pair. Theoretically, the larger τ results in more distinguishable attribute values, thus obtains better ranking performance. However, the too large τ may increase the difficulty for the DRA model training. To support this point, we show the effect of τ on the OSR dataset in Figure 7. We can see that the performances are consistent while the τ is set between 0 and 2, and too large τ results in unstable ranking accuracies for all the 6 relative attributes.

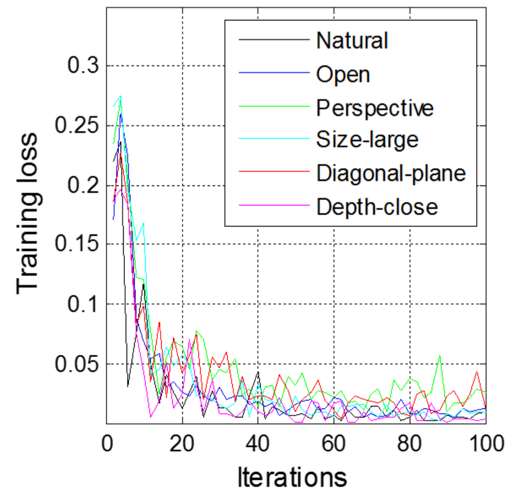


Fig. 4. Training losses with different iterations for all the 6 relative attributes on the OSR dataset.

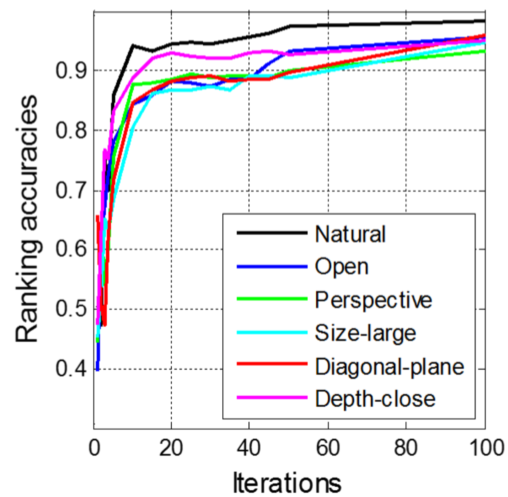


Fig. 5. Ranking accuracies with different iterations for all the 6 relative attributes on the OSR dataset.

V. Conclusions

In this paper, we proposed a novel deep relative attributes algorithm for relative attribute learning. The proposed DRA model adopts five convolutional layers and five fully connected layers to satisfy the two constraints of ordered image pairs and un-ordered image pairs for relative attribute learning. As a result, the proposed DRA can transform an image with raw pixels into the relative strength of the attribute. To facilitate the training of the DRA model, the weights in the low level layers are initialized with the corresponding weights of the large scale image classification model. We evaluate the proposed DRA on 3 popularly used datasets with state-of-the-art attribute learning methods, and the significant improvements demonstrate its effectiveness. In the future work, we will train the ranking functions for all relative attributes simultaneously by introducing the multi-task learning.

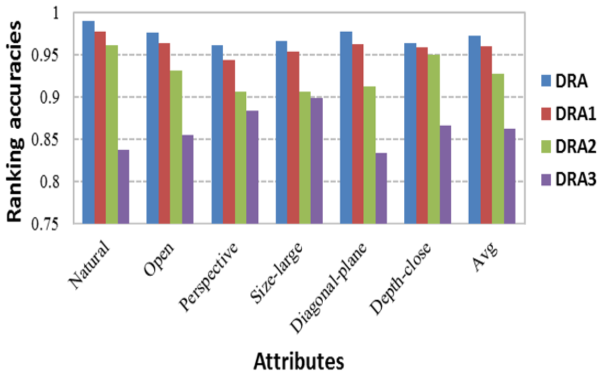


Fig. 6. Performance on the OSR dataset with different settings. DRA denotes the proposed deep relative attribute learning method. DRA1 model has the DRA structure without the $\mathbf{F}_8$ layer. DRA2 denotes the DRA1 structure without dropout after the $\mathbf{F}_7$ layer. DRA3 denotes the DRA structure without the $\mathbf{F}_8$ and $\mathbf{F}_9$ layers.

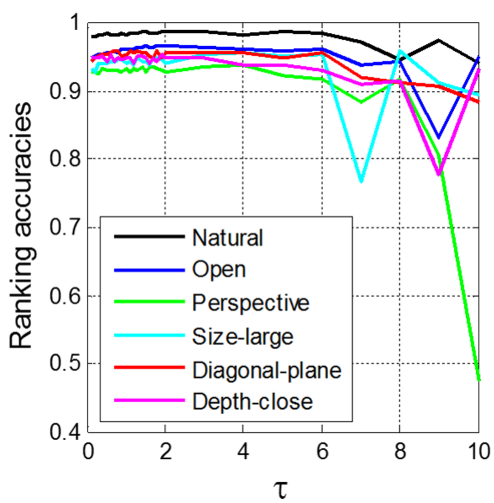


Fig. 7. Ranking accuracies with different τ for all the 6 relative attributes on the OSR dataset.

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China (No. 61225009, 61303173, 61432019, 61572498, 61532009, 61472379, 61572296), the National Program on Key Basic Research Project (973 Program, Project No. 2012CB316304), and the Importation and Development of High-Caliber Talents Project of Beijing Municipal Institutions under Grant IDHT20140224. The authors extend their appreciation to the Deanship of Scientific Research at King Saud University, Riyadh, Saudi Arabia for funding this work through the research group project no. RGP-229.

REFERENCES

[1] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," *arXiv*, 2014.

[2] D. Parikh and K. Grauman, "Relative attributes," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2011, pp. 503–510.

[3] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, "Attribute and simile classifiers for face verification," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2009, pp. 365–372.

[4] J. Wang, K. Markert, and M. Everingham, "Learning models for object recognition from natural language descriptions," in *Proceedings of the British Machine Vision Conference (BMVC)*, 2009, pp. 1–11.

[5] Y. Wang and G. Mori, "A discriminative latent model of object classes and attributes," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2010, pp. 155–168.

[6] S. Branson, C. Wah, F. Schroff, B. Babenko, P. Welinder, P. Perona, and S. Belongie, "Visual recognition with humans in the loop," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2010, pp. 438–451.

[7] B. Chen, Y. Chen, Y. Kuo, and W. H. Hsu, "Scalable face image retrieval using attribute-enhanced sparse codewords," *IEEE Transactions on Multimedia*, vol. 15, no. 5, pp. 1163–1173, 2013.

[8] X. Wang, T. Zhang, D. Tretter, and Q. Lin, "Personal clothing retrieval on photo collections by color and attributes," *IEEE Transactions on Multimedia*, vol. 15, no. 8, pp. 2035–2045, 2013.

[9] J. Huang, W. Xia, and S. Yan, "Deep search with attribute-aware deep network," in *Proceedings of the ACM International Conference on Multimedia, MM '14, Orlando, FL, USA, November 03 - 07, 2014*, 2014, pp. 731–732.

[10] X. Cao, X. Wei, X. Guo, Y. Han, and J. Tang, "Augmented image retrieval using multi-order object layout with attributes," in *Proceedings of the ACM International Conference on Multimedia, MM '14, Orlando, FL, USA, November 03 - 07, 2014*, 2014, pp. 1093–1096.

[11] J. Cai, Z. Zha, W. Zhou, and Q. Tian, "Attribute-assisted reranking for web image retrieval," in *Proceedings of the ACM Multimedia Conference, MM '12, Nara, Japan, October 29 - November 02, 2012*, 2012, pp. 873–876.

[12] Y. Lin, "Investigating 3d model and part information for improving content-based and attribute-based object retrieval," in *Proceedings of the ACM Multimedia Conference, MM '12, Nara, Japan, October 29 - November 02, 2012*, 2012, pp. 1409–1412.

[13] H. Zhang, Z. Zha, J. Bian, Y. Gao, H. Luan, and T. Chua, "Attribute feedback," in *Proceedings of the ACM Multimedia Conference, MM '12, Nara, Japan, October 29 - November 02, 2012*, 2012, pp. 1339–1340.

[14] H. Zhang, Z. Zha, Y. Yang, S. Yan, Y. Gao, and T. Chua, "Attribute-augmented semantic hierarchy: towards bridging semantic gap and intention gap in image retrieval," in *Proceedings of the ACM Multimedia Conference, MM '13, Barcelona, Spain, October 21-25, 2013*, 2013, pp. 33–42.

[15] L. Chen, P. Zhang, and B. Li, "Instructive video retrieval based on hybrid ranking and attribute learning: A case study on surgical skill training," in *Proceedings of the ACM International Conference on Multimedia, MM '14, Orlando, FL, USA, November 03 - 07, 2014*, 2014, pp. 1045–1048.

[16] P. Cui, Z. Wang, and Z. Su, "What videos are similar with you?: Learning a common attributed representation for video recommendation," in *Proceedings of the ACM International Conference on Multimedia, MM '14, Orlando, FL, USA, November 03 - 07, 2014*, 2014, pp. 597–606.

[17] A. Farhadi, I. Endres, D. Hoiem, and D. A. Forsyth, "Describing objects by their attributes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 1778–1785.

[18] C. H. Lampert, H. Nickisch, and S. Harmeling, "Attribute-based classification for zero-shot visual object categorization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 3, pp. 453–465, 2014.

[19] H. Li, D. Li, and X. Luo, "BAP: bimodal attribute prediction for zero-shot image categorization," in *Proceedings of the ACM International Conference on Multimedia, MM '14, Orlando, FL, USA, November 03 - 07, 2014*, 2014, pp. 1013–1016.

[20] Y. Han, Y. Yang, Z. Ma, H. Shen, N. Sebe, and X. Zhou, "Image attribute adaptation," *IEEE Transactions on Multimedia*, vol. 16, no. 4, pp. 1115–1126, 2014.

[21] Y. Han, F. Wu, X. Lu, Q. Tian, Y. Zhuang, and J. Luo, "Correlated attribute transfer with multi-task graph-guided fusion," in *Proceedings of the ACM Multimedia Conference, MM '12, Nara, Japan, October 29 - November 02, 2012*, 2012, pp. 529–538.

[22] Y. Chen, A. Cheng, and W. H. Hsu, "Travel recommendation by mining people attributes and travel group types from community-contributed photos," *IEEE Transactions on Multimedia*, vol. 15, no. 6, pp. 1283–1295, 2013.

[23] J. Bian, Z. Zha, H. Zhang, Q. Tian, and T. Chua, "Visual query attributes suggestion," in *Proceedings of the ACM Multimedia Conference, MM '12, Nara, Japan, October 29 - November 02, 2012*, 2012, pp. 869–872.

[24] X. Yang, T. Zhang, C. Xu, and M. S. Hossain, "Automatic visual concept learning for social event understanding," *IEEE Trans. Multimedia*, vol. 17, no. 3, pp. 346–358, 2015.

[25] S. Qian, T. Zhang, R. Hong, and C. Xu, "Cross-domain collaborative learning in social multimedia," in *Proceedings of the 23rd Annual ACM Conference on Multimedia Conference, MM '15, Brisbane, Australia, October 26 - 30, 2015*, 2015, pp. 99–108.

[26] S. Qian, T. Zhang, C. Xu, and J. Shao, "Multi-modal event topic model for social event analysis," *IEEE Trans. Multimedia*, vol. 18, no. 2, pp. 233–246, 2016.

[27] S. Qian, T. Zhang, C. Xu, and M. S. Hossain, "Social event classification via boosted multimodal supervised latent dirichlet allocation," *TOMCCAP*, vol. 11, no. 2, pp. 27:1–27:22, 2014.

[28] T. Zhang and C. Xu, "Cross-domain multi-event tracking via CO-PMHT," *TOMCCAP*, vol. 10, no. 4, pp. 31:1–31:19, 2014.

[29] A. Parkash and D. Parikh, "Attributes for classifier feedback," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2012, pp. 354–368.

[30] A. Biswas and D. Parikh, "Simultaneous active learning of classifiers & attributes via relative feedback," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 644–651.

[31] L. Chen, Q. Zhang, and B. Li, "Predicting multiple attributes via relative multi-task learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 1027–1034.

[32] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS)*, 2012, pp. 1106–1114.

[33] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv*, 2014.

[34] V. Ferrari and A. Zisserman, "Learning visual attributes," in *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS)*, 2007, pp. 433–440.

[35] C. H. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 951–958.

[36] O. Russakovsky and L. Fei-Fei, "Attribute learning in large-scale datasets," in *Proceedings of the European Conference on Computer Vision (ECCV), Workshop on Parts and Attributes*, 2010.

[37] G. Wang and D. A. Forsyth, "Joint learning of visual attributes, object classes and visual saliency," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2009, pp. 537–544.

[38] A. Farhadi, I. Endres, and D. Hoiem, "Attribute-centric recognition for cross-category generalization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010, pp. 2352–2359.

[39] M. Rohrbach, M. Stark, G. Szarvas, I. Gurevych, and B. Schiele, "What helps where - and why? semantic relatedness for knowledge transfer," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010, pp. 910–917.

[40] T. L. Berg, A. C. Berg, and J. Shih, "Automatic attribute discovery and characterization from noisy web data," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2010, pp. 663–676.

[41] D. Parikh and K. Grauman, "Interactively building a discriminative vocabulary of nameable attributes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 1681–1688.

[42] H. Zhang, Y. Yang, H. Luan, S. Yang, and T. Chua, "Start from scratch: Towards automatically identifying, modeling, and naming visual attributes," in *Proceedings of the ACM International Conference on Multimedia, MM '14, Orlando, FL, USA, November 03 - 07, 2014*, 2014, pp. 187–196.

[43] Y. Chen, W. H. Hsu, and H. M. Liao, "Automatic training image acquisition and effective feature selection from community-contributed photos for facial attribute detection," *IEEE Transactions on Multimedia*, vol. 15, no. 6, pp. 1388–1399, 2013.

[44] C. Lin, Y. Chen, B. Chen, Y. Hou, and W. H. Hsu, "Facial attribute space compression by latent human topic discovery," in *Proceedings of the ACM International Conference on Multimedia, MM '14, Orlando, FL, USA, November 03 - 07, 2014*, 2014, pp. 1157–1160.

[45] A. Kovashka, D. Parikh, and K. Grauman, "Whittlesearch: Image Search with Relative Attribute Feedback," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2012.

[46] G. E. Hinton, S. Osindero, and Y. W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[47] R. Salakhutdinov and G. E. Hinton, "Deep boltzmann machines," in *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2009, pp. 448–455.

[48] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," in *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS)*, 2006, pp. 153–160.

[49] P. Vincent, H. Larochelle, Y. Bengio, and P. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2008, pp. 1096–1103.

[50] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, November 1998.

[51] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014, pp. 818–833.

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

[53] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *CoRR*, vol. abs/1409.4842, 2014.

[54] X. Yang, T. Zhang, C. Xu, and M. Yang, "Boosted multifeature learning for cross-domain transfer," *TOMCCAP*, vol. 11, no. 3, pp. 35:1–35:18, 2015.

[55] S. Ji, W. Xu, M. Yang, and K. Yu, "3d convolutional neural networks for human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221–231, 2013.

[56] Y. Sun, X. Wang, and X. Tang, "Deep convolutional network cascade for facial point detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 3476–3483.

[57] X. Yang, T. Zhang, and C. Xu, "Cross-domain feature learning in multimedia," *IEEE Trans. Multimedia*, vol. 17, no. 1, pp. 64–78, 2015.

[58] J. Hu, J. Lu, and Y. Tan, "Discriminative deep metric learning for face verification in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 1875–1882.

[59] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, 2014, pp. 1701–1708.

[60] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu, "Learning fine-grained image similarity with deep ranking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 1386–1393.

Xiaoshan Yang received the master's degree in computer science from Beijing Institute of Technology, Beijing, China, in 2012. He is currently pursuing the Ph.D. degree at the Multimedia Computing Group, National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China. He was an intern in China-Singapore Institute of Digital Media, Singapore, from Sept. 2013 to April. 2014. His research interests include multimedia analysis and computer vision.

Tianzhu Zhang (M'11) received the bachelor's degree in communications and information technology from Beijing Institute of Technology, Beijing, China, in 2006, and the Ph.D. degree in pattern recognition and intelligent systems from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2011. Currently, he is an Associate Professor at the Institute of Automation, Chinese Academy of Sciences. His current research interests include computer vision and multimedia, especially action recognition, object classification and object tracking.

Changsheng Xu (M'97–SM'99–F'14) is a Professor in National Lab of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences and Executive Director of China-Singapore Institute of Digital Media. His research interests include multimedia content analysis/indexing/retrieval, pattern recognition and computer vision. He has hold 30 granted/pending patents and published over 200 refereed research papers in these areas. Dr. Xu is an Associate Editor of IEEE Trans. on Multimedia, ACM Trans. on Multimedia Computing, Communications and Applications and ACM/Springer Multimedia Systems Journal. He received the Best Associate Editor Award of ACM Trans. on Multimedia Computing, Communications and Applications in 2012 and the Best Editorial Member Award of ACM/Springer Multimedia Systems Journal in 2008. He served as Program Chair of ACM Multimedia 2009. He has served as associate editor, guest editor, general chair, program chair, area/track chair, special session organizer, session chair and TPC member for over 20 IEEE and ACM prestigious multimedia journals, conferences and workshops. He is IEEE Fellow, IAPR Fellow and ACM Distinguished Scientist.

Shuicheng Yan (M'06–SM'09) is currently a (Dean's Chair) Associate Professor with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore. He has authored or coauthored over 370 technical papers over a wide range of research topics, with Google Scholar citation $>$ 21000 times and H-index-61. His research interests include computer vision, multimedia, and machine learning. Dr. Yan was or will be General or Program Co-Chair of MMM'13, PCM'13, ACM MM'15, ICMR'17, and ACM MM'17. He was the recipient of the Best Paper Award from ACM MM'13 (Best Paper and Best Student Paper), ACM MM'12 (demo), PCM'11, ACM MM'10, ICME'10, and ICIMCS'09, the winning prize of the classification task in PASCAL VOC 2010-2012, the winning prize of the segmentation task in PASCAL VOC'12, the Honorable Mention prize of the detection task in PASCAL VOC'10, 2010 TCSVT Best Associate Editor (BAE) Award, 2010 Young Faculty Research Award, 2011 Singapore Young Scientist Award, and 2012 NUS Young Researcher Award. He was the coauthor of the papers that was the recipient of the Best Student Paper Award of PREMIA'09, PREMIA'11, PREMIA'12, PREMIA'14, and PREMIA'15.

M. Shamim Hossain (Graduate Student M'02–SM'09) is an Associate Professor with the Department of Software Engineering, King Saud University, Riyadh, KSA. Dr. Shamim Hossain received his Ph.D. in Electrical and Computer Engineering from the University of Ottawa, Canada, in 2009. His research interests include Internet of things (IoT), cloud and multimedia for healthcare, Multimedia big data, social media, serious games, and biologically inspired approach for multimedia and software system. He has authored and co-authored around 100 publications including refereed IEEE/ACM/Springer/Elsevier journals, conference papers, books, and book chapters. He has served as a member of the organizing and technical committees of several international conferences and workshops. He was the recipient of the Outstanding Paper award from an IEEE Conference, and "Research in Excellence" award from King Saud University. He has served as co-chair, general chair, workshop chair, publication chair, publicity chair, and TPC for over 12 IEEE and ACM conferences and workshops. He is on the editorial board of IEEE Access, and International Journal of Multimedia Tools and Applications (Springer). He serves/served as a Lead Guest Editor of IEEE Transactions on Cloud Computing, IEEE Transactions on Information Technology in Biomedicine (Currently, IEEE J-BHI), IEEE Communication Magazine, Future Generation Computer Systems (Elsevier), Computers & Electrical Engineering (Elsevier), International Journal of Multimedia Tools and Applications (Springer), Cluster Computing (Springer), SENSORS (MDPI) and International Journal of Distributed Sensor Networks (Hindawi). Dr. Shamim is a Senior Member of IEEE, a member of ACM SIGMM.

Ahmed Ghoneim (M'10) received the M.Sc. degree in software modeling from the University of Menoufia, Shebeen El-Kom, Egypt, in 1999, and the Ph.D. degree in software engineering from the University of Magdeburg, Magdeburg, Germany, in 2007. He is currently an Assistant Professor with the Department of Software Engineering, King Saud University, Riyadh, Saudi Arabia. His research activities include software evolution, service oriented engineering, software development methodologies, quality of services, net-centric computing, and human-computer interaction.