Neurocomputing 178 (2016) 36–45

Contents lists available at ScienceDirect

Neurocomputing

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

Deep Boosting: Joint feature selection and analysis dictionary learning in hierarchy

Zhanglin Peng [a], Ya Li [a], Zhaoquan Cai [b], Liang Lin [a,*]

[a] Sun Yat-sen University, Guangzhou, China
[b] Huizhou University, Huizhou, China

ARTICLE INFO

Article history:
Received 13 February 2015
Received in revised form
15 July 2015
Accepted 16 July 2015
Available online 6 November 2015

Keywords:
Representation Learning
Compositional boosting
Dictionary learning
Image Classification

ABSTRACT

This work investigates how the traditional image classification pipelines can be extended into a deep architecture, inspired by recent successes of deep neural networks. We propose a deep boosting framework based on layer-by-layer joint feature boosting and dictionary learning. In each layer, we construct a dictionary of filters by combining the filters from the lower layer, and iteratively optimize the image representation with a joint discriminative-generative formulation, i.e. minimization of empirical classification error plus regularization of analysis image generation over training images. For optimization, we perform two iterating steps: (i) to minimize the classification error, select the most discriminative features using the gentle adaboost algorithm; (ii) according to the feature selection, update the filters to minimize the regularization on analysis image representation using the gradient descent method. Once the optimization is converged, we learn the higher layer representation in the same way. Our model delivers several distinct advantages. First, our layer-wise optimization provides the potential to build very deep architectures. Second, the generated image representation is compact and meaningful by jointly considering image classification and generation. In several visual recognition tasks, our framework outperforms existing state-of-the-art approaches.

1. Introduction

Visual recognition is one of the most challenging domains in the field of computer vision and smart computing. Many complex image and video understanding systems employ visual recognition as the basic component for further analysis. Thus the design of robust visual recognition algorithm is becoming a fundamental engineering in computer vision literature and has been attracting many related researchers. Since the inadequate visual representation will greatly influence the performance of visual recognition system, almost all of the related methods are concentrated on developing the effective visual representation.

Traditional visual recognition systems always adopt the shallow model to construct the image/video representation. Among them, the *bag-of-visual-words* (BoW) model, which is the most successful one for visual content representation, has been widely adopted in many computer vision tasks, such as object recognition [1,2] and image classification [3,4]. The basic pipeline of BoW model consists of local feature extraction [5,6], feature encoding [7–9] and pooling operation. In order to improve the performance of BoW, two crucial schemes have been involved. First, the traditional BoW model discards the spatial information of local descriptors, which seriously limited the descriptive power of the feature representation. To overcome this problem, the Spatial Pyramid Matching method was proposed in [3] to capture geometrical relationships among local features. Second, dictionaries adopted to encode the local feature in traditional methods are learned in a unsupervised manner and can hardly capture the discriminative visual pattern for each category. This issue inspired a series of works [10–12] to train more discriminative dictionaries via supervised learning, which can be implemented by introducing the discriminative term into dictionary learning phase as the regularization according to various criteria.

As the research going, the deep models, which can be seen as a type of hierarchical representation [13–15] have played an significant role in computer vision and machine learning literature [16–18] in recent years. Generally, such hierarchical architecture represents different layer of vision primitives such as pixels, edges, object parts and so on [19]. The basic principles of such deep models are concentrated on two folds: (1) layerwise learning philosophy, whose goal is to learn single layer of the model individually and stack them to form the final architecture; (2) feature combination rules, which aim at utilizing the combination (linear

* Corresponding author.
E-mail address: linliang@ieee.org (L. Lin).

http://dx.doi.org/10.1016/j.neucom.2015.07.116
0925-2312/

or nonlinear) of low layer detected features to construct the high layer impressive features by introducing the activation function.

In this paper, the related exciting researches inspire us to explore how the traditional image classification pipelines, which include feature encoding, spatial pyramid representation and salient pattern extraction (*e.g.*, max spatial pooling operation), can be extended into a deep architecture. To this end, this paper proposes a novel deep boosting framework, which aims to construct the effective discriminative features for image classification task, jointly adopting feature boosting and dictionary learning. For each layer, followed the famous boosting principle [20], our proposed method sequentially selects the discriminative visual features to learn the strong classifier by minimizing empirical classification error. On the other hand, the analysis dictionary learning strategy is involved to make the selected features more suitable for the object category. A two-step learning process is investigated to iteratively optimize the objective function. In order to construct high-level discriminative representations, we composite the learned filters corresponding to selected features in the same layer, and feed the compositional results into next layer to build the higher-layer analysis dictionary. Another key to our approach is introducing the model compression strategy when constructing the analysis dictionary, that reduces the complexity of the feature space and shortens the model training time. The experiment shows that our method achieves excellent performance on general object recognition tasks. Fig. 1 illustrates the pipeline of our deep boosting method (applying two layers as the illustration). Compared with the traditional BoW based method [7], the analysis operation in our model (*i.e.*, convolution) is same as the encoding process that maps the image into the feature space. While the pooling stage is same as the traditional method to compute the histogram representation adopting spatial pyramid matching. Different from traditional models capturing the salient properties of visual patterns by max spatial pooling operation, we adopt the feature boosting to the discriminative features mining for image representation.

The main contributions of this paper are three folds. (1) A novel deep boosting framework is proposed and it leverages the generative and discriminative feature representation. (2) It presents a novel formulation which jointly adopting feature boosting and analysis dictionary learning for image representation. (3) In the experiment on several standard benchmarks, it shows that the learned image representation well discovers the discriminative features and achieves the good performance on various object recognition tasks.

The rest of the paper is organized as follows. Section 2 presents a brief review of related work, followed by the overview of background technique details in Section 3. Then we introduce our deep boosting framework in Section 4. Section 5 gives the experimental results and comparisons. Section 6 concludes the paper.

2. Related work

In the past few decades, many works have been done to design different kinds of features to express the characteristics of the image for further visual tasks. These hand-craft features vary from global expressions [21] to the local representation [5]. Such designed features can be roughly divided into two types [22], the one is geometric features and the other is texture features. Geometric features which explicitly record the locations of edges are employed to describe the noticeable structures of local areas. Such features include Canny edge descriptor [23], Gabor-like primitives [24] and shape context descriptor [25,26]. In contrast, the texture features express the cluttered object appearance by histogram statistics. SIFT [5], HoG [6] and GIST [27] are delegates of such feature representation. Beyond such hand-craft feature descriptors, Bag-of-Feature (BoF) model seems to be the most classical image representation method in computer vision area. A lot of illuminating studies [4,3,7,8] were published to improve this traditional approach in different aspects. Among these extensions, a class of sparse coding based methods [7,8], which employ spatial pyramid matching kernel (SPM) proposed by Lazebnik et al., has achieved great success in image classification problem. However, despite we are developing more and more effective representation methods, the lack of high-level image expression still plagues us to build up the ideal vision system.

On the other hand, learning hierarchical models to simultaneously construct multiple levels of visual representation has been paid much attention recently. The proposed hierarchical image representation is partially motivated by recent developed deep learning approaches [13,14,28]. Different from previous hand-craft feature design method, deep model learns the feature representation from raw data and validly generates the high-level semantic representation. And such abstract semantic representations are expected to provide more intra-class variability. Recently, many vision tasks achieve significant improvement using the convolutional architectures [16–18]. A deep convolutional architecture consists of multiple stacked individual layers, followed by an empirical loss layer. Among all of these layers, the convolutional layer, the feature pooling layer and the full connection layer play major roles in abstract feature representation. The stochastic gradient descent algorithm is always applied to the parameters

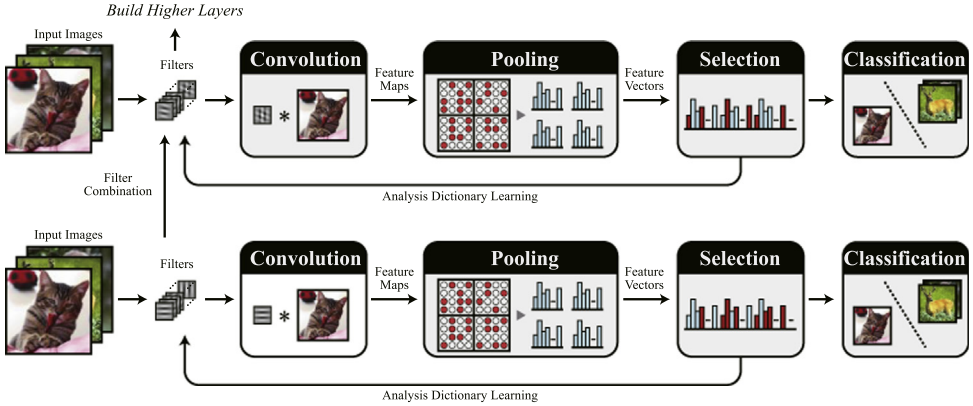


Fig. 1. A two-layer illustration of proposed deep boosting framework. The horizontal pipelines show the layer-wised image representation via joint feature boosting and analysis dictionary learning. When optimization in the single layer is done, the compositional filters are fed into the higher-layer to generate the novel analysis dictionary for further processing. Note that the feature set in the higher-layer only dependents on the training images and combined filters in the relevant layer.

training in each layers according to back-propagation principle. However, as shown in recent study [28], these network-based hierarchical models always contain thousands of parameters. Learning a useful network usually depends on expertise of parameter tuning (*e.g.*, tuning the learning rate and parameter decay rate in each layer) and is too complex to control in real visual application. In contrast, we build up our hierarchical image representation according to the simple but effective rules. Our method can also achieve the near optimal classification rate in each layer.

Another related work to this paper is learning a dictionary in an analysis prior [29–31]. The key idea of analysis-based model is utilizing analysis operator (also known as analysis dictionary) to deal with latent clean signal and leading to a sparse outcome. In this paper, we consider the analysis-based prior as a regularization prior to learn more discriminative features to a certain category. Please refer to Section 3 for more details about analysis dictionary learning.

3. Background overview

3.1. Gentle Adaboost

We start with a brief review of Gentle Adaboost algorithm [20]. Without loss of generality, considering the two-class classification problem, let $(x_1, y_1)\ldots(x_N, y_N)$ be the training samples, where x_i is a feature representation of the sample and $y_i \in \{-1, 1\}$. w_i is the sample weight related to x_i. Gentle Adaboost [20,32] provides a simple additive model with the form,

$$F(x_i) = \sum_{m=1}^{M} f_m(x_i), \tag{1}$$

where f_m is called weak classifier in the machine learning literature. It often defines f_m as the regression stump $f_m(x_i) = a\hbar(x_i^d > \delta) + b$, $\hbar(\cdot)$ denotes the indicator function which returns 1 when $x_i^d > \delta$ and 0 otherwise, x_i^d is the d-th dimension of the feature vector x_i, δ is a threshold, a and b are two parameters contributing to the linear regression function. In iteration m, the algorithm learns the parameter (d, δ, a, b) of $f_m(\cdot)$ by weighted least-squares of y_i to x_i with weight w_i,

$$\min_{1 \le d \le D} \sum_{i=1}^{N} w_i \| a^d \hbar(x_i^d > \delta^d) + b^d - y_i \|^2, \tag{2}$$

where D is the dimension of the feature space. In order to give much attention to the cases that are misclassified in each round, Gentle Adaboost adjusts the sample weight in the next iteration as $w_i \leftarrow w_i e^{-y_i f_m(x_i)}$ and updates $F(x_i) \leftarrow F(x_i) + f_m(x_i)$. At last, the algorithm outputs the result of strong classifier as the form of sign function $sign[F(x_i)]$. In this paper, we adopt Gentle Adaboost as the basic component of proposed model. Please refer to [20,32] for more technique details.

3.2. Analysis dictionary learning

Our work is also inspired by the recent developed analysis-based sparse representation prior learning [29–31], which represents the input signal from a dual viewpoint of the commonly used synthesis model [33]. The main idea of analysis prior learning is to learn the analysis operators (*e.g.*, convolution operator) that can return the special responses (*e.g.*, sparse response as usual) from the latent signal according to the given constraint. Let $\widehat{I}$ be the observed signal (*e.g.*, natural image) with noisy which is often assumed as zero-mean white Gaussian. An analysis-based prior seeks the latent signal I whose analysis transform result is sparse,

$$\min_{I,G} \frac{1}{2} \| \widehat{I} - I \|_2^2 + \psi \Phi(G * I), \tag{3}$$

where $\psi \ge 0$ is a scalar constant and the symbol $*$ indicates the analysis operation. The first term denotes the reconstruction error and the second one denotes the sparsity constraint of the forward transform coefficient. G is usually a redundant dictionary employing as the analysis operator. In different context, such analysis prior G is more frequently adopted to enforce some regularity on the signal. In this paper, we utilize the philosophy of analysis-based prior to seek the discriminative filters for image feature representation. Please refer to [29–31] for more technique details and theoretical analysis.

4. Problem formulation

Considering the two-class classification problem, for given training data and its corresponding label $\{(x_i, y_i) | i \in \{1, \ldots, N\}\}$, $y_i \in \{-1, 1\}$. In order to construct the rich and discriminative image representation for each category, we propose a deep boosting framework based on compositional feature selection and analysis dictionary learning. For a single layer, we firstly introduce the term of empirical error to the discriminative features mining. This is equal to learn the weak classifier in Gentle Adaboost algorithm. For each category, suppose that if we can find an analysis dictionary, denoted by $G \in R^{p \times M}$, that the selected feature can be more suitable for such category by the analysis transformation, then the feature representation would be more effective for visual recognition. Based on this idea, the fundamental of our single layer image representation is expressed as follows:

$$\min_{G} \frac{1}{2} \sum_{i=1}^{N} l(-y_i F(x_i))) + \lambda \sum_{I_j \notin \Omega} \| G * I_j \|_2^2, \tag{4}$$

where x_i is the feature representation corresponding to image I_i and $l(\cdot)$ denotes the empirical error of the classifier. Ω indicates positive training set and $I_j \notin \Omega$ means that the image I_j does not belong to the set of positive samples. We define $G = [g_1, g_2, \ldots g_m \ldots, g_M]$ as the analysis dictionary and each g_m indicates a linear filter. Thus $G * I$ can be considered as a series of convolutional operations and the output is M feature maps, each of which is related to a special linear filter. The properties of our proposed model are two folds. On one hand, different from traditional analysis prior learning, we adopt the empirical error, which is more suitable for training the classifier, to replace the reconstruction error in Eq. (2). On the other hand, the analysis operator is introduced as the regularized term to learn more discriminative features for each category. In the second term of Eq. (4), we desire the analysis dictionary (*i.e.*, a set of filters) has large filter response over the positive training set. In this way, the analysis dictionary learning process could discover category coherent features (*i.e.*, one category one analysis dictionary) to promote the discriminative ability of weak classifiers. It is equivalent to make the analysis dictionary has the small response over negative samples, thus we extract negative training samples and minimize the objective function to train the analysis dictionary. Note that, if the learned filter has the small response to both the positive and negative samples, the related feature representation will be eliminated in the further iteration of feature selection process. In this way, the discriminative of our image representation is enhanced by joint feature boosting and analysis dictionary learning, leading the model more robust and compact as well.

In Eq. (4), x_i is the feature vector of i-th image associated with the analysis transformation (*i.e.*, filter response or convolution

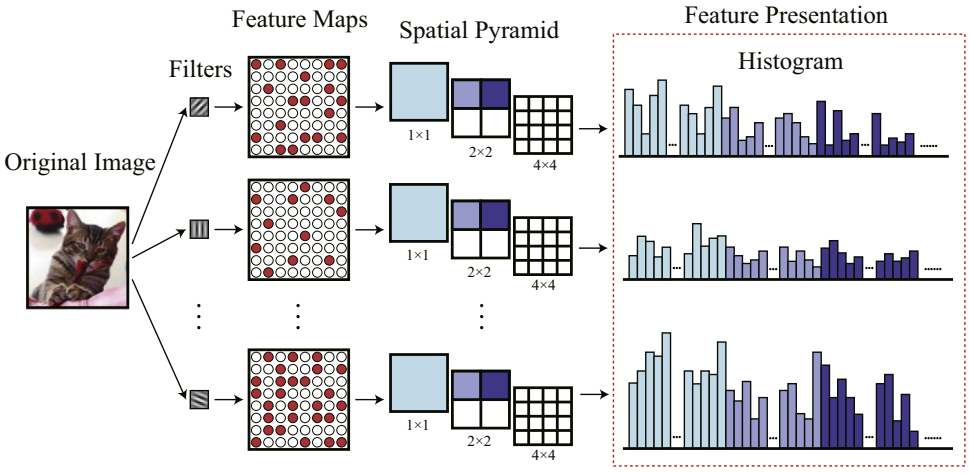


Fig. 2. Toy example of constructing a three-level pyramid histogram as the image feature representation.

result). In order to obtain such feature representation, we employ the pyramid-wise histograms to quantize the filter responses, which provide some degree of translation invariance for the extracted features, as in hand-crafted features (*e.g.*, SIFT or HoG), learned features (*e.g.*, Bag-of-Visual-Words model), and average or maximum pooling process in convolution neural network. Suppose M is the total number of filters. Before construct the pyramid-wise histograms for a special image I, we firstly activate the maximum filter responses of each pixel and abandon the others as follows:

$$u_m = \begin{cases} \|u_m\| & \text{if } \|u_m\| = \max\{\|u_1\|, \|u_2\|, \ldots, \|u_M\|\} \\ 0 & \text{otherwise} \end{cases}, \tag{5}$$

where u_m indicates the m-th filter response of pixel $u \in I$.

According to the previous operation, we can obtain M feature maps for a training image, each of which has only a few locations being activated according to Eq. (5) (presented by red solid circle in Fig. 2). As shown in Fig. 2, we apply a three-level spatial pyramid representation of each resulting feature map, resulting $1+2\times 2+4\times 4=21$ individual spatial blocks. We compute the histogram (with C bins, $C=50$ in the rest of the paper) of the filter responses in each block. Finally, we can get the "long" feature vector formed by concatenating the histograms of all blocks from all feature maps. The dimension of such feature vector is $21\times 50\times M$. Note that M is not a constant scalar in this paper, and the value could be dynamically changed with the process of analysis dictionary learning. Please refer to Section 4.2 for more details.

4.1. Feature boosting

In order to optimize the objective function in Eq. (4), we propose a two-step optimizing strategy integrating the feature boosting and dictionary learning. In this subsection, we describe the details of feature boosting method by setting up the relationship between the weak classifier and the image feature representation. After the pyramid-wise histogram calculated, we select the discriminative features and obtain the single layer classifier through the given feature set. Follow the previous notation, let $x_i \in R^D$ be the feature representation of image I_i, where D is the dimension of the feature space and $D=21\times 50\times M$ as described in the previous content. In the feature boosting phase, Gentle Adaboost is applied to the discriminative features (*i.e.*, weak classifiers) mining, which can separate the positive and negative samples nicely in each round. Note that in the rest of the paper, we apply x_i^d to denote the value of x_i in the d-th dimension. In each round of feature boosting procedure, the algorithm retrieves all of the candidate regression functions $\{f^1, f^2, \ldots, f^D\}$, each of which is formulated as:

$$f^d(x_i) = a\phi(x_i^d - \delta) + b, \tag{6}$$

where $\phi(\cdot)$ is the sigmoid function with the form $\phi(x)=1/(1+e^{-x})$. For each round, the candidate function with minimum empirical error is selected as the current weak classifier f, such that

$$\min_d \sum_{i=1}^{N} w_i \|f^d(x_i) - y_i\|^2, \tag{7}$$

where $f^d(x_i)$ is associated with the d-th element of x_i and the function parameter (δ, a, b). According to the above discussion, we build the bridge between the weak classifier and the feature representation, thus the weak classifiers learning can be viewed as the feature boosting procedure in our model. The feature boosting is usually terminated when the training error is converged.

4.2. Analysis dictionary learning

To the regularization perspective, another advantage of method is introducing analysis dictionary learning, which is conducted by selected features in the feature boosting phase, to emphasize the discriminative ability of analysis operator for the target category. In our framework, since we rely on discriminative filters to generate higher-layer proper analysis dictionary, we only consider to update a subset of filters which is corresponding to the selected features. We first need to construct the relationship between feature responses and filters. For any feature response, a four-item index is recorded as,

$$[isActivited, w, h, g], \tag{8}$$

where *isActivited* indicates whether the feature response is selected in feature boosting stage. w,h are the horizontal and vertical coordinate in the image lattice domain respectively. g denotes the relative filter defined in Eq. (4). Then we apply the gradient descent algorithm to optimize filters which is corresponding to selected features. As Fig. 1 illustrates, we combine any two optimized filters but not the features to generate filters in the next layer. In this way, the filter's optimization in the next layer is independent with previous features. Note that in the first few layers, the number of filters is limited, thus almost every filter is taken into account in optimization. However, it will be shown in Section 4.3 that the collection of compositional filters becomes large along with the architecture going deep, thus the screening

mechanism is introduced to control the complexity and keep the effectiveness of the model.

Integrating the two stages described in Sections 4.1 and 4.2, we achieve the feature boosting and analysis dictionary learning for the single layer. The algorithm is summarized in Algorithm 1. In the next subsection we will introduce the filter combination rules to construct the hierarchical architecture of our model.

4.3. Deep boosting framework

In the context of boosting method, the strong classifier, which is usually the weighted linear combination of weak classifiers, is hardly to decease the test error when training error is approaching to zero. Based on this fact, it is our interest to learn high-level feature representations with more discriminative ability. In order to achieve this goal, we propose the filter combination rules and the output compositional filters of each layer are treated as a whole to generate the analysis dictionary in the next layer.

For each image category, whose corresponding analysis dictionary in layer l is denoted by $[G]_l$, we combine any two optimized filters (presented by solid circle in Fig. 3(a)) in the l-th layer as follows,

$$[g_k]_{l+1} = \phi([g_i]_l + [g_j]_l), \tag{9}$$

where $\phi(\cdot)$ is the sigmoid function. $[g_i]_l$ and $[g_j]_l$ indicate the i-th and j-th filters in the optimized subset of $[G]_l$. As illustrated in Fig. 3(a), the number of filters in each layer is quite different and we only adopt the optimized ones, which are related to selected features, to construct the image filters for the next layer.

4.4. Model compression approach

Although we carefully select filters for further combination, the number of compositional filters will still be out of control when architecture going deep. Assuming there exists M_l optimized filters in layer l, thus we can obtain he maximum number $\frac{1}{2} \times M_l \times (M_l - 1)$ of compositional filters. In this way, the dimension of each image in the layer $l+1$ would be $\frac{1}{2} \times M_l \times (M_l - 1) \times 21 \times 50$, which make the feature space is too complex and the training time becomes intolerable. To this end, we introduce model compression in the training phase. For any couple of filers, the L2 distance is calculated to measure the similarity between them. If the distance is smaller than the threshold δ (set as 0.7 in all the experiment), we maintain the two filters are similar and one of them is dropped out randomly (presented by hollow circle in Fig. 3(a)). Fig. 3(b) and (c) illustrate the similarity matrix of filters in different layer. The intensity of every square indicates the similar degree of two filters. Please refer to Figs. 6 and 7 for more details about the classification accuracy and training time comparison with and without model compression for different depth of proposed framework.

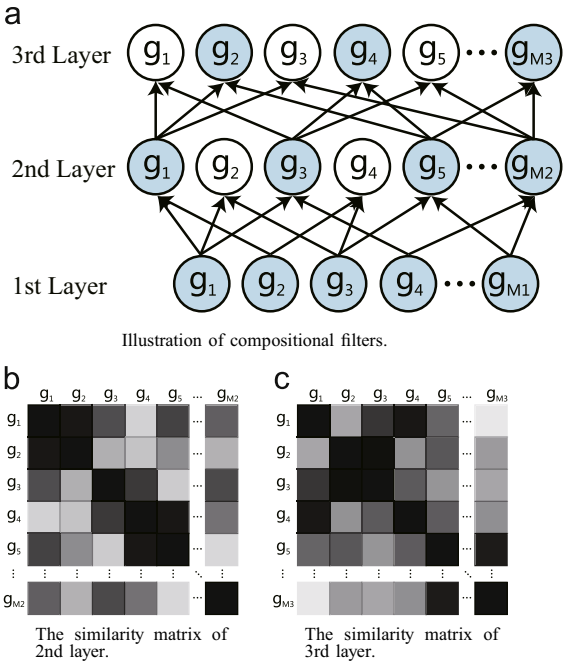


Fig. 3. Illustration of compositional filters for deep boosting. We composite filters in a pairwise manners in each layer and treat the output compositional filters as base filters (presented by solid circle in Fig. 3(a)) in next layer. After combination, the similar matrix of filters is built up to drop out redundancies (presented by hollow circle in Fig. 3(a)). (a) Illustration of compositional filters. (b) The similarity matrix of 2nd layer. (c) The similarity matrix of 3rd layer.

According to Section 4.3, we build up the hierarchical architecture of our deep boosting framework. In the testing phase, we employ the weak classifiers learned in every layer to produce the final classifier. The overall of our proposed method is summarized in Algorithm 2.

Algorithm 1. Joint Feature Boosting and Analysis Dictionary Learning.

Input:
Positive and negative training samples $(x_1, y_1) \ldots (x_N, y_N)$, the number of selected features Π.
Output:
A pool of selected features Ψ, the learned dictionary G.
Initialization:
The dictionary G;
Repeat:
1. Start with score $F(x) = 0$ and sample weights $w_i = 1/N$, $i = 1, 2, \ldots, N$.
2. Select features and learn the strong classifier as follows:
Repeat for $m = 1, 2, \ldots, \Pi$:
(a) Learn the current weak classifier f_m by Eq. (6).
(b) Update $w_i \leftarrow w_i e^{-y_i f_m(x)}$ and renormalize.
(c) Update $F(x) \leftarrow F(x) + f_m(x)$.
3. Update the dictionary G by gradient descent method.
4. Generate new feature vectors of each image using G according to Section 4.
until The objective function in Eq. (4) converges.

Algorithm 2. Deep Boosting Framework.

Input:
Positive and negative training images and corresponding labels $(I_1, y_1) \ldots (I_N, y_N)$, the number of selected features Π_l in layer l, the total layer number L.
Output:
The final classifier $F^L(x)$ for a special category.
Initialization:
Initialize G' in first layer applying Gabor wavelets.
Repeat for $l = 1, 2, \ldots, L$:
1. Generate new feature x of image I using G according to Section 4.
2. Boost features with dictionary learning according to Algorithm 1.
3. Build up filters of next layers according to Eq. (9).

4.5. Preprocessing and multi-class decision

At the beginning, we initialize the filters with the size of 5×5 adopting Gabor wavelets. Let I be an image defined on image

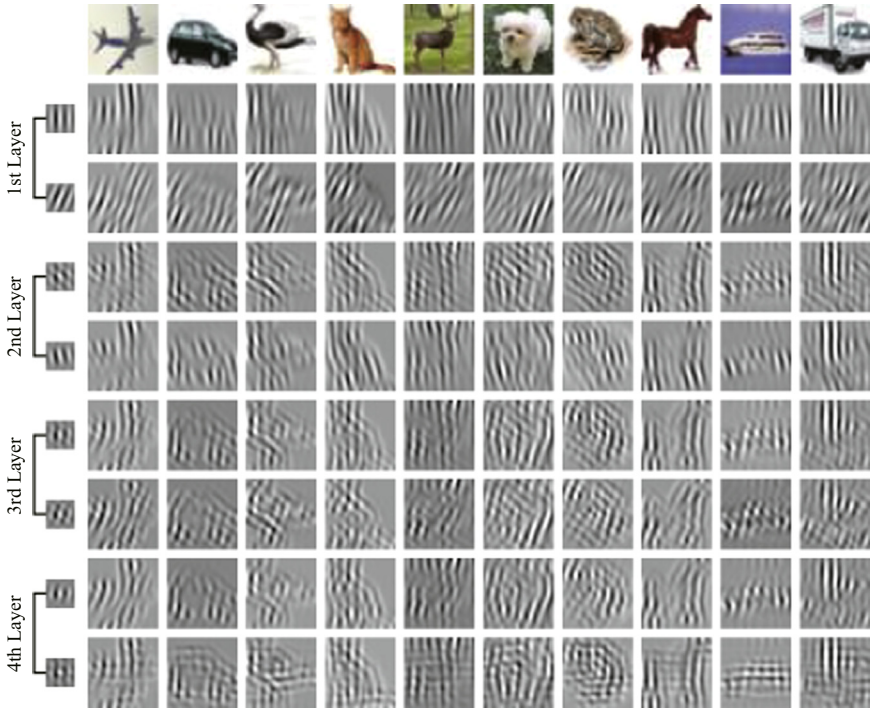


Fig. 4. The learned templates in the first four layers for each image categories. When the model goes deeper, we get higher level primitives and the more discriminative features.

lattice domain and G' be the Gabor wavelet elements with parameters (w,h,α,s), where (w,h) is the central position belonging to the lattice domain, α and s denote the orientation and scale parameters. Different orientation and scale parameters makes Gabor wavelets variant. For simplicity, we apply 1 scale and 16 orientations in our implementation, so there are total 16 filters at first layer. Notably, multi scales promote the performance while the filter combination process becomes complicated, because the combination is only allowed in the same scale. Followed by [34], we utilize the normalize term to make the Gabor responses comparable during the inception phase between different training images:

$$\delta^2(s)=\frac{1}{|P|A}\sum_{\alpha}\sum_{w,h}|\langle I,G'_{w,h,\alpha,s}\rangle|^2, \tag{10}$$

where $|P|$ is the total number of pixels in image I, and A is the number of orientations. $\langle\cdot\rangle$ denotes the convolution process. For each image I, we normalize the local energy as $|\langle I,G'_{w,h,\alpha,s}\rangle|^2/\delta^2(s)$ and define positive square root of such normalized result as feature response.

To the multiclass situation, we consider the naive *one-vs-all* scheme to train multiple binary classifiers, each one learns to distinguish the samples in a single class from the samples in all remaining classes. Given the training data $\{(x_i,y_i)\}_{i=1}^N, y_i\in\{1,2,\ldots,K\}$, we train K strong classifiers, each of which returns a classification score for a special test image. In the testing phase, we predict the label of image referring to the classifier with the maximum score. The reason why we adopt *one-vs-all* or OVA scheme throughout the paper is concentrated on two folds. On one hand, according to Eq. (4), we desire each learned analysis dictionary should have powerful capability to distinguish the images from one category. Thus we select the negative samples from all other categories to optimize the filters in Eq. (4) (*i.e.*, leaning the class-specific analysis dictionary) and this strategy is naturally consistent with the OVA scheme. On the other hand, as shown in [35], many multiclass models may not offer advantages over the simple OVA scheme in the solution of classification problem. Under such circumstances, we finally choose the OVA strategy followed by its intuitive concept.

Table 1
Classification accuracy on STL-10.

Method	Accuracy ($\pm\sigma$)
1-layer Vector Quantization [36]	54.9% ($\pm$0.4%)
1-layer Sparse Coding [36]	59.0% ($\pm$0.8%)
3-layer Learned Receptive Field [37]	60.1% ($\pm$1.0%)
OURS-5	59.3% ($\pm$0.8%)

Table 2
Classification accuracy on STL-10 dataset with and without regularized term.

	Accuracy ($\pm\sigma$)
With regularized term	59.3% ($\pm$0.8%)
Without regularized term	55.8% ($\pm$1.5%)

5. Experiment

We conduct several experiments to investigate the properties of proposed deep boosting framework and evaluate the performance for different challenging visual recognition tasks (*i.e.*, facial age estimation, natural image classification and similar appearance categories recognition). All of the experiments are carried out on a PC with Core i7-3960X 3.30 GHZ CPU and 24 GB memory. In these tasks, we demonstrate superior or comparable performances of our framework over other state-of-the-art approaches.

5.1. Learning image template for image categories

In the first experiment we focus on whether our algorithm can learn and select meaningful and discriminative features for

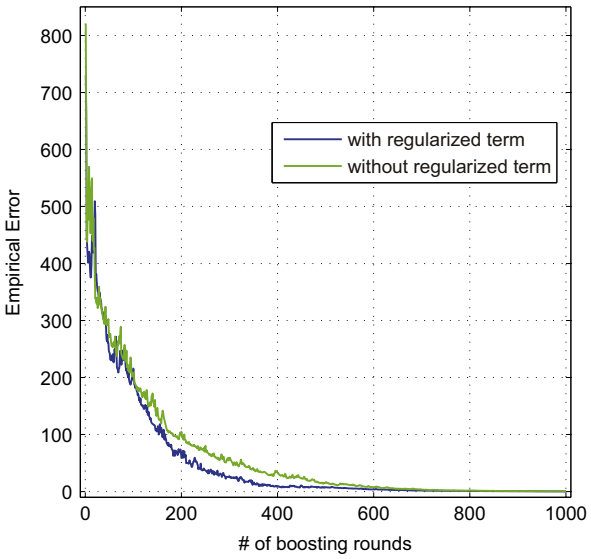


Fig. 5. The empirical error at boosting rounds. The method with regularized term has better convergence rate.

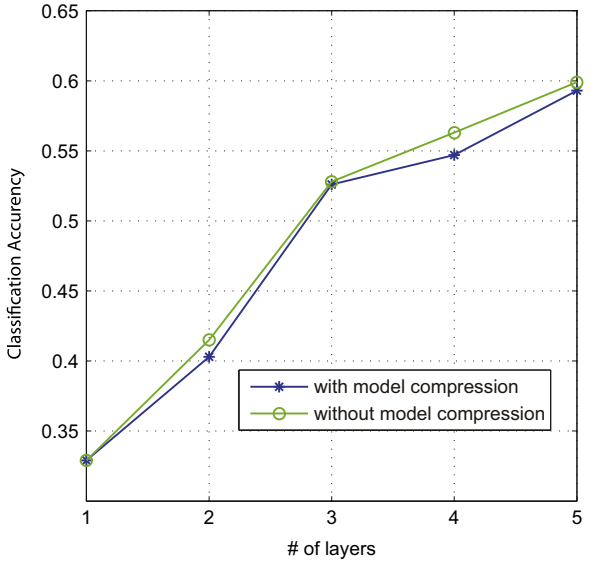


Fig. 6. Classification accuracy at different layers. The method conduct better performance with the growth of model.

different image categories. Take CIFAR-10 dataset, for example. The CIFAR-10 dataset[1] consists of 60 K 32×32 color images in 10 classes (with 6 K images per class), including airplane, automobile, bird, cat, deer, dog, frog, horse, ship and truck. We randomly select 1000 images per class as the training samples to learn the hierarchical image representation. Fig. 4 shows some learned templates in different layers for each image categories. According to the visualizations, it is obviously that the higher layer it goes, the more informative features we gain.

5.2. Natural image classification

The same to CIFAR-10, the STL-10[2] is also a ten-category image dataset, but with the image size 96×96. It has 1300 images per class. There are 500 training images and 800 test images. The training set is mapped to ten predefined folds. Due to its relatively

[1] http://www.cs.toronto.edu/~kriz/cifar.html
[2] http://cs.stanford.edu/~acoates/stl10/

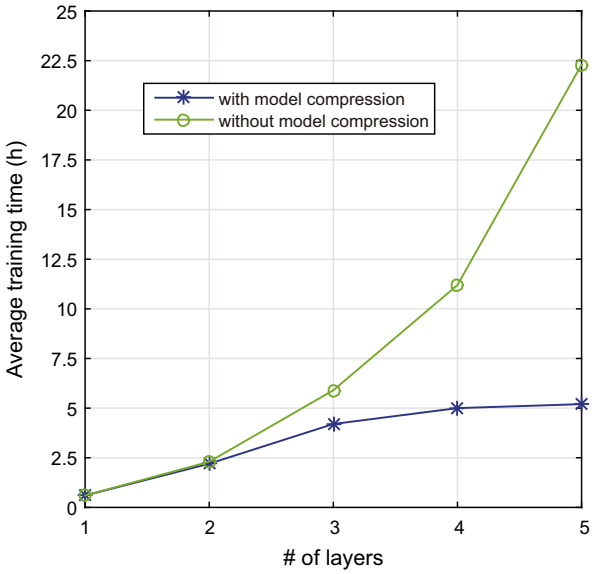


Fig. 7. The average training time of categories at different layers. The average training time of categories greatly reduce when the model is compressed.

large image size, much prior research chose to downsample the images to 32×32. Table 1 shows the comparison of average test accuracies on all folds of STL-10. It is clear that our method can achieve very competitive results compared to other state-of-the-art methods.

5.2.1. Impact of analysis dictionary learning

In this section, we are interested in the performance of our method in the context of analysis dictionary learning. As we mentioned above, the analysis operator is introduced as a regularized term to learn more discriminative features over the positive samples. We desire that the analysis dictionary is able to make the margin between positive and negative training sets as larger as possible. That is, the analysis dictionary has large response over the positive training set, but not vice versa. Note that, the related feature representation will be eliminated in the further iteration of feature selection process, if the learned filter responds a small value both to the negative set and to the positive set. In this way, we will gain more discriminative features in feature boosting procedure, resulting a more robust and compact image representation model.

Table 2 shows the classification accuracy with and without regularized term. The result using regularized term outperforms the other and the standard deviation among folds is smaller, which illustrates that the feature is more discriminative and the model is more robust. In Fig. 5, the empirical error in boosting phase is shown. For the more discriminative features, it is reasonable to accelerate convergence rate using regularized term.

5.2.2. Impact of model depth and compression

In this experiment, we perform classification experiments on the STL-10 in the context of different number of layers. We learn the deep boosting model to construct multiple levels of visual representation simultaneously. In order to construct high-level discriminative representations, we composite the learned filters corresponding to selected features in the same layer, and feed the compositional results into next layer to build the higher-layer analysis dictionary. Hopefully when the model goes higher, the features are more discriminative. Fig. 6 exhibits the performance of image classification on STL-10 at different layers. The results demonstrate that the features in higher layer conduct better performance. In order to avoid the sudden explosion of filters, we drop out similar filters randomly after pairwise combination of the

Fig. 8. The LHI-Animal-Faces dataset. Three images are shown for each category.

Table 3
Classification accuracy on LHI-Animal-Faces.

Method	Accuracy (%)
HoG+SVM	70.8
HIT [22]	75.6
LSVM [39]	77.6
AOT [38]	79.1
OURS-5	**81.5**

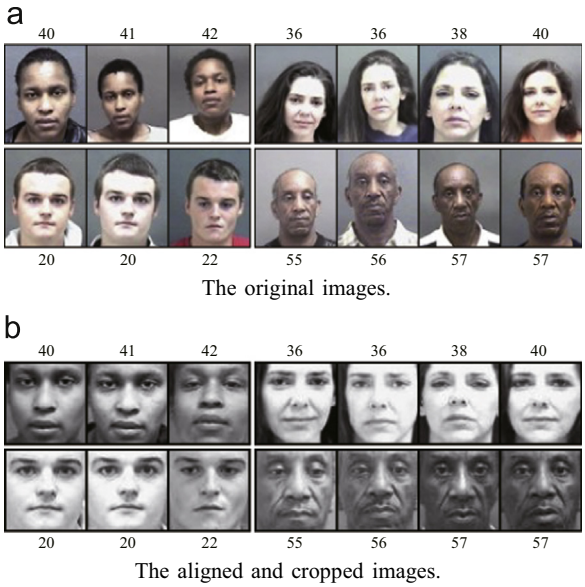


Fig. 9. The MORPH-II dataset. Four individuals in different races and genders are picked as an example. The ages are given around the images. (a) The original images. (b) The aligned and cropped images.

Table 4
MAE (in Years) on MORPH-II (the lower the better).

Method	MAE
MLBP+SVM	6.85
HoG+SVM	6.19
SIFT+SVM	8.77
WAS [42]	9.21
AGES [43]	6.61
IIS-LLD [41]	5.67
OURS-2	**5.61**

learned filters. Although it losses accuracy slightly, we control the training time and make the limitless growth of model possible, which is illustrated in Figs. 6 and Fig. 7.

5.3. Similar appearance categories recognition

The LHI-Animal-Faces dataset[3] [22] consists of about 2200 images for 20 categories. Fig. 8 provides an overview of the

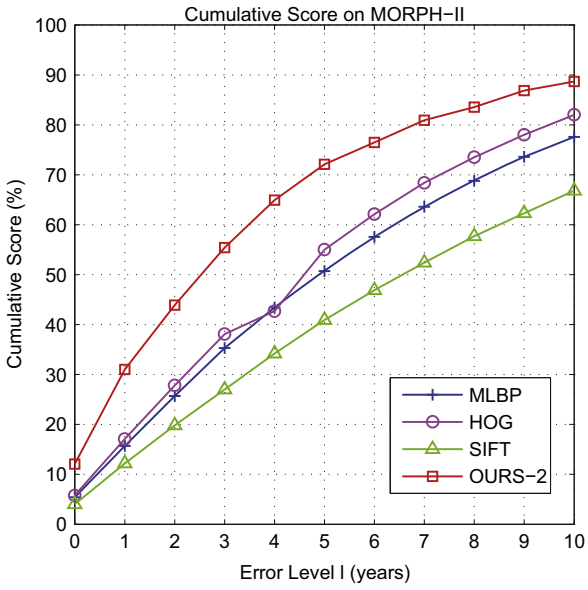


Fig. 10. Cumulative scores at different error levels on MORPH-II.

dataset. In contrast with other general classification datasets, LHI-Animal-Faces contains only animal or human faces, which are similar to each other. It is challenging to discern them for their evolutional relationship and shared parts. Besides, interesting within-class variation is shown in the face categories, including rotation, flip transforms, posture variation and sub-types.

We compare our result with those reported in [38] obtained by other methods, which include HoG feature trained with SVM [6], HIT [22], AOT [38] and partbased HoG feature trained with latent SVM [39]. In the experiment, we split the dataset as training set and test set following AOT [38]. For our method, we resize all the images to the uniform size of 60×60 pixels and the number of layers is 5. Table 3 exhibits the classification accuracy on LHI-Animal-Faces. It has shown that our method achieves a 2.4% increase, compared with the second best competitor.

5.4. Facial age estimation

Human age estimation based on facial images plays an important role in many applications, *e.g.*, intelligent advertisement, security surveillance monitoring and automatic face simulation. To our best knowledge, MORPH-II[4] is the largest publicly available dataset for facial age estimation. In the MORPH-II dataset, there are more than 55,000 facial images from more than 13,000 individuals with only about 4 labeled images per individual. The ages vary over a wide range from 16 to 77. The individuals come from different races, among them Africans accounted for about 77%, the Europeans about 19%, and the remaining includes Hispanic, Asian and other races. Some sample images are shown in Fig. 9(a).

We use two usually performance measures in our comparative study, *i.e.*, MAE (Mean Absolute Error) and CumScore (Cumulative

[3] http://www.stat.ucla.edu/~zzsi/hit/changelog.html

[4] http://www.faceaginggroup.com/morph/

Score) [40]. Suppose there are N test images, the MAE is the sum of average absolute errors between the true ages a_i and the predicted ages $\overline{a}_i$, $i = 1, 2, \ldots, N$. The MAE is calculated as,

$$MAE = \frac{1}{N}\sum_{i}^{N} |a_i - \overline{a}_i|, \tag{11}$$

where $|\cdot|$ denotes the absolute value of a scalar value.

The CumScore is the cumulate accuracy rate. A certain error range (*i.e.*, l years) is acceptable for many real applications. The cumulative score at error level l can be calculated as,

$$CumScore(l) = N_{e \le l}/N \times 100\%, \tag{12}$$

where $N_{e \le l}$ is the number of test images, which have absolute prediction error no more than l years.

For an input image, we locate the face with bounding box and detect the five facial key points in the bounding box. The five facial key points include two eye centers, nose tip, and two mouth corners. Then we align the facial image based on these key points. Finally, the images are resized to the size of 60×60 pixels. The aligned images are shown in Fig. 9(b).

We compare our results with several existing algorithms designed for the age estimation, *i.e.*, IIS-LLD [41], WAS [42] and AGES [43]. Moreover, we also conduct experiments using some feature descriptors usually used in face recognition, including Multi-level LBP [44], HoG [6] and SIFT [5]. For all of these features, age estimation is treated as classification problem using multi-class SVMs. For our method, we set the number of layers to 2 and six-folder cross-validation is performed. Table 4 summarizes the results based on the MAE measure. We can see that our method achieves better results compared to other state-of-the-art methods for age estimation. We also report the results in terms of the cumulative scores at different error levels from 0 to 10 in Fig. 10, exhibiting that our method outperforms other state-of-the-arts at almost all levels.

6. Conclusion

In this paper, we propose a novel deep boosting framework, which is applied to construct the high-level discriminative features for general image recognition task. For each layer, the feature boosting and analysis dictionary learning are integrated into a unified framework for discriminative feature selection and learning. In order to construct high-level image representation, the combined filters in the same layer are fed into next layer to generate the novel analysis dictionary. The experiments in several benchmarks demonstrate the effectiveness of proposed method and achieve good performance on various visual recognition tasks.

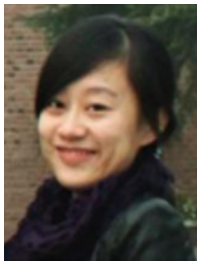
Acknowledgements

This work was supported by the National Natural Science Foundation of China (Nos. 61170193, 61370185), Guangdong Science and Technology Program (No. 2012B031500006), Guangdong Natural Science Foundation (Nos. S2012020011081, S2013010013432), Special Project on Integration of Industry, Education and Research of Guangdong Province (No. 2012B091000101), and Program of Guangzhou Zhujiang Star of Science and Technology (No. 2013J2200067). Corresponding authors of this work is Liang Lin.

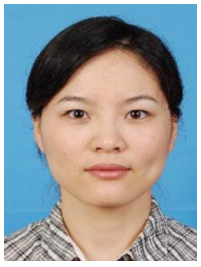
References

[1] K. Grauman, T. Darrell, The pyramid match kernel: discriminative classification with sets of image features, in: Tenth IEEE International Conference on Computer Vision, 2005, ICCV 2005, vol. 2, IEEE, 2005, pp. 1458–1465.
[2] J. Winn, A. Criminisi, T. Minka, Object categorization by learned universal visual dictionary, in: Tenth IEEE International Conference on Computer Vision, 2005, ICCV 2005, vol. 2, IEEE, 2005, pp. 1800–1807.
[3] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: spatial pyramid matching for recognizing natural scene categories, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2006, vol. 2, IEEE, 2006, pp. 2169–2178.
[4] L. Fei-Fei, P. Perona, A Bayesian hierarchical model for learning natural scene categories, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005, CVPR 2005, vol. 2, IEEE, 2005, pp. 524–531.
[5] K. Mikolajczyk, C. Schmid, Scale & affine invariant interest point detectors, Int. J. Comput. Vis. 60 (1) (2004) 63–86.
[6] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005, CVPR 2005, vol. 1, IEEE, 2005, pp. 886–893.
[7] J. Yang, K. Yu, Y. Gong, T. Huang, Linear spatial pyramid matching using sparse coding for image classification, in: IEEE Conference on Computer Vision and Pattern Recognition, 2009, CVPR 2009, IEEE, 2009, pp. 1794–1801.
[8] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, Y. Gong, Locality-constrained linear coding for image classification, in: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2010, pp. 3360–3367.
[9] X. Zhou, K. Yu, T. Zhang, T.S. Huang, Image classification using super-vector coding of local image descriptors, in: Computer Vision–ECCV 2010, Springer, 2010, pp. 141–154.
[10] Z. Jiang, Z. Lin, L.S. Davis, Label consistent k-svd: learning a discriminative dictionary for recognition, IEEE Trans. Pattern Anal. Mach. Intell. 35 (11) (2013) 2651–2664.
[11] J. Mairal, J. Ponce, G. Sapiro, A. Zisserman, F.R. Bach, Supervised dictionary learning, Adv. Neural Inf. Process. Syst., 2009, pp. 1033–1040.
[12] M. Yang, L. Zhang, X. Feng, D. Zhang, Sparse representation based fisher discrimination dictionary learning for image classification, Int. J. Comput. Vis. 109 (3) (2014) 209–232.
[13] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324.
[14] G.E. Hinton, S. Osindero, Y.-W. Teh, A fast learning algorithm for deep belief nets, Neural Comput. 18 (7) (2006) 1527–1554.
[15] H. Lee, R. Grosse, R. Ranganath, A.Y. Ng, Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations, in: Proceedings of the 26th Annual International Conference on Machine Learning, ACM, 2009, pp. 609–616.
[16] R. Zhang, L. Lin, R. Zhang, W. Zuo, L. Zhang, Bit-scalable deep hashing with regularized similarity learning for image retrieval and person re-identification, IEEE Trans. Image Process 24 (12) (2015) 4766–4779.
[17] L. Lin, T. Wu, J. Porway, Z. Xu, A stochastic graph grammar for compositional object representation and recognition, Pattern Recognit. 42 (7) (2009) 1297–1307.
[18] S. Ding, L. Lin, G. Wang, H. Chao, Deep feature learning with relative distance comparison for person re-identification, Pattern Recognit. 48 (10) (2015) 2993–3003.
[19] M.D. Zeiler, R. Fergus, Visualizing and understanding convolutional networks, in: Computer Vision–ECCV 2014, Springer, 2014, pp. 818–833.
[20] J. Friedman, T. Hastie, R. Tibshirani, et al., Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors), Ann. Stat. 28 (2) (2000) 337–407.
[21] Y. Rui, T.S. Huang, S.-F. Chang, Image retrieval: current techniques, promising directions, and open issues, J. Vis. Commun. Image Represent. 10 (1) (1999) 39–62.
[22] Z. Si, S.-C. Zhu, Learning hybrid image templates (hit) by information projection, IEEE Trans. Pattern Anal. Mach. Intell. 34 (7) (2012) 1354–1367.
[23] J. Canny, A computational approach to edge detection, IEEE Trans. Pattern Anal. Mach. Intell., 6 (1986) 679–698.
[24] B.A. Olshausen, et al., Emergence of simple-cell receptive field properties by learning a sparse code for natural images, Nature 381 (6583) (1996) 607–609.
[25] L. Lin, X. Wang, W. Yang, J.-H. Lai, Discriminatively trained and-or graph models for object shape detection, IEEE Trans. Pattern Anal. Mach. Intell. 37 (5) (2015) 959–972.
[26] P. Luo, L. Lin, X. Liu, Learning compositional shape models of multiple distance metrics by information projection, IEEE Trans. Neural Netw. Learn. Syst.
[27] A. Oliva, A. Torralba, Modeling the shape of the scene: a holistic representation of the spatial envelope, Int. J. Comput. Vis. 42 (3) (2001) 145–175.
[28] P. Luo, X. Wang, X. Tang, A deep sum-product architecture for robust facial attributes analysis, in: IEEE International Conference on Computer Vision (ICCV), 2013, IEEE, 2013, pp. 2864–2871.
[29] M. Elad, P. Milanfar, R. Rubinstein, Analysis versus synthesis in signal priors, Inverse Probl. 23 (3) (2007) 947.
[30] P. Sprechmann, R. Litman, T.B. Yakar, A.M. Bronstein, G. Sapiro, Supervised sparse analysis and synthesis operators, Adv. Neural Inf. Process. Syst. (2013) 908–916.

[31] R. Rubinstein, T. Peleg, M. Elad, Analysis k-svd: a dictionary-learning algorithm for the analysis sparse model, IEEE Trans. Signal Process. 61 (3) (2013) 661–677.
[32] A. Torralba, K.P. Murphy, W.T. Freeman, Sharing features: efficient boosting procedures for multiclass object detection, in: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004, CVPR 2004, vol. 2, IEEE, 2004, pp. 762–769.
[33] S. Gu, L. Zhang, W. Zuo, X. Feng, Projective dictionary pair learning for pattern classification, Adv. Neural Inf. Process. Syst. (2014) 793–801.
[34] Y.N. Wu, Z. Si, H. Gong, S.-C. Zhu, Learning active basis model for object detection and recognition, Int. J. Comput. Vis. 90 (2) (2010) 198–235.
[35] R. Rifkin, A. Klautau, In defense of one-vs-all classification, J. Mach. Learn. Res. 5 (2004) 101–141.
[36] A. Coates, A.Y. Ng, The importance of encoding versus training with sparse coding and vector quantization, in: Proceedings of the 28th International Conference on Machine Learning (ICML-11), 2011, pp. 921–928.
[37] A. Coates, A.Y. Ng, Selecting receptive fields in deep networks, Adv. Neural Inf. Process. Syst. (2011) 2528–2536.
[38] Z. Si, S.-C. Zhu, Learning and-or templates for object recognition and detection, IEEE Trans. Pattern Anal. Mach. Intell. 35 (9) (2013) 2189–2205.
[39] P.F. Felzenszwalb, R.B. Girshick, D. McAllester, D. Ramanan, Object detection with discriminatively trained part-based models, IEEE Trans. Pattern Anal. Mach. Intell. 32 (9) (2010) 1627–1645.
[40] K. Smith-Miles, X. Geng, Z.-H. Zhou, Correction to "automatic age estimation based on facial aging patterns", IEEE Trans. Pattern Anal. Mach. Intell. 30 (2) (2008) 0368.
[41] X. Geng, C. Yin, Z.-H. Zhou, Facial age estimation by learning from label distributions, IEEE Trans. Pattern Anal. Mach. Intell. 35 (10) (2013) 2401–2412.
[42] A. Lanitis, C.J. Taylor, T.F. Cootes, Toward automatic simulation of aging effects on face images, IEEE Trans. Pattern Anal. Mach. Intell. 24 (4) (2002) 442–455.
[43] X. Geng, Z.-H. Zhou, K. Smith-Miles, Automatic age estimation based on facial aging patterns, IEEE Trans. Pattern Anal. Mach. Intell. 29 (12) (2007) 2234–2240.
[44] D.T. Nguyen, S.R. Cho, K.R. Park, Human age estimation based on multi-level local binary pattern and regression method, Futur. Inf. Technol., 2014, pp. 433–438.



Zhanglin Peng received the B.E. degree from the School of Software, Sun Yat-sen University Guangzhou, China, in 2013. She is currently pursuing the M.E. degree with the School of Information Science and Technology. Her research interests include computer vision and machine learning.



Ya Li is currently a Ph.D. candidate in School of Information Science and Technology at Sun Yat-sen University, China. She received the B.E. degree from Zhengzhou University, Zhengzhou, China, in 2002 and M.E. degree from Southwest Jiaotong University, Chengdu, China, in 2006. She is also an assistant professor in Guangzhou University, Guangzhou, China. Her current research focuses on computer vision and machine learning.

Zhaoquan Cai is a Professor with Huizhou University, Huizhou, China. His research interest include computer networks, intelligent computing, and database systems.

Liang Lin is a Professor with the School of Data and Computer Science, Sun Yat-sen University (SYSU), China. He received the B.S. and Ph.D. degrees from the Beijing Institute of Technology (BIT), Beijing, China, in 1999 and 2008, respectively. From 2006 to 2007, he was a joint Ph.D. student with the Department of Statistics, University of California, Los Angeles (UCLA). His Ph.D. dissertation was nominated by the China National Excellent Ph.D. Thesis Award in 2010. He was a Post-Doctoral Research Fellow with the Center for Vision, Cognition, Learning, and Art of UCLA. His research focuses on new models, algorithms and systems for intelligent processing and understanding of visual data. He has published more than 60 papers in top tier academic journals and conferences, and has served as an associate editor for journal Neurocomputing and The Visual Computer. He was supported by several promotive programs or funds for his works, such as Program for New Century Excellent Talents of Ministry of Education (China) in 2012, and Guangdong NSFs for Distinguished Young Scholars in 2013. He received the Best Paper Runners-Up Award in ACM NPAR 2010, Google Faculty Award in 2012, and Best Student Paper Award in IEEE ICME 2014.