Facial Age Estimation by Learning from Label Distributions

Xin Geng, Chao Yin, and Zhi-Hua Zhou, *Fellow, IEEE*

Abstract—One of the main difficulties in facial age estimation is that the learning algorithms cannot expect sufficient and complete training data. Fortunately, the faces at close ages look quite similar since aging is a slow and smooth process. Inspired by this observation, instead of considering each face image as an instance with one label (age), this paper regards each face image as an instance associated with a label distribution. The label distribution covers a certain number of class labels, representing the degree that each label describes the instance. Through this way, one face image can contribute to not only the learning of its chronological age, but also the learning of its adjacent ages. Two algorithms named IIS-LLD and CPNN are proposed to learn from such label distributions. Experimental results on two aging face databases show remarkable advantages of the proposed label distribution learning algorithms over the compared single-label learning algorithms, either specially designed for age estimation or for general purpose.

Index Terms—Age estimation, face image, label distribution, machine learning

1 Introduction

People's behavior and preferences are different at different ages [2], which indicates vast potential applications of automatic age estimation. Among many age-related traits, facial appearance might be the most common one that people rely on for age estimation in daily life. As the typical example shown in Fig. 1, the appearance of human faces exhibits remarkable changes with the progress of aging. However, the human estimation of facial age is usually not as accurate as other kinds of facial information, such as identity, expression and gender. Hence developing automatic facial age estimation methods that are comparable or even superior to the human ability in age estimation has become an attractive yet challenging topic emerging in recent years [9].

One of the early works on *exact* age estimation was done by Lanitis et al. [20], [19], where the aging pattern was represented by a quadratic function called *aging function*. Based on this, they proposed the WAS (Weighted Appearance Specific) method [20] and the AAS (Appearance and Age Specific) method [19]. Later, Geng et al. [12], [11] proposed the AGES algorithm based on the subspace trained on a data structure called *aging pattern vector*. After that, various methods were developed for facial age estimation. For example, Fu et al. [8], [10] proposed an age estimation method based on multiple linear regression on the discriminative aging manifold of face images. Guo et al. [14] used the SVR (Support Vector Regression) method to design a locally adjusted robust regressor for the prediction of human ages. They later proposed to use the Biologically Inspired Features (BIF) [16] and the Kernel Partial Least Squares (KPLS) regression [15] for age estimation. Yan et al. [36] regarded age estimation as a regression problem with nonnegative label intervals and solved the problem through semidefinite programming. They also proposed an EM algorithm to solve the regression problem and speed up the optimization process [35]. By using the Spatially Flexible Patch (SFP) as the feature descriptor, the age regression was further improved with the patch-based Gaussian mixture model [38] and the patch-based hidden Markov model [41]. Noticing the advantages of personalized age estimation, Zhang and Yeung [40] formulated the problem as a multi-task learning problem and proposed the multi-task warped Gaussian process to learn a separate age estimator for each person. In order to build a robust facial age estimation system, Ni et al. [22], [23] proposed a method based on the mining of the noisy aging face images collected from the web images and videos. One of the most recent progresses was made by Chang et al. [3], who transformed an age estimation task into multiple cost-sensitive binary classification subproblems, and solved the problem with an ordinal hyperplane ranking algorithm.

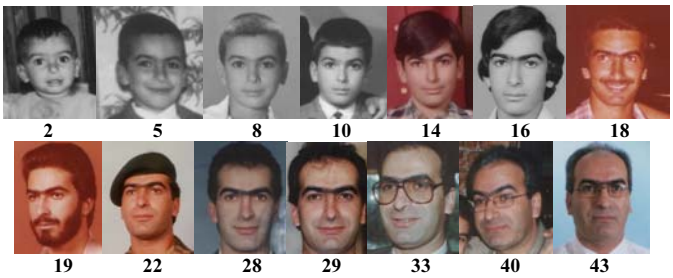

Fig. 1. The aging faces of one subject in the FG-NET Database [20]. The chronological ages are given under the images.

Although a number of algorithms have been successfully developed for facial age estimation, many challenges still remain, among which perhaps the most prominent one is that the learning algorithms cannot expect sufficient and complete training data [11], [40]. Since different people age differently

- *Xin Geng and Chao Yin are with the School of Computer Science and Engineering, Southeast University, Nanjing 211189, China. E-mail: {xgeng, cyin}@seu.edu.cn*
- *Zhi-Hua Zhou is with the National Key Lab for Novel Software Technology, Nanjing University, Nanjing 210046, China. E-mail: zhouzh@lamda.nju.edu.cn*

[11], a 'sufficient and complete' data set should contain the complete aging patterns of as many people as are necessary to represent the whole population. However, the aging progress cannot be artificially controlled. The collection of the aging images thus usually requires great effort in searching for photos taken years ago, and future images cannot be acquired. In practice, almost nobody can guarantee to have at least one photo at each of his/her past ages. Consequently, it is very rare that the complete aging pattern of a person can be successfully collected. Moreover, age is a concept that can be gradually refined from years to months, and even to days. It is practically impossible to have one or more instances at each of these aging points. Due to the above reasons, the aging data can hardly be 'sufficient and complete'. The available data sets [20], [27] typically just contain a very limited number of aging images for each person, and the images at the higher ages are especially rare.

Without sufficient and complete training data, additional knowledge about the aging faces can be introduced to reinforce the learning process. By another close look at Fig. 1, one may find that the faces at the close ages look quite similar. This results from the fact that aging is a slow and gradual process. For example, although a person on the day before his 26'th birthday is still of the age 25, his facial appearance will be almost exactly the same one day later when he turns 26 years old. So, while his chronological age on the day is 25, the age 26 can also be used to describe his facial appearance. This is consistent with the real life experience that people usually predict another person's facial age in the way like "around 25 years old", which indicates using not only 25, but also the neighboring ages to describe the appearance of the face. In this sense, although the chronological age is unambiguous, the facial appearance age is ambiguous, i.e., multiple age numbers might be used to describe the appearance of one face.

Inspired by this observation, the basic idea behind this paper is to utilize the images at the neighboring ages while learning a particular age. This is achieved by introducing a new labeling method, i.e., assigning a *label distribution* to each image rather than a single label of the chronological age. The label distribution covers a certain number of neighboring ages, representing the degree that each age describes the facial appearance. A suitable label distribution will make a face image contribute to not only the learning of its chronological age, but also the learning of its neighboring ages. Compared with the traditional ways of labeling (e.g., single-label and multi-label [32]) in supervised learning, label distribution provides more flexibility in representing ambiguity, which is further discussed in Section 2. Accordingly, the algorithms learning from label distributions should be able to deal with such ambiguity, which is further discussed in Section 3. In this paper, two novel algorithms for label distribution learning are proposed and applied to the problem of facial age estimation.

The rest of the paper is organized as follows. First, the concept of label distribution is introduced in Section 2. Then, two label distribution learning algorithms are proposed in Section 3. After that, the experiments on facial age estimation are reported in Section 4. Finally, the conclusions are drawn and some discussions of future work are given in Section 5.

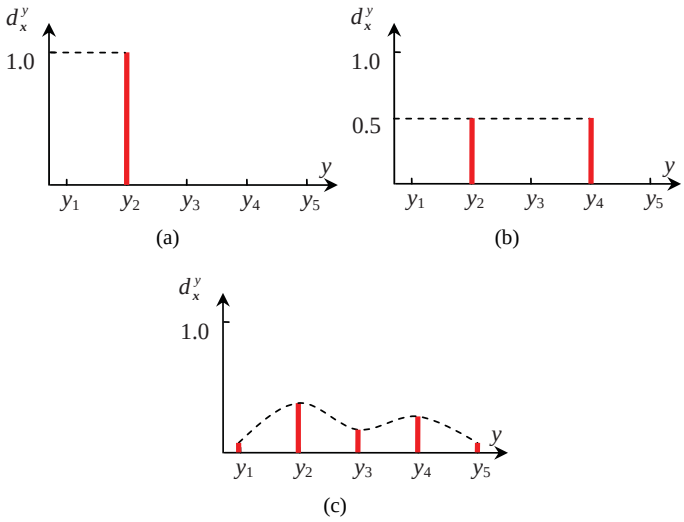

Fig. 2. Three cases of label distribution: (a) single label, (b) multiple labels, and (c) a general case of label distribution.

2 Label Distribution

In the label distribution of an instance $\boldsymbol{x}$, a real number $d_{\boldsymbol{x}}^y \in [0,1]$ called *description degree* is assigned to each label y, representing the degree that y describes $\boldsymbol{x}$. The description degrees of all the labels sum up to 1, indicating a full class description of the instance. Since age is essentially a continuous time spectrum, the age label distribution can be defined as a continuous distribution. But in practice, age is usually measured in years, which is actually a discrete sampling over the time spectrum. Thus the label distribution is defined as a discrete distribution in this paper. Under this definition, the traditional ways to label an instance with a single label or multiple labels can all be viewed as special cases of label distribution. Some typical examples of the label distributions for five class labels are shown in Fig. 2. For case (a), a single label is assigned to the instance, so $d_{\boldsymbol{x}}^{y_2} = 1$ means that the class label y_2 fully describes the instance. For case (b), two labels (y_2 and y_4) are assigned to the instance, so each of them by default describes 50% of the instance, i.e., $d_{\boldsymbol{x}}^{y_2} = d_{\boldsymbol{x}}^{y_4} = 0.5$. Finally, case (c) represents a general case of label distribution satisfying the constraints $d_{\boldsymbol{x}}^y \in [0,1]$ and $\sum_y d_{\boldsymbol{x}}^y = 1$.

Special attention should be paid to the meaning of $d_{\boldsymbol{x}}^y$, which is *not* the *probability* that y correctly labels $\boldsymbol{x}$, but the proportion that y accounts for in a full class description of $\boldsymbol{x}$. Thus, all the labels with a non-zero description degree are actually the 'correct' labels to describe the instance, but just with different importance measured by $d_{\boldsymbol{x}}^y$. Recognizing this, one can distinguish label distribution from the previous studies on probabilistic labels [30], [5], [25], where the basic assumption is that there is only one 'correct' label for each instance. Probabilistic labels are mainly used in the cases where the real label of the instance cannot be obtained with certainty. In practice, it is usually difficult to determine the probability (or confidence) of a label. In most cases, it relies on the prior knowledge of the human experts, which is a highly

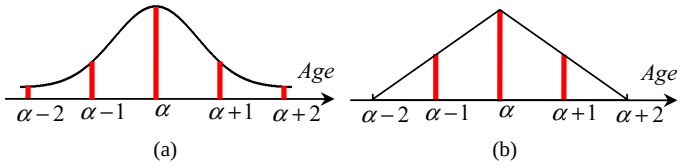

Fig. 3. Typical label distributions for the image at the chronological age α: (a) Gaussian distribution and (b) triangle distribution.

subjective and variable process. As a result, the problem of learning from probabilistic labels has not been extensively studied to date. Fortunately, although not a probability by definition, $d_{\boldsymbol{x}}^y$ still shares the same constraints with probability, i.e., $d_{\boldsymbol{x}}^y \in [0,1]$ and $\sum_y d_{\boldsymbol{x}}^y = 1$. Thus many theories and methods in statistics can be applied to label distributions.

It is also worthwhile to distinguish description degree from the concept *membership* used in *fuzzy classification* [42]. Membership is a truth value that may range between completely true and completely false. It is designed to handle the status of *partial truth* which often appears in the non-numeric linguistic variables. For example, the age 25 might have a membership of 0.7 to the linguistic category 'young', and 0.3 to 'middle age'. But for a particular face, its association with the chronological age 25 will be either completely true or completely false. On the other hand, description degree reflects the *ambiguity* of the class description of the instance, i.e., one class label may only partially describe the instance. For example, due to the appearance similarity of the neighboring ages, both the chronological age 25 and the neighboring ages 24 and 26 can be used to describe the appearance of a 25-year-old face. For each of 24, 25, and 26, it is completely true that it can be used to describe the face (in the sense of appearance). Each age's description degree indicates how much the age contributes to the full class description of the face.

The prior label distribution assigned to a face image at the chronological age α should satisfy the following two properties: 1) The description degree of α is the highest in the label distribution, which ensures the leading position of the chronological age in the class description; 2) The description degree of other ages decreases with the increase of the distance away from α, which makes the age closer to the chronological age contribute more to the class description. While there are many possibilities, Fig. 3 shows two kinds of prior label distributions for the images at the chronological age α, i.e., the Gaussian distribution and the triangle distribution. Note that the age y is regarded as a discrete class label in this paper while both the Gaussian and triangle distributions are defined by continuous density functions $p(y)$. Directly letting $d_{\boldsymbol{x}}^y = p(y)$ might induce $\sum_y d_{\boldsymbol{x}}^y \neq 1$. Thus a normalization process $d_{\boldsymbol{x}}^y = p(y)/\sum_y p(y)$ is required to ensure $\sum_y d_{\boldsymbol{x}}^y = 1$.

3 Learning from Label Distributions

3.1 Problem Formulation

As mentioned before, many theories and methods from statistics can be borrowed to deal with label distributions. First of all, the description degree $d_{\boldsymbol{x}}^y$ could be represented by the form of conditional probability, i.e., $d_{\boldsymbol{x}}^y = P(y|\boldsymbol{x})$. This might be explained as that given an instance $\boldsymbol{x}$, the probability of the presence of y is equal to its description degree. Then, the problem of label distribution learning can be formulated as follows.

> Let $\mathcal{X} = \mathbb{R}^q$ denote the input space and $\mathcal{Y} = \{y_1, y_2, \cdots, y_c\}$ denote the finite set of possible class labels. Given a training set $S = \{(\boldsymbol{x}_1, D_1), (\boldsymbol{x}_2, D_2), \cdots, (\boldsymbol{x}_n, D_n)\}$, where $\boldsymbol{x}_i \in \mathcal{X}$ is an instance, $D_i = \{d_{\boldsymbol{x}_i}^{y_1}, d_{\boldsymbol{x}_i}^{y_2}, \cdots, d_{\boldsymbol{x}_i}^{y_c}\}$ is the label distribution associated with $\boldsymbol{x}_i$, the goal of label distribution learning is to learn a conditional probability mass function $p(y|\boldsymbol{x})$ from S, where $\boldsymbol{x} \in \mathcal{X}$ and $y \in \mathcal{Y}$.

For the problem of age estimation, suppose the same shape of prior label distribution (e.g., Fig. 3) is assigned to each face image, then the highest description degree for each image will be the same, say, p_{max}. Since the description degree of the chronological age should always be the highest in the label distribution, for a face image $\boldsymbol{x}_\alpha$ at the chronological age α, the label distribution learner should output

$$p(\alpha|\boldsymbol{x}_\alpha) = p_{max}, \tag{1}$$

$$p(\alpha+\Delta|\boldsymbol{x}_\alpha) = p_{max} - p_\Delta, \tag{2}$$

where $p_\Delta \in [0,1]$ is the description degree difference from p_{max} when the age changes to a neighboring age $\alpha+\Delta$. Similarly, for a face image $\boldsymbol{x}_{\alpha+\Delta}$ at the chronological age $\alpha+\Delta$,

$$p(\alpha+\Delta|\boldsymbol{x}_{\alpha+\Delta}) = p_{max}. \tag{3}$$

As mentioned before, the faces at the close ages are quite similar, i.e., $\boldsymbol{x}_{\alpha+\Delta} \approx \boldsymbol{x}_\alpha$, thus,

$$p(\alpha+\Delta|\boldsymbol{x}_{\alpha+\Delta}) \approx p(\alpha+\Delta|\boldsymbol{x}_\alpha). \tag{4}$$

So, p_Δ is a small positive number, which indicates that $p(\alpha+\Delta|\boldsymbol{x}_\alpha)$ is just a little bit smaller than $p(\alpha|\boldsymbol{x}_\alpha)$. Note that the above analysis does not depend on any particular form of the prior label distribution except that it must satisfy the two properties mentioned in Section 2. This proves that when applied to age estimation, label distribution learning tends to learn the similarity among the neighboring ages, no matter what the (reasonable) prior label distribution might be.

Suppose $p(y|\boldsymbol{x})$ is a parametric model $p(y|\boldsymbol{x};\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is the vector of the model parameters. Given the training set S, the goal of label distribution learning is to find the $\boldsymbol{\theta}$ that can generate a distribution similar to D_i given the instance $\boldsymbol{x}_i$. If the Kullback-Leibler divergence is used as the measurement of the similarity between two distributions, then the best model parameter vector $\boldsymbol{\theta}^*$ is determined by

$$\begin{aligned}\boldsymbol{\theta}^* &= \operatorname*{argmin}_{\boldsymbol{\theta}} \sum_i \sum_j \left(d_{\boldsymbol{x}_i}^{y_j} \ln \frac{d_{\boldsymbol{x}_i}^{y_j}}{p(y_j|\boldsymbol{x}_i;\boldsymbol{\theta})} \right) \\ &= \operatorname*{argmax}_{\boldsymbol{\theta}} \sum_i \sum_j d_{\boldsymbol{x}_i}^{y_j} \ln p(y_j|\boldsymbol{x}_i;\boldsymbol{\theta}).\end{aligned} \tag{5}$$

It is interesting to examine the traditional learning paradigms under the optimization criterion shown in Eq. (5).

For *single-label learning* (see Fig. 2(a)), $d_{\boldsymbol{x}_i}^{y_j} = Kr(y_j, y(\boldsymbol{x}_i))$, where $Kr(\cdot,\cdot)$ is the Kronecker delta function and $y(\boldsymbol{x}_i)$ is the single class label of $\boldsymbol{x}_i$. Consequently, Eq. (5) can be simplified to

$$\boldsymbol{\theta}^* = \operatorname*{argmax}_{\boldsymbol{\theta}} \sum_i \ln p(y(\boldsymbol{x}_i)|\boldsymbol{x}_i;\boldsymbol{\theta}). \tag{6}$$

This is actually the maximum likelihood (ML) estimation of $\boldsymbol{\theta}$. The later use of $p(y|\boldsymbol{x};\boldsymbol{\theta})$ for classification is equivalent to the maximum a posteriori (MAP) decision.

For *multi-label learning* [32], each instance is associated with a label set (see Fig. 2(b)). Consequently, Eq. (5) can be changed into

$$\boldsymbol{\theta}^* = \operatorname*{argmax}_{\boldsymbol{\theta}} \sum_i \frac{1}{|Y_i|} \sum_{y\in Y_i} \ln p(y|\boldsymbol{x}_i;\boldsymbol{\theta}), \tag{7}$$

where Y_i is the label set associated with $\boldsymbol{x}_i$. Eq. (7) can be viewed as a ML criterion weighted by the reciprocal cardinality of the label set associated with each instance. In fact, this is equivalent to first applying the Entropy-based Label Assignment (ELA) [32], a well-known technique dealing with multi-label data, to transform the multi-label instances into the weighted single-label instances, and then optimizing the ML criterion based on the weighted single-label instances.

It can be seen from the above analysis that with proper constraints, a label distribution learning model can be transformed into the commonly used methods for single-label or multi-label learning. On the one hand, label distribution learning is a more general learning framework which includes single-label learning as its special case. On the other hand, current multi-label learning is mainly concerned with classification. If we consider that one instance belongs to multiple labels with description degrees, then this is equivalent to label distribution learning. Thus, label distribution learning can be regarded as a new branch of multi-label learning, being parallel to multi-label classification. Accordingly, the algorithms that learn from the label distributions should be designed within this new learning framework. In the rest of this section, two different label distribution learning algorithms will be proposed. The first one is called IIS-LLD, which assumes the form of $p(y|\boldsymbol{x})$ to be the *maximum entropy model* [1]. The second one is called CPNN, which models $p(y|\boldsymbol{x})$ by a three layer neural network without assumption of the form of $p(y|\boldsymbol{x})$.

3.2 The IIS-LLD Algorithm

Suppose $f_k(\boldsymbol{x}, y)$ is a *feature function* which depends on both the instance $\boldsymbol{x}$ and the label y. Then, the expected value of f_k w.r.t. the empirical joint distribution $\tilde{p}(\boldsymbol{x}, y)$ in the training set is

$$\tilde{f}_k = \sum_y \int \tilde{p}(\boldsymbol{x}, y) f_k(\boldsymbol{x}, y)\,\mathrm{d}\boldsymbol{x}. \tag{8}$$

The expected value of f_k w.r.t. the conditional model $p(y|\boldsymbol{x};\boldsymbol{\theta})$ and the empirical distribution $\tilde{p}(\boldsymbol{x})$ in the training set is

$$\hat{f}_k = \sum_y \int \tilde{p}(\boldsymbol{x}) p(y|\boldsymbol{x};\boldsymbol{\theta}) f_k(\boldsymbol{x}, y)\,\mathrm{d}\boldsymbol{x}. \tag{9}$$

One reasonable choice of $p(y|\boldsymbol{x};\boldsymbol{\theta})$ is the one that has the maximum *conditional entropy* subject to the constraint $\tilde{f}_k = \hat{f}_k$. It can be proved [1] that such a model (a.k.a. the *maximum entropy model*) has the exponential form

$$p(y|\boldsymbol{x};\boldsymbol{\theta}) = \frac{1}{Z} \exp\left(\sum_k \theta_k f_k(\boldsymbol{x}, y)\right), \tag{10}$$

where $Z = \sum_y \exp\left(\sum_k \theta_k f_k(\boldsymbol{x}, y)\right)$ is the normalization factor and θ_k is the k-th model parameter in $\boldsymbol{\theta}$. In practice, the features usually depend only on the instance but not on the class label. Thus, Eq. (10) can be rewritten as

$$p(y|\boldsymbol{x};\boldsymbol{\theta}) = \frac{1}{Z} \exp\left(\sum_k \theta_{y,k} g_k(\boldsymbol{x})\right), \tag{11}$$

where $g_k(\boldsymbol{x})$ is a class-independent feature function.

Substituting Eq. (11) into Eq. (5) and recognizing $\sum_j d_{\boldsymbol{x}_i}^{y_j} = 1$ yields the target function of $\boldsymbol{\theta}$

$$\begin{aligned} T(\boldsymbol{\theta}) &= \sum_{i,j} d_{\boldsymbol{x}_i}^{y_j} \ln p(y_j|\boldsymbol{x}_i;\boldsymbol{\theta}) \\ &= \sum_{i,j} d_{\boldsymbol{x}_i}^{y_j} \sum_k \theta_{y_j,k} g_k(\boldsymbol{x}_i) - \\ &\quad \sum_i \ln \sum_j \exp\left(\sum_k \theta_{y_j,k} g_k(\boldsymbol{x}_i)\right). \end{aligned} \tag{12}$$

Directly setting the gradient of Eq. (12) w.r.t. $\boldsymbol{\theta}$ to zero does not yield a closed-form solution. Thus the optimization of Eq. (12) uses a strategy similar to Improved Iterative Scaling (IIS) [24], a well-known algorithm for maximizing the likelihood of the maximum entropy model. IIS starts with an arbitrary set of parameters. Then for each step, it updates the current estimate of the parameters $\boldsymbol{\theta}$ to $\boldsymbol{\theta} + \boldsymbol{\Delta}$, where $\boldsymbol{\Delta}$ maximizes a lower bound to the change in likelihood $\Omega = T(\boldsymbol{\theta} + \boldsymbol{\Delta}) - T(\boldsymbol{\theta})$. This iterative process, nevertheless, needs to be migrated to the new target function $T(\boldsymbol{\theta})$. Furthermore, the constraint on the feature functions required by IIS, $f_k(\boldsymbol{x}, y) \geq 0$ (hence $g_k(\boldsymbol{x}) \geq 0$) should be removed to ensure the freedom in choosing any feature extractors suitable for the data.

In detail, the change of $T(\boldsymbol{\theta})$ between adjacent steps is

$$\begin{aligned} T(\boldsymbol{\theta} + \boldsymbol{\Delta}) - T(\boldsymbol{\theta}) &= \sum_{i,j} d_{\boldsymbol{x}_i}^{y_j} \sum_k \delta_{y_j,k} g_k(\boldsymbol{x}_i) - \\ &\quad \sum_i \ln \sum_j p(y_j|\boldsymbol{x}_i;\boldsymbol{\theta}) \exp\left(\sum_k \delta_{y_j,k} g_k(\boldsymbol{x}_i)\right), \end{aligned} \tag{13}$$

where $\delta_{y_j,k}$ is the increment for $\theta_{y_j,k}$. Applying the inequality $-\ln x \geq 1 - x$ yields

$$\begin{aligned} T(\boldsymbol{\theta} + \boldsymbol{\Delta}) - T(\boldsymbol{\theta}) &\geq \sum_{i,j} d_{\boldsymbol{x}_i}^{y_j} \sum_k \delta_{y_j,k} g_k(\boldsymbol{x}_i) + n - \\ &\quad \sum_{i,j} p(y_j|\boldsymbol{x}_i;\boldsymbol{\theta}) \exp\left(\sum_k \delta_{y_j,k} g_k(\boldsymbol{x}_i)\right). \end{aligned} \tag{14}$$

Differentiating the right side of Eq. (14) w.r.t. $\delta_{y_j,k}$ yields the coupled equations of $\delta_{y,k}$ which are hard to be solved.

Algorithm 1: IIS-LLD

Input: The training set $S = \{(\boldsymbol{x}_i, D_i)\}_{i=1}^n$, the feature functions $g_k(\boldsymbol{x})$, and the convergence criterion ε

Output: $p(y|\boldsymbol{x};\boldsymbol{\theta})$

Initialize the model parameter vector $\boldsymbol{\theta}^{(0)}$;
$i \leftarrow 0$;
repeat
  $i \leftarrow i + 1$;
  Solve Eq. (18) for $\delta_{y,k}$;
  $\boldsymbol{\theta}^{(i)} \leftarrow \boldsymbol{\theta}^{(i-1)} + \boldsymbol{\Delta}$;
until $T(\boldsymbol{\theta}^{(i)}) - T(\boldsymbol{\theta}^{(i-1)}) < \varepsilon$;
$p(y|\boldsymbol{x};\boldsymbol{\theta}) \leftarrow \frac{1}{Z}\exp\left(\sum_k \theta_{y,k}^{(i)} g_k(\boldsymbol{x})\right)$;

To decouple the interaction among $\delta_{y,k}$, Jensen's inequality is applied here, i.e., for a probability mass function $p(x)$,

$$\exp\left(\sum_x p(x)q(x)\right) \leq \sum_x p(x)\exp\left(q(x)\right). \quad (15)$$

The last term of Eq. (14) can be rewritten as

$$\sum_{i,j} p(y_j|\boldsymbol{x}_i;\boldsymbol{\theta})\exp\left(\sum_k \delta_{y_j,k} s(g_k(\boldsymbol{x}_i)) g^{\#}(\boldsymbol{x}_i)\frac{|g_k(\boldsymbol{x}_i)|}{g^{\#}(\boldsymbol{x}_i)}\right), \quad (16)$$

where $g^{\#}(\boldsymbol{x}_i) = \sum_k |g_k(\boldsymbol{x}_i)|$ and $s(g_k(\boldsymbol{x}_i))$ is the sign of $g_k(\boldsymbol{x}_i)$. Since $|g_k(\boldsymbol{x}_i)|/g^{\#}(\boldsymbol{x}_i)$ can be viewed as a probability mass function, Jensen's inequality can be applied to Eq. (14) to yield

$$T(\boldsymbol{\theta}+\boldsymbol{\Delta}) - T(\boldsymbol{\theta}) \geq \sum_{i,j} d_{\boldsymbol{x}_i}^{y_j} \sum_k \delta_{y_j,k} g_k(\boldsymbol{x}_i) + n -$$

$$\sum_{i,j} p(y_j|\boldsymbol{x}_i;\boldsymbol{\theta}) \sum_k \frac{|g_k(\boldsymbol{x}_i)|}{g^{\#}(\boldsymbol{x}_i)} \exp(\delta_{y_j,k} s(g_k(\boldsymbol{x}_i)) g^{\#}(\boldsymbol{x}_i)). \quad (17)$$

Denote the right side of Eq. (17) as $\mathcal{A}(\boldsymbol{\Delta}|\boldsymbol{\theta})$, which is a lower bound to $T(\boldsymbol{\theta}+\boldsymbol{\Delta}) - T(\boldsymbol{\theta})$. Setting the derivative of $\mathcal{A}(\boldsymbol{\Delta}|\boldsymbol{\theta})$ w.r.t. $\delta_{y_j,k}$ to zero gives

$$\frac{\partial \mathcal{A}(\boldsymbol{\Delta}|\boldsymbol{\theta})}{\partial \delta_{y_j,k}} = \sum_i d_{\boldsymbol{x}_i}^{y_j} g_k(\boldsymbol{x}_i) -$$

$$\sum_i p(y_j|\boldsymbol{x}_i;\boldsymbol{\theta}) g_k(\boldsymbol{x}_i) \exp(\delta_{y_j,k} s(g_k(\boldsymbol{x}_i)) g^{\#}(\boldsymbol{x}_i)) = 0. \quad (18)$$

What is nice about Eq. (18) is that $\delta_{y,k}$ appears alone, and therefore can be solved one by one through nonlinear equation solvers, such as the Gauss-Newton method. This algorithm is named as IIS-LLD (i.e., IIS - Learning from Label Distributions) and summarized in Algorithm 1.

3.3 The CPNN Algorithm

One of the main assumptions made in the IIS-LLD algorithm is the derivation of $p(y|\boldsymbol{x})$ as the maximum entropy model [1]. While it is a reasonable assumption without additional information, there is no particular evidence supporting it in the problem of age estimation. Alternatively, using a three layer neural network to approximate $p(y|\boldsymbol{x})$ is one approach to removing this assumption. A natural design of such a neural network would have q (the dimensionality of $\boldsymbol{x}$) input units which receive $\boldsymbol{x}$, and c (the number of different labels) output units each of which outputs the description degree of a label y. However, for the problem of age estimation, the number of ages c is usually large (e.g., $c = 70$ in the FG-NET database [20]), which results in many weights between the hidden layer and the output layer. With limited training samples, it will be difficult for the learning algorithm to converge if there are too many weights in the neural network.

Fortunately, since age is a totally ordered label (i.e., a non-negative integer), it can be regarded as a special numerical input into the neural network. Thus the input of the network includes both $\boldsymbol{x}$ and y, and the output of the network is a single value which is expected to be $p(y|\boldsymbol{x})$. The network is therefore called Conditional Probability Neural Network (CPNN). Sarajedini et al. [29] once proposed an unsupervised learning algorithm for conditional probability density function (pdf) estimation, which is based on Modha's neural network pdf estimator [21]. The CPNN proposed in this paper has a similar network structure but is trained in a supervised manner, i.e., the true label distributions are known when training the neural network.

In Modha's pdf estimator [21], the input of the neural network is $\boldsymbol{x}$ and the output is $p(\boldsymbol{x})$. The activation function for the hidden and output layers are the sigmoid function and the exponential function, respectively. The output of the neural network can be written as

$$p(\boldsymbol{x};\boldsymbol{\theta}) = \exp(c(\boldsymbol{\theta}) + f(\boldsymbol{x};\boldsymbol{\theta})), \quad (19)$$

where $\boldsymbol{\theta}$ is the weight vector. The net activation of the output unit $f(\boldsymbol{x};\boldsymbol{\theta})$ is

$$f(\boldsymbol{x};\boldsymbol{\theta}) = \sum_{m=1}^{M_2} \boldsymbol{\theta}_{31m} G\left(\sum_{k=0}^{M_1} \boldsymbol{\theta}_{2mk}\boldsymbol{x}_k\right), \quad (20)$$

where G is the sigmoid activation function, M_l is the number of units on the l-th layer, and $\boldsymbol{\theta}_{lmk}$ is the weight of the m-th unit on the l-th layer associated with the output of the k-th unit on the $(l-1)$-th layer. The input vector $\boldsymbol{x}$ is augmented with the bias input $\boldsymbol{x}_0 \equiv 1$. The bias $c(\boldsymbol{\theta})$ in Eq. (19) ensures that $\int p(\boldsymbol{x})\mathrm{d}\boldsymbol{x} = 1$.

When the input includes both $\boldsymbol{x}$ and a discrete y, the output becomes

$$p(\boldsymbol{x}, y;\boldsymbol{\theta}) = \exp(c(\boldsymbol{\theta}) + f(\boldsymbol{x}, y;\boldsymbol{\theta})). \quad (21)$$

Thus the conditional probability can be calculated as

$$\begin{aligned} p(y|\boldsymbol{x};\boldsymbol{\theta}) &= \frac{p(\boldsymbol{x},y;\boldsymbol{\theta})}{p(\boldsymbol{x};\boldsymbol{\theta})} = \frac{p(\boldsymbol{x},y;\boldsymbol{\theta})}{\sum_y p(\boldsymbol{x},y;\boldsymbol{\theta})} \\ &= \frac{\exp(c(\boldsymbol{\theta}) + f(\boldsymbol{x},y;\boldsymbol{\theta}))}{\sum_y \exp(c(\boldsymbol{\theta}) + f(\boldsymbol{x},y;\boldsymbol{\theta}))} \\ &= \frac{\exp(f(\boldsymbol{x},y;\boldsymbol{\theta}))}{\sum_y \exp(f(\boldsymbol{x},y;\boldsymbol{\theta}))}. \end{aligned} \quad (22)$$

As suggested by Sarajedini et al. [29], $p(y|\boldsymbol{x};\boldsymbol{\theta})$ can be regarded as the output of another neural network:

$$p(y|\boldsymbol{x};\boldsymbol{\theta}) = \exp(b(\boldsymbol{x};\boldsymbol{\theta}) + f(\boldsymbol{x},y;\boldsymbol{\theta})). \quad (23)$$

Comparing Eq. (23) with (22), the new bias should be

$$b(\boldsymbol{x};\boldsymbol{\theta}) = -\ln\left(\sum_y \exp(f(\boldsymbol{x},y;\boldsymbol{\theta}))\right). \quad (24)$$

Recall Eq. (5), then the target function to minimize is

$$\begin{aligned} T(\boldsymbol{\theta}) &= -\sum_i \sum_j d_{\boldsymbol{x}_i}^{y_j} \ln p(y_j|\boldsymbol{x}_i;\boldsymbol{\theta}) \\ &= -\sum_i \sum_j d_{\boldsymbol{x}_i}^{y_j} (b(\boldsymbol{x}_i;\boldsymbol{\theta}) + f(\boldsymbol{x}_i,y_j;\boldsymbol{\theta})). \end{aligned} \quad (25)$$

The gradient of Eq. (25) w.r.t. $\boldsymbol{\theta}$ is

$$\frac{\partial T(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = -\sum_i \sum_j d_{\boldsymbol{x}_i}^{y_j} \left(\frac{\partial b(\boldsymbol{x}_i;\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} + \frac{\partial f(\boldsymbol{x}_i,y_j;\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right), \quad (26)$$

where

$$\frac{\partial b(\boldsymbol{x}_i;\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = -\frac{\sum_j \left(\exp(f(\boldsymbol{x}_i,y_j;\boldsymbol{\theta})) \times \frac{\partial f(\boldsymbol{x}_i,y_j;\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right)}{\sum_j \exp(f(\boldsymbol{x}_i,y_j;\boldsymbol{\theta}))}. \quad (27)$$

The partial derivative of $f(\boldsymbol{x}_i,y_j;\boldsymbol{\theta})$ in Eq. (26) and (27) can be calculated by backpropagation [21], i.e.,

$$\frac{\partial f(\boldsymbol{x}_i,y_j;\boldsymbol{\theta})}{\partial \boldsymbol{\theta}_{lmk}} = z_{(l-1)k}^i \delta_{lm}^i, \quad (28)$$

where $z_{(l-1)k}^i$ is the output of the k-th unit on the $(l-1)$-th layer, and δ_{lm}^i is the partial derivative of $f(\boldsymbol{x}_i,y_j;\boldsymbol{\theta})$ w.r.t. the net activation of the m-th unit on the l-th layer I_{lm}^i. For the output layer ($l=3$),

$$\delta_{31}^i = \frac{\partial f(\boldsymbol{x}_i,y_j;\boldsymbol{\theta})}{\partial I_{31}^i} = 1. \quad (29)$$

For the hidden layer ($l=2$),

$$\begin{aligned} \delta_{2m}^i &= \frac{\partial f(\boldsymbol{x}_i,y_j;\boldsymbol{\theta})}{\partial I_{2m}^i} = G'(I_{2m}^i)\delta_{31}^i \boldsymbol{\theta}_{31m} \\ &= G'(I_{2m}^i)\boldsymbol{\theta}_{31m}. \end{aligned} \quad (30)$$

Finally, after $\partial T(\boldsymbol{\theta})/\partial \boldsymbol{\theta}$ is obtained, the weights are updated by the RPROP algorithm [28]. The CPNN algorithm is summarized in Algorithm 2.

Algorithm 2: CPNN

Input: The training set $S = \{(\boldsymbol{x}_i, D_i)\}_{i=1}^n$, the number of hidden layer units M_2, and the convergence criterion ε

Output: $p(y|\boldsymbol{x};\boldsymbol{\theta})$

```
Initialize the weights of the neural network θ^(0);
i ← 0;
repeat
    i ← i + 1;
    Calculate ∂T(θ)/∂θ by Eq. (26);
    Get θ^(i) by updating θ^(i−1) with RPROP;
until T(θ^(i)) − T(θ^(i−1)) < ε;
p(y|x;θ) ← the output of the neural network;
```

3.4 Classifiers based on Label Distribution Learning

After $p(y|\boldsymbol{x})$ is learned from the training set, either through IIS-LLD or CPNN, the label distribution of any new instance $\boldsymbol{x}'$ can be generated by $p(y|\boldsymbol{x}')$. The availability of the explicit label distribution for $\boldsymbol{x}'$ provides many possibilities in classifier design. To name just a few, if the expected class label for $\boldsymbol{x}'$ is single, then the predicted label could be $y^* = \mathrm{argmax}_y\, p(y|\boldsymbol{x}')$, together with a confidence measure $p(y^*|\boldsymbol{x}')$. If multiple labels are allowed, then the predicted label set could be $L = \{y|p(y|\boldsymbol{x}') > \xi\}$, where ξ is a predefined threshold. Moreover, all the labels in L can be ranked according to their description degrees. For the problem of exact age estimation, the predicted age could be the one with the maximum description degree. For the problem of age range estimation, the predicted age range could be the one with the maximum sum of description degrees of all the ages within an age range.

4 Experiments

4.1 Methodology

Two data sets are used in the experiments. The first is the FG-NET Aging Database [20]. There are $1,002$ face images from 82 subjects in this database. Each subject has 6-18 face images at different ages. Each image is labeled by its chronological age. The ages are distributed in a wide range from 0 to 69. Besides age variation, most of the age-progressive image sequences display other types of facial variations, such as significant changes in pose, illumination, expression, *etc*. A typical aging face sequence in this database is shown in Fig. 1.

The second data set is the much larger MORPH database [27]. There are $55,132$ face images from more than $13,000$ subjects in this database. The average number of images per subject is 4. The ages of the face images range from 16 to 77 with a median age of 33. The faces are from different races, among which the African faces account for about 77%, the European faces account for about 19%, and the remaining 4% includes Hispanic, Asian, Indian, and other races. Some typical aging faces in this database are shown in Fig. 4.

The feature extractor used for the FG-NET database is the Appearance Model [6]. The main advantage of this model is that the extracted features combine the shape and intensity of the face images, both of which are important in the aging progress. In this experiment, the first 200 model parameters are used as the extracted features. The features used for the MORPH database are the Biologically Inspired Features (BIF) [16]. By simulating the primate visual system, BIF has shown good performance in facial age estimation [16]. The dimensionality of the BIF vectors is further reduced to 200 using Marginal Fisher Analysis (MFA) [37].

According to the chronological age of each face image, a label distribution is generated using the Gaussian or triangle distribution shown in Fig. 3. Then the label distribution learning algorithms (IIS-LLD and CPNN) are applied to the image

TABLE 1
Human Tests on Age Perception

Data Set	# Samples	# Testees			Testees' Age			MAE (HumanA)			MAE (HumanB)		
		Males	Females	Total	Min.	Max.	Avg.	Min.	Max.	Avg.	Min.	Max.	Avg.
FG-NET	51	24	5	29	22	44	25	4.88	15.67	8.13	4.14	8.33	6.23
MORPH	60	28	12	40	16	64	26	5.47	13.28	8.24	4.78	11.03	7.23

Fig. 4. Typical aging faces of two subjects in the MORPH database [27]. The chronological ages are given under the images.

set with the generated label distributions. The predicted age for a test image $\boldsymbol{x}'$ is determined by $y^* = \mathrm{argmax}_y\, p(y|\boldsymbol{x}')$. To study the usefulness of the adjacent ages, IIS-LLD and CPNN are also applied to the special label distribution of the single-label case shown in Fig. 2(a). The three kinds of label distributions are denoted by "Gaussian", "Triangle", and "Single", respectively. Several existing algorithms specially designed for the problem of facial age estimation are compared as the baseline methods, which include OHRank [3], AGES [11], WAS [20], and AAS [19]. Some conventional general-purpose classification methods for single-label data are also compared, which include kNN (k-Nearest Neighbors), BP (Backpropagation neural network), C4.5 (C4.5 decision tree), SVM (Support Vector Machine), and a fuzzy classifier ANFIS (Adaptive-Network-Based Fuzzy Inference System) [17]. For all of these general-purpose methods, age estimation is formulated as a standard multi-class classification problem, i.e., each face is labeled by its chronological age, and the number of classes is equal to the number of possible ages.

When generating the label distributions, the standard deviation of the "Gaussian" distribution varies within four different values 1, 2, 3 and 4. The bottom length of the "Triangle" distribution also varies within four different values 4, 6, 8, and 10. All of these label distribution settings are tested and the best results are reported. For all the compared algorithms, several parameter configurations are tested and the best results are reported. For CPNN, the number of hidden layer units is set to 400. For OHRank, the absolute cost function and the RBF kernel are used. For AGES, the aging pattern subspace dimensionality is set to 20. In AAS, the error threshold in the appearance cluster training step is set to 3. For kNN, k is set to 30 and Euclidean distance is used to find the neighbors. The BP neural network has a hidden layer of 100 neurons with sigmoid activation functions. The parameters of C4.5 are set to the default values of the J4.8 implementation (i.e., the confidence threshold 0.25 for pruning and minimum 2 instances per leaf). SVM is implemented as the 'C-SVC' type in LIBSVM using the RBF kernel with the inverse width of 1. Finally, the number of membership functions in ANFIS is set to 2.

The performance of the age estimators is evaluated by MAE (Mean Absolute Error), i.e., the average absolute difference between the estimated age and the chronological age. The algorithms are tested through the LOPO (Leave-One-Person-Out) mode [12] on the FG-NET database, i.e., in each fold, the images of one person are used as the test set and those of the others are used as the training set. After 82 folds, each subject has been used as test set once, and the final results are calculated from all the estimates. Since there are more than $13,000$ subjects in the MORPH database, the LOPO test will be too time-consuming. Thus the algorithms are tested through the 10-fold cross validation on the MORPH database.

As an important baseline, the human ability in age perception is also tested. About 5% of the images from the FG-NET database (i.e., 51 face images) and 60 images from the MORPH database are uniformly sampled from the age ranges shown in Table 4. These images are used as the test samples presented to the human testees. All the testees are Chinese students or staff members from the authors' universities. Some other ground truth of the human tests, including the number of test samples, the number of testees, and the testees' own age, is shown in Table 1.

There are two stages in the human tests. In each stage, the images are randomly presented to the testees, and the testees are asked to choose one age from a given range (0-69 for FG-NET and 16-77 for MORPH) for each image. The difference between the two stages is that in the first stage (HumanA), only the gray-scale face regions (i.e., the color images are converted to the gray-scale images and the background of the images is removed) are shown, while in the second stage (HumanB), the whole color images are shown. Fig. 5 gives an example of the same face shown in the HumanA test and HumanB test, respectively. HumanA intends to test the age estimation ability purely based on the intensity of the face image, which is also the input to the algorithms, while HumanB intends to test the age estimation ability based on multiple traits including face, hair, skin color, clothes, background, etc.

In both the HumanA and HumanB tests, each testee is required to label all the test samples, and the MAE of each testee is recorded. The minimum, maximum, and average MAE of all the testees involved in each test are given in Table 1. The average MAE can be regarded as a measurement of the human accuracy in age estimation. As can be seen, the testees perform

(a) (b)

Fig. 5. An example of the same face shown in (a) the HumanA test, and (b) the HumanB test.

TABLE 2
MAE (in Years) of Different Age Estimators

Method		Data Set	
		FG-NET	MORPH
IIS-LLD	Gaussian	**5.77** (1, 1)	**6.67±0.15** (1, 1)
	Triangle	**5.90** (1, 0)	**6.09±0.14** (1, 1)
	Single	**6.27** (1, 0)	**6.35±0.17** (1, 1)
CPNN	Gaussian	**4.76** (1, 1)	**4.87±0.31** (1, 1)
	Triangle	**5.07** (1, 1)	**4.91±0.29** (1, 1)
	Single	**5.31** (1, 1)	**6.59±0.31** (1, 1)
OHRank		**6.27** (1, 0)	**6.28±0.18** (1, 1)
AGES		**6.77** (1, 1)	**6.61±0.11** (1, 1)
WAS		**8.06** (0, 1)	9.21±0.16 (1, 1)
AAS		14.83 (1, 1)	10.10±0.26 (1, 1)
kNN		8.24 (0, 1)	9.64±0.24 (1, 1)
BP		11.85 (1, 1)	12.59±1.38 (1, 1)
C4.5		9.34 (1, 1)	**7.48±0.12** (1, 0)
SVM		**7.25** (1, 1)	**7.34±0.17** (1, 0)
ANFIS		8.86 (0, 1)	9.24±0.17 (1, 1)
Human Tests[1]	HumanA	8.13	8.24
	HumanB	6.23	7.23

[1] The human tests are performed on 5% samples from the FG-NET database and 60 samples from the MORPH database.

remarkably better in the HumanB test than in the HumanA test, which indicates that the additional information (hair, skin color, clothes, background, etc.) provided in the HumanB test is helpful to improve the human accuracy in age estimation.

4.2 Results

The MAEs of all the age estimators are tabulated in Table 2. The standard deviations on the MORPH database are also given in the table. Note that the number of images for each person in the FG-NET database varies dramatically. Consequently, the standard deviation of the LOPO test on the FG-NET database becomes unstable. So it is not shown in Table 2. The MAEs of the algorithms higher than that of HumanA are highlighted by boldface and those higher than that of HumanB are underlined. Since the results of the human tests are the mean MAEs of multiple testees, the two-tailed *t*-tests at the 5% significance level are performed to see whether the differences between the results of the human tests and the algorithms are statistically significant. The results of the *t*-tests are given in the brackets right after the MAE of each algorithm in Table 2. The number '1' represents significant difference, '0' represents otherwise. The first number is the *t*-test result on HumanA, the second is that on HumanB.

As can be seen, the overall performance of the label distribution learning algorithms (IIS-LLD and CPNN) is significantly better than that of the single-label based algorithms, either specially designed for age estimation (OHRank, AGES, WAS, and AAS) or for general-purpose classification (kNN, BP, C4.5, SVM, and ANFIS). There are mainly two reasons for the good performance of the label distribution learning algorithms. Firstly, the prior label distributions of the training samples make it possible that one instance contributes to the learning of multiple classes. Secondly, as discussed in Section 3.1, the label distribution learning algorithms tend to learn the similarity among the neighboring ages, no matter what the (reasonable) prior label distribution might be. The second reason also explains why the "Single" case of IIS-LLD or CPNN can achieve state-of-the-art results even when the prior label distribution in this case is equivalent to single label. Refer back to Eq. (6), the learning target of the "Single" case is to ensure the dominating position of the chronological age in the label distribution. Although no prior knowledge about the neighboring ages is given, the label distribution learning algorithms can learn it based on the similarity of the face images at the close ages.

In all cases, IIS-LLD and CPNN perform significantly better than HumanA. Except for the "Triangle" and "Single" cases of IIS-LLD on FG-NET, IIS-LLD and CPNN perform even significantly better than HumanB. Considering that more information is actually provided to the human testees in the HumanB test, it can be concluded that *under the experimental settings of this paper*, IIS-LLD and CPNN can both achieve better performance than that of the human testees. However, it would be too optimistic to claim that the algorithms can outperform humans in general. The main reason is that people usually perform better for faces belonging to their own race than for those belonging to another race [4]. While most images in the FG-NET and MORPH databases are Caucasian and African faces, the testees involved in the human tests are all Chinese. Thus the results of the human tests are actually biased toward a more difficult task: estimate the age of the faces from a different race.

The comparison between IIS-LLD and CPNN in Table 2 shows clear advantage of CPNN. There are mainly two reasons why CPNN performs better. Firstly, CPNN learns $p(y|\boldsymbol{x})$ without prior assumptions of its form, while IIS-LLD assumes $p(y|\boldsymbol{x})$ to be the maximum entropy model, which does not necessarily match the problem of age estimation well. Secondly, all the class labels share the same set of model parameters in CPNN while IIS-LLD learns the parameters for each class label separately, i.e., the $\theta_{y,k}$ in Eq. (11) can be learned separately for each y. Thus CPNN can better utilize the correlation among the class labels. Nevertheless, there are also at least two disadvantages of CPNN compared to IIS-LLD. Firstly, relying more on the training data than IIS-LLD makes CPNN more vulnerable to overfitting, which can be evidenced by its higher standard deviations in Table 2. Secondly, in IIS-LLD, the difference of the target function values between the adjacent steps is maximized, and the parameter increment $\delta_{y_j,k}$ in Eq. (18) appears alone. So, IIS-LLD runs faster than CPNN. The pros and cons of these two label distribution learning

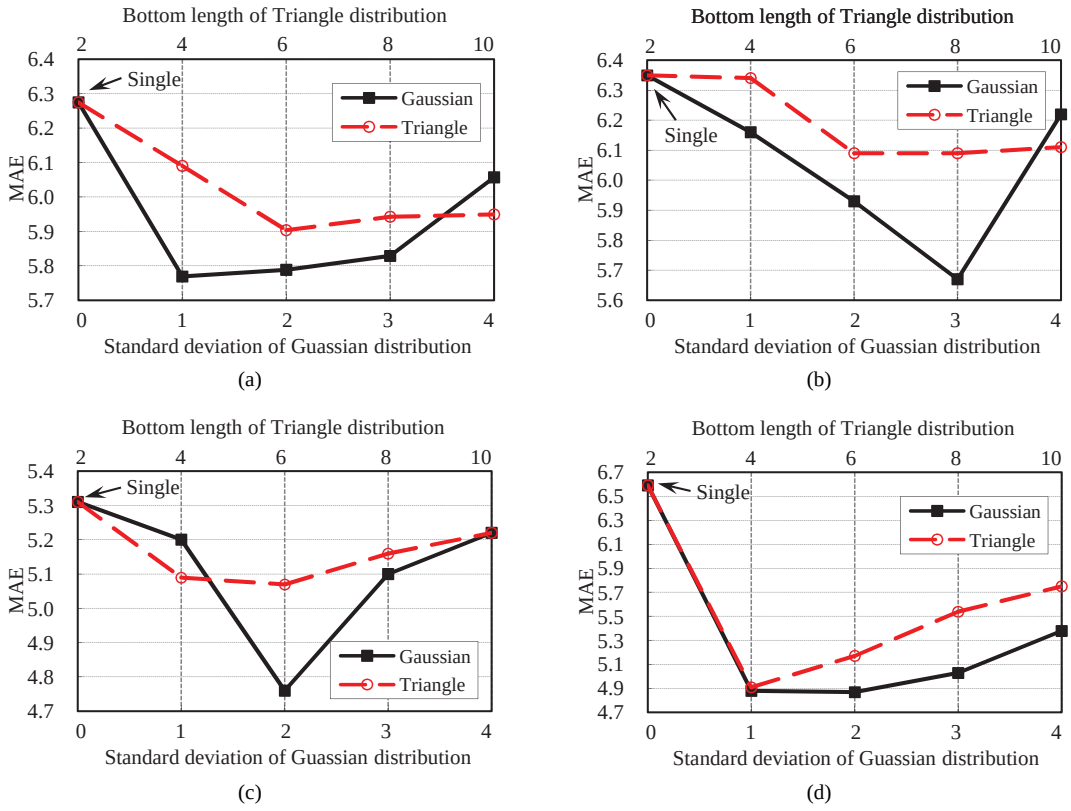

Fig. 6. The performance on different "Gaussian" distributions (with different σ) and different "Triangle" distributions (with different l): (a) IIS-LLD on FG-NET, (b) IIS-LLD on MORPH, (c) CPNN on FG-NET, and (d) CPNN on MORPH.

TABLE 3
Pros and Cons of IIS-LLD and CPNN

	Pros	Cons
IIS-LLD	• Faster • Less vulnerable to overfitting	• Prior assumption of maximum entropy model • Separate model parameters for each label
CPNN	• No assumption of the form of $p(y\|\boldsymbol{x})$ • The labels share the same model parameters	• Slower • More vulnerable to overfitting

algorithms are summarized in Table 3.

Further looking into the three label distribution cases reveals that the MAE of both IIS-LLD and CPNN can be ranked as: "Gaussian" $<$ "Triangle" $<$ "Single". The "Gaussian" distribution utilizes all the neighboring ages, the "Triangle" distribution utilizes those ages within the triangle, and the "Single" distribution only utilizes the chronological age. This supports the idea to use suitable label distributions to cover as many as possible correlated class labels.

In addition to the coverage of the distribution, the performance of label distribution learning may also be affected by how the related labels are covered, which is determined by the parameters of the label distributions. Fig. 6 shows the MAEs of IIS-LLD and CPNN on the FG-NET and MORPH databases with different standard deviations $\sigma = 0, 1, 2, 3, 4$ for the "Gaussian" distribution, and different bottom length $l = 2, 4, 6, 8, 10$ for the "Triangle" distribution. Note that both $\sigma = 0$ and $l = 2$ correspond to the "Single" distribution. Fig. 6 reveals that too concentrative (small σ or l) and too dispersive distributions (large σ or l) could both lead to performance deterioration. This is consistent with the intuition that the related classes are helpful but should not threaten the priority of the original class. A proper setting of the scale of the distribution is important to achieve a good performance. But generally speaking, $\sigma = 2$ and $l = 6$ are good choices in most situations.

Among the baseline methods, the AGES algorithm [11] relies on the data structure *aging pattern vector*, which is composed by all the aging faces of one person. Thus, AGES is a typical algorithm that is sensitive to the quantity of the training samples. To reveal the effectiveness of using label distribution learning to deal with the 'insufficient training data' problem, IIS-LLD and CPNN are compared in different age ranges with AGES on the FG-NET database. The results are tabulated in Table 4. The performance worse than that of the reference method AGES in the same age range are underlined. As can be seen, the number of samples in different age ranges decreases rapidly with increasing age. Samples in the higher age groups (e.g., 60-69) are especially rare. It is interesting to find that in the age ranges with relatively sufficient training data, the performance of the label distribution learning algorithms could be worse than that of AGES. For example, in the age ranges 0-9 and 10-19, all the three cases of IIS-LLD perform worse than AGES. The "Triangle" case of CPNN is also worse than AGES in the

TABLE 4
MAE in Different Age Ranges on the FG-NET Database

Range	# Samples	IIS-LLD			CPNN			AGES
		Gaussian	Triangle	Single	Gaussian	Triangle	Single	
0-9	371	2.83	2.83	3.06	2.04	2.41	2.16	2.30
10-19	339	5.21	5.17	4.99	3.38	3.30	3.55	3.83
20-29	144	6.60	6.39	6.72	5.73	6.33	6.56	8.01
30-39	79	11.62	11.66	12.10	10.51	10.71	12.62	17.91
40-49	46	12.57	15.78	18.89	14.74	14.83	15.89	25.26
50-59	15	21.73	22.27	27.40	22.00	26.33	25.73	36.40
60-69	8	24.00	26.25	32.13	25.50	28.75	31.50	45.63

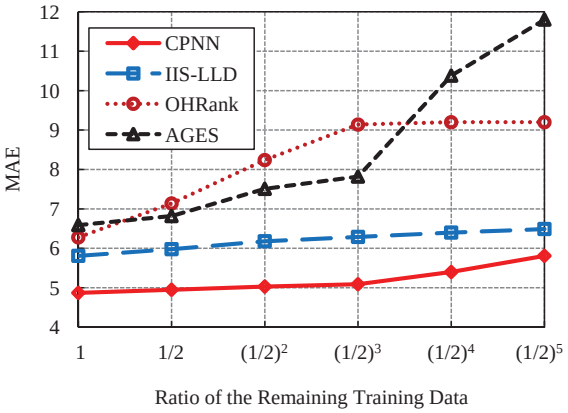

Fig. 7. MAE of IIS-LLD, CPNN, OHRank, and AGES while the MORPH training data are exponentially reduced.

age range 0-9. This is because that both IIS-LLD and CPNN are based on the general-purpose models, i.e., the maximum entropy model or the multi-layer neural network, while AGES builds on the problem-specific data structure *aging pattern vector*. The advantage of the problem-specific model generally becomes more apparent when there are sufficient training data. Another more important fact revealed by Table 4 is that the main advantage of label distribution learning comes from the classes with insufficient training samples. The less training samples there are, the more apparent the superiority of label distribution learning becomes. For example, in the age range 0-9 with maximum number of training samples, the MAE of the "Gaussian" case of CPNN is 6% lower than that of the "Single" case, and 11% lower than that of AGES, while in the age range 60-69 with minimum number of training samples, the advantages increase to 19% and 44%, respectively. This may be seen as evidence supporting the idea put forward in this paper, i.e., that label distribution learning is an effective way to relieve the 'insufficient training data' problem.

To further verify this idea, the training data from the MORPH database are gradually reduced while the test data remain the same. To make the results more obvious, the training data are exponentially reduced, i.e., half of the current training data are randomly removed each time. The MAE curves of the "Gaussian" case of the label distribution learning algorithms (IIS-LLD and CPNN) and the best two baseline methods (OHRank and AGES) are compared in Fig. 7. As can be seen, the MAEs of OHRank and AGES increase rapidly with the decrease of the training data. Insufficient training data greatly affect their performance. On the contrary, both IIS-LLD and CPNN perform relatively steadily even when the training data are exponentially reduced. The MAE of CPNN using as few as 3% ($(1/2)^5$) of the original training data is only 19% higher than that using all the MORPH training data. The performance of IIS-LLD is even better: only 12% higher MAE is observed while 97% ($1-(1/2)^5$) of the training data are removed. This illustrates the effectiveness of label distribution learning on insufficient training data. Moreover, although the MAE of IIS-LLD is generally higher than that of CPNN, it appears more steady than CPNN with the decrease of the training data. This is mainly because that IIS-LLD is based on the presumed maximum entropy model while CPNN learns the model from the training data. Consequently, IIS-LLD relies less on the training data than CPNN does.

5 Conclusion and Discussion

This paper proposes a novel approach to facial age estimation based on label distribution learning, which extends our preliminary research [13], [39]. By exchanging the single label of an instance for a label distribution, one instance can contribute to the learning of multiple classes. It is particularly useful when dealing with the problems where the classes are correlated, and the training data for some classes are insufficient. Two algorithms named IIS-LLD and CPNN are proposed in this paper to learn from such label distributions. They are tested on two aging face databases. Experimental results show the advantages of utilizing the correlated classes via label distribution learning.

While achieving good performance on the problem of facial age estimation, label distribution learning might also be useful to other problems. Generally speaking, there are at least three scenarios where label distribution learning could be helpful:

1) There is a natural measurement of description degree that associates the class labels with the instances. For example, it was found [34] that one kind of protein might be related to several kinds of cancer, and the expression levels of the protein are different in different related cancer cells. Thus, the expression level (after proper normalization) can be regarded as the description degree of the cancer to the protein.
2) When there are multiple labeling sources (e.g., multiple experts) for one instance, it is usually better for the learning algorithm to integrate the labels from all the sources rather than to decide one or more 'winning

TABLE 5
AKLD Comparison Results on the Yeast Gene Data Sets

Data Set (# labels)	kNN-LD	IIS-LLD (imp.)	CPNN (imp.)
cold (4)	.4697	.3812 (19%)	.3214 (32%)
heat (6)	.3546	.3313 (7%)	.3112 (12%)
spo (6)	.3997	.3259 (18%)	.3127 (22%)
spo5 (5)	.5616	.5042 (10%)	.4910 (13%)

label(s)' via majority voting [26]. One good way to incorporate all the labeling sources is to generate a label distribution for the instance: the label favored by more sources is given a higher description degree, while that chosen by fewer sources is assigned with a lower description degree.

3) Some classes are highly correlated with other classes (e.g., the neighboring ages). Utilizing such correlation is one of the most important approaches to improve the learning process [33], [18], [31]. Label distribution learning provides a new way toward this purpose. The key step is to transform a single-label or multi-label learning problem into a label distribution learning problem. This can be achieved by generating a label distribution for each instance according to the correlation among the classes.

The methods proposed in this paper for facial age estimation are typical examples of scenario 3). We are also working on applications in other scenarios, and the preliminary results show a favorable prospect. For example, we have applied label distribution learning to several data sets from the bioinformatics field which match scenario 1). The data sets were collected from a series of experiments (i.e., 'cold', 'heat', 'spo', and 'spo5') on the budding yeast *Saccharomyces cerevisiae* [7]. There are in total $2,465$ yeast genes included, each of which is represented by an associated phylogenetic profile of length 24. The labels correspond to the time points in different experiments. The gene expression level (after normalization) at each time point provides a natural measurement of the description degree of the corresponding label. Since there are no other label distribution learning algorithms except for the ones proposed in this paper, we extend the standard kNN algorithm to a label distribution version named kNN-LD, and use it as the baseline method. For a given instance $\boldsymbol{x}$, kNN-LD first finds its k nearest neighbors in the training set, and then calculates the mean of the label distributions of the k neighbors as the label distribution of $\boldsymbol{x}$. The performance of the algorithms are measured by the Average Kullback-Leibler Divergence (AKLD) between the predicted label distribution and the real label distribution. Table 5 lists the 10-fold cross validation results of kNN-LD, IIS-LLD, and CPNN on the four data sets. The number of labels in each data set is given in the brackets after the name of the data set. The improvements (percentage decrease of AKLD) of IIS-LLD and CPNN over kNN-LD are given in the brackets after the AKLD values. As can be seen, the more sophisticated IIS-LLD and CPNN can remarkably improve the performance of the simple extension method kNN-LD. While this is only an initial result, it reveals the exciting potentials of label distribution learning in applications other than facial age estimation. Further investigation of label distribution learning in the aforementioned three scenarios would be an interesting and promising future work.

ACKNOWLEDGMENTS

This research was supported by the National Science Foundation of China (60905031, 61273300, 61073097, 61105043, 61232007), the Jiangsu Science Foundation (BK2009269), the National Fundamental Research Program of China (2010CB327903), the Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry, the Excellent Young Teachers Program of SEU, the Open Projects Program of National Laboratory of Pattern Recognition, the Key Lab of Computer Network and Information Integration of Ministry of Education of China, and the Australian Research Council (DP0987421). The authors would like to thank Dr. Guodong Guo for providing the BIF features of the MORPH database.

REFERENCES

[1] A. L. Berger, S. D. Pietra, and V. J. D. Pietra, "A maximum entropy approach to natural language processing," *Computational Linguistics*, vol. 22, no. 1, pp. 39–71, 1996.

[2] B. Bruyer and J.-C. Scailquin, "Person recognition and ageing: The cognitive status of addresses - an empirical question," *Int'l Journal of Psychology*, vol. 29, no. 3, pp. 351–366, 1994.

[3] K.-Y. Chang, C.-S. Chen, and Y.-P. Hung, "Ordinal hyperplanes ranker with cost sensitivities for age estimation," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Colorado Springs, CO, 2011, pp. 585–592.

[4] H. Dehon and S. Brédart, "An 'other-race' effect in age estimation from faces," *Perception*, vol. 30, no. 9, pp. 1107–13, 2001.

[5] T. Denoeux and L. M. Zouhal, "Handling possibilistic labels in pattern classification using evidential reasoning," *Fuzzy Sets and Systems*, vol. 122, no. 3, pp. 409–424, 2001.

[6] G. J. Edwards, A. Lanitis, and C. J. Cootes, "Statistical face models: Improving specificity," *Image Vision Comput.*, vol. 16, no. 3, pp. 203–211, 1998.

[7] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein, "Cluster analysis and display of genome-wide expression patterns," *Proceedings of the National Academy of Science*, vol. 95, no. 25, pp. 14 863–14 868, 1998.

[8] Y. Fu, Y. Xu, and T. S. Huang, "Estimating human age by manifold analysis of face pictures and regression on aging features," in *Proc. IEEE Int'l Conf. Multimedia and Expo*, Beijing, China, 2007, pp. 1383–1386.

[9] Y. Fu, G. Guo, and T. S. Huang, "Age synthesis and estimation via faces: A survey," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 32, no. 11, pp. 1955–1976, 2010.

[10] Y. Fu and T. Huang, "Human age estimation with regression on discriminative aging manifold," *IEEE Trans. Multimedia*, vol. 10, no. 4, pp. 578–584, 2008.

[11] X. Geng, Z.-H. Zhou, and K. Smith-Miles, "Automatic age estimation based on facial aging patterns," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 29, no. 12, pp. 2234–2240, 2007.

[12] X. Geng, Z.-H. Zhou, Y. Zhang, G. Li, and H. Dai, "Learning from facial aging patterns for automatic age estimation," in *Proc. the 14th ACM Int'l Conf. Multimedia*, Santa Barbara, CA, 2006, pp. 307–316.

[13] X. Geng, K. Smith-Miles, and Z.-H. Zhou, "Facial age estimation by learning from label distributions," in *Proc. 24th AAAI Conf. Artificial Intelligence*, Atlanta, GA, 2010, pp. 451–456.

[14] G. Guo, Y. Fu, C. R. Dyer, and T. S. Huang, "Image-based human age estimation by manifold learning and locally adjusted robust regression," *IEEE Trans. Image Processing*, vol. 17, no. 7, pp. 1178–1188, 2008.

[15] G. Guo and G. Mu, "Simultaneous dimensionality reduction and human age estimation via kernel partial least squares regression," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Colorado Springs, CO, 2011, pp. 657–664.

[16] G. Guo, G. Mu, Y. Fu, and T. S. Huang, "Human age estimation using bio-inspired features," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Miami, FL, 2009, pp. 112–119.
[17] J.-S. R. Jang, "ANFIS: Adaptive-network-based fuzzy inference system," *IEEE Trans. Syst., Man, Cybern. B*, vol. 23, no. 3, pp. 665–685, 1993.
[18] F. Kang, R. Jin, and R. Sukthankar, "Correlated label propagation with application to multi-label learning," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, New York, NY, 2006, pp. 1719–1726.
[19] A. Lanitis, C. Draganova, and C. Christodoulou, "Comparing different classifiers for automatic age estimation," *IEEE Trans. Systems, Man, and Cybernetics - Part B*, vol. 34, no. 1, pp. 621–628, 2004.
[20] A. Lanitis, C. J. Taylor, and T. Cootes, "Toward automatic simulation of aging effects on face images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 4, pp. 442–455, 2002.
[21] D. S. Modha and Y. Fainman, "A learning law for density estimation," *IEEE Transactions on Neural Networks*, vol. 5, no. 3, pp. 519–523, 1994.
[22] B. Ni, Z. Song, and S. Yan, "Web image mining towards universal age estimator," in *Proc. the 17th ACM Int'l Conf. Multimedia*, Vancouver, Canada, 2009, pp. 85–94.
[23] B. Ni, Z. Song, and S. Yan, "Web image and video mining towards universal and robust age estimator," *IEEE Transactions on Multimedia*, vol. 13, no. 6, pp. 1217–1229, 2011.
[24] S. D. Pietra, V. J. D. Pietra, and J. D. Lafferty, "Inducing features of random fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 4, pp. 380–393, 1997.
[25] B. Quost and T. Denoeux, "Learning from data with uncertain labels by boosting credal classifiers," in *Proc. 1st ACM SIGKDD Workshop on Knowledge Discovery from Uncertain Data*, Paris, France, 2009, pp. 38–47.
[26] V. C. Raykar, S. Yu, L. H. Zhao, G. H. Valadez, C. Florin, L. Bogoni, and L. Moy, "Learning from crowds," *Journal of Machine Learning Research*, vol. 11, no. Apr, pp. 1297–1322, 2010.
[27] K. Ricanek and T. Tesafaye, "MORPH: A longitudinal image database of normal adult age-progression," in *Proc. 7th Int'l Conf. Automatic Face and Gesture Recognition*, Southampton, UK, 2006, pp. 341–345.
[28] M. Riedmiller and H. Braun, "A direct adaptive method for faster backpropagation learning: the RPROP algorithm," *IEEE Transactions on Neural Networks*, vol. 1, no. 3, pp. 586–591, 1993.
[29] A. Sarajedini, R. Hecht-Nielsen, and P. M. Chau, "Conditional probability density function estimation with sigmoidal neural networks," *IEEE Transactions on Neural Networks*, vol. 10, no. 2, pp. 231–238, 1999.
[30] P. Smyth, "Learning with probabilistic supervision," in *Computational Learning Theory and Natural Learning System*, T. Petsche, Ed. MA: MIT Press, 1995, vol. III, pp. 163–182.
[31] Y.-Y. Sun, Y. Zhang, and Z.-H. Zhou, "Multi-label learning with weak label," in *Proc. 24th AAAI Conf. Artificial Intelligence*, Atlanta, GA, 2010, pp. 593–598.
[32] G. Tsoumakas and I. Katakis, "Multi-label classification: An overview," *Int'l Journal of Data Warehousing and Mining*, vol. 3, no. 3, pp. 1–13, 2007.
[33] H. Wang, M. Huang, and X. Zhu, "A generative probabilistic model for multi-label classification," in *Proc. 8th IEEE Int'l Conf. Data Mining*, Pisa, Italy, 2008, pp. 628–637.
[34] J. D. Wulfkuhle, L. A. Liotta, and E. F. Petricoin, "Proteomic applications for the early detection of cancer," *Nature Reviews Cancer*, vol. 3, no. 4, pp. 267–275, 2003.
[35] S. Yan, H. Wang, T. S. Huang, Q. Yang, and X. Tang, "Ranking with uncertain labels," in *Proc. IEEE Int'l Conf. Multimedia and Expo*, Beijing, China, 2007, pp. 96–99.
[36] S. Yan, H. Wang, X. Tang, and T. S. Huang, "Learning auto-structured regressor from uncertain nonnegative labels," in *Proc. IEEE Int'l Conf. Computer Vision*, Rio de Janeiro, Brazil, 2007, pp. 1–8.
[37] S. Yan, D. Xu, B. Zhang, H. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: A general framework for dimensionality reduction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 1, pp. 40–51, 2007.
[38] S. Yan, X. Zhou, M. Liu, M. Hasegawa-Johnson, and T. S. Huang, "Regression from patch-kernel," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Anchorage, AK, 2008.
[39] C. Yin and X. Geng, "Facial age estimation by conditional probability neural network," in *Proc. Chinese Conf. Pattern Recognition*, Beijing, China, 2012, pp. 243–250.
[40] Y. Zhang and D.-Y. Yeung, "Multi-task warped gaussian process for personalized age estimation," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, San Francisco, CA, 2010, pp. 2622–2629.
[41] X. Zhuang, X. Zhou, M. Hasegawa-Johnson, and T. S. Huang, "Face age estimation using patch-based hidden markov model supervectors," in *Proc. Int'l Conf. Pattern Recognition*, Tampa, FL, 2008, pp. 1–4.
[42] H.-J. Zimmermann, Ed., *Practical Applications of Fuzzy Technologies*. Netherlands: Kluwer Academic Publishers, 1999.

Xin Geng received the B.Sc. (2001) and M.Sc. (2004) degrees in computer science from Nanjing University, China, and the Ph.D (2008) degree from Deakin University, Australia. His research interests include pattern recognition, machine learning, and computer vision. He has published over 30 refereed papers in these areas, including those published in prestigious journals and top international conferences. He has been a Guest Editor of several international journals, and served as a Program Committee Member for a number of international conferences. He is also a frequent reviewer for various international journals and conferences.

Chao Yin received the B.Sc. degree in software engineering from Anhui University, China, in 2010. He is currently a graduate student in the School of Computer Science and Engineering at Southeast University, China. His research interests include pattern recognition, machine learning, and Data Mining.

Zhi-Hua Zhou (S'00-M'01-SM'06-F'13) received the BSc, MSc and PhD degrees in computer science from Nanjing University, China, in 1996, 1998 and 2000, respectively, all with the highest honors. He joined the Department of Computer Science & Technology at Nanjing University as an assistant professor in 2001, and is currently professor and Director of the LAMDA group. His research interests are mainly in artificial intelligence, machine learning, data mining, pattern recognition and multimedia information retrieval. In these areas he has published over 90 papers in leading international journals or conference proceedings, and holds 12 patents. He has won various awards/honors including the National Science & Technology Award for Young Scholars of China, the Fok Ying Tung Young Professorship 1st-Grade Award, the Microsoft Young Professorship Award and nine international journals/conferences paper awards or competition awards. He serves/ed as an Associate Editor-in-Chief of the *Chinese Science Bulletin*, Associate Editor of the *IEEE Transactions on Knowledge and Data Engineering* and *ACM Transactions on Intelligent Systems and Technology*, and editorial boards member of more than ten other journals. He is the founder and Steering Committee Chair of ACML, and Steering Committee member of PAKDD and PRICAI. He serves/ed as General Chair/Co-chair of ACML'12, ADMA'12 and PCM13, Program Chair/Co-Chair for PAKDD'07, PRICAI'08, ACML'09 and SDM13, Workshop Chair of KDD'12, Program Vice Chair or Area Chair of various conferences, and chaired many domestic conferences in China. He is the Chair of the Machine Learning Technical Committee of the Chinese Association of Artificial Intelligence, Chair of the Artificial Intelligence & Pattern Recognition Technical Committee of the China Computer Federation, Vice Chair of the Data Mining Technical Committee of IEEE Computational Intelligence Society and the Chair of the IEEE Computer Society Nanjing Chapter. He is a fellow of the IAPR, the IEEE, and the IET/IEE. He is the corresponding author of this paper.