Self-Paced Deep Regression Forests with Consideration on Underrepresented Examples

Lili Pan[1], Shijie Ai[1], Yazhou Ren[1], and Zenglin Xu[2,3,1]

[1] University of Electronic Science and Technology of China
[2] Harbin Institute of Technology, Shenzhen, China
[3] Center for Artificial Intelligence, Peng Cheng Lab, Shenzhen, China
lilipan@uestc.edu.cn, asj1995@163.com, yazhou.ren@uestc.edu.cn, zenglin@gmail.com

Abstract. Deep discriminative models (*e.g.* deep regression forests, deep neural decision forests) have achieved remarkable success recently to solve problems such as facial age estimation and head pose estimation. Most existing methods pursue robust and unbiased solutions either through learning discriminative features, or reweighting samples. We argue what is more desirable is learning gradually to discriminate like our human beings, and hence we resort to self-paced learning (SPL). Then, a natural question arises: *can self-paced regime lead deep discriminative models to achieve more robust and less biased solutions?* To this end, this paper proposes a new deep discriminative model—self-paced deep regression forests with consideration on underrepresented examples (SPUDRFs). It tackles the fundamental ranking and selecting problem in SPL from a new perspective: fairness. This paradigm is fundamental and could be easily combined with a variety of deep discriminative models (DDMs). Extensive experiments on two computer vision tasks, *i.e.*, facial age estimation and head pose estimation, demonstrate the efficacy of SPUDRFs, where state-of-the-art performances are achieved.

Keywords: underrepresented examples, self-paced learning, entropy, deep regression forests.

1 Introduction

Deep discriminative models (*e.g.* deep regression forests, deep neural decision forests) have recently been applied to many computer vision problems with remarkable success. They compute the input to output mapping for regression or classification by virtue of deep neural networks [23,14,45,44,3,4]. In general, DDMs probably perform better when large amounts of effective training data (less noisy and balanced) is available. However, such ideal data is hard to collect, especially when large amounts of labels are required.

Computer vision literatures are filled with scenarios in which we are required to learn DDMs, not only robust to confusing and noisy examples, but also capable to conquer imbalanced data problem [48,37,5,22,24]. One typical approach is

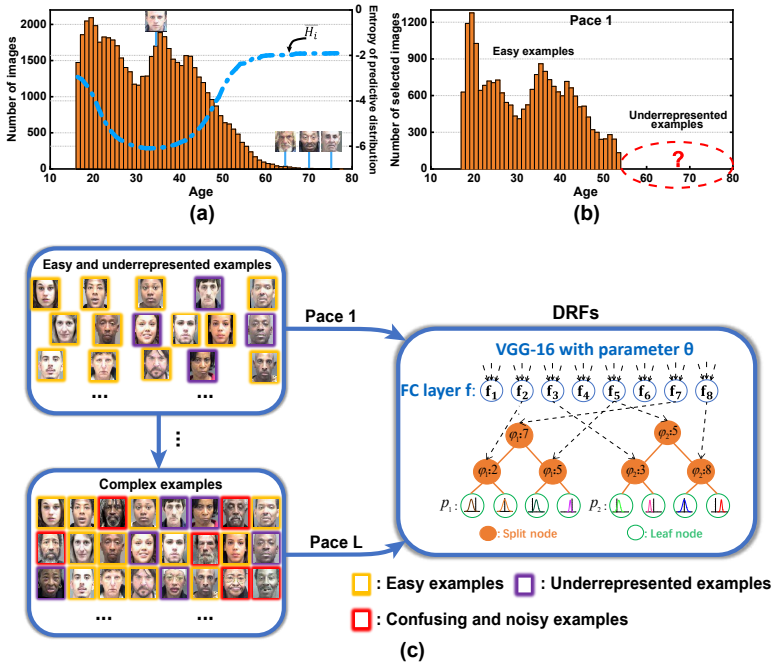


Fig. 1. The motivation of considering underrepresented examples in DRFs. **(a):** The histogram shows the number of face images at different ages, and the average entropy curve represents the predictive uncertainty. We observe the high entropy values correspond to *underrepresented samples*. **(b):** The histogram of the selected face images at pace 1 in SPL. **(c):** The proposed new self-paced learning paradigm: easy and underrepresented samples first.

to learn discriminative features through rather deep neural networks, and feed them into a *cost-sensitive* discriminative function, often with regularization [28]. The other typical approach reweights training samples according to their cost values [5,22] or gradient directions [37] (*i.e.* meta learning). These strategies are unlike our human beings, who lean things gradually—start with easy concepts and build up to complex ones, and can exclude extremely hard ones. More importantly, we have a sense of *uncertainty* for some samples (*e.g.* seldom seen) and progressively improve our capability to recognize them. Thus, the main challenge towards realistic discrimination lies how to mimic our human discrimination system might work.

This line of thinking makes us resort to self-paced learning—a gradual learning regime inspired by the manner of humans [26]. In fact, there are rare studies on the problem of self-paced DDMs. Then, a natural question arises: *can the self-paced regime lead DDMs to achieve more robust and less biased solutions?*

Motivated by this, we propose a new self-paced learning paradigm for DDMs, which tackles the fundamental ranking and selecting problem in SPL from a new

perspective: fairness. To the best of our knowledge, this is the first work considering *ranking fairness* in SPL. Specifically, we focus on deep regression forests (DRFs), a typical discriminative method, and propose self-paced deep regression forests with consideration on underrepresented examples (SPUDRFs). First, by virtue of SPL, our model distinguishes confusing and noisy examples from regular ones, and emphasizes more on "good" examples to obtain robust solutions. Second, our method considers underrepresented examples, which may incur neglect in SPL since visual data is often imbalanced, renderring less bisaed solutions. Third, we build up a new self-paced learning paradigm: ranking samples on the basis of both likelihood and entropy (predictive uncertainty), as shown in Fig. 1, which could be easily combined with a variety of DDMs.

For verification, we apply the SPUDRFs framework on two computer vision problems: (i) facial age estimation, and (ii) head pose estimation. Extensive experimental results demonstrate the efficacy of our proposed new self-paced paradigm for DDMs. Moreover, on both aforementioned problems, SPUDRFs almost achieve the state-of-the-art performances.

2 Related Work

This section reviews the deep discriminative methods for facial age estimation and head pose estimation, and SPL methods.

Facial Age Estimation. DDM based facial age estimation methods, for example [34,4,8,44,29], employ DNNs to precisely model the mapping from image to age. Ordinal-based approaches [34,4] resort to a set of sequential binary queries—each query refers to a comparison with a predefined age, to exploit the inter-relationship (ordinal information) among age labels. Improved deep label distribution learning (DLDL-v2) [8] explores the underlying age distribution patterns to effectively accommodates age ambiguity. Besides, deep regression forests (DRFs) [44] connect random forests to deep neural networks and achieve promising results. BridgeNet [29] uses local regressors to partition the data space and gating networks to provide continuity-aware weights. The final age estimation result is the mixture of the weighted regression results. Overall, these DDM based approaches have enhanced age estimation performance largely; however, they plausibly ignore one problem: the interference arising from confusing and noisy examples—facial images with PIE (*i.e.* pose, illumination and expression) variation, occlusion, misalignment and so forth.

Head Pose Estimation. For head pose estimation, Riegler [41] *et al.* utilized convolutional neural networks (CNNs) to learn patch features of facial images and achieved better performance. In [16], Huang *et al.* adopted multi-layer perceptron (MLP) networks for head pose estimation and proposed multi-modal deep regression networks to fuse RGB and depth information. In [46], Wang *et al.* proposed a deep coarse-to-fine network for head pose estimation. In [43], Ruiz *et al.* used a large synthetically expanded head pose dataset to train rather deep multi-loss CNNs for head pose estimation and gained satisfied accuracy. In [25], Kuhnke *et al.* proposed domain adaptation for head pose estimation, assuming

shared and continuous label spaces. Despite seeing much success, these methods seldom consider the potential problems caused by imbalanced and noisy training data, which may exactly exist in visual problems.

Self-Paced Learning. The SPL is a gradual learning paradigm, which builds on the intuition that, rather than considering all training samples simultaneously, the algorithm should be presented with the training data from easy to difficult, which facilitates learning [26,33]. Variants of SPL methods have been proposed recently with varying degrees of success. For example, in [21], Zhao *et al.* generalized the conventional binary (hard) weighting scheme for SPL to a more effective real valued (soft) weighting manner. In [32], Ma *et al.* proposed self-paced co-training which applies self-paced learning to multi-view or multi-modality problems. In [13], Han *et al.* made some efforts on mixture of regressions with SPL strategy, to avoid poorly conditioned linear sub-regressors. In [39,38], Ren *et al.* introduced soft weighting schemes of SPL to reduce the negative influence of outliers and noisy samples. In fact, the majority of these mentioned methods can be cast as the combination of SPL and shallow classifiers, where SVM and logistic regressors are usually involved. In computer vision, due to the remarkable performance of DNNs, some authors have realized SPL may guide DDMs to achieve more robust solutions recently. In [27], Li *et al.* sought to enhance the learning robustness of CNNs with SPL, and proposed SP-CNNs. However, [27] omits one important problem in the discriminative model: the imbalance of training data. In contrast to SP-CNNs, our SPUDRFs model has three advantages: (i) it emphasizes ranking fairness (*i.e.* considering underrepresented examples) in SPL, and hence tends to achieve less biased solutions; (ii) its learning regime is fundamental and can be easily combined with other DDMs, especially the ones with predictive uncertainty; (iii) it creatively explores how SPL can integrate with DMMs with a probabilistic interpretation.

Our work is inspired by the existing works [20,47] which take the class diversity in the sample selection of SPL into consideration. Jiang *et al.* [20] encouraged the class diversity in sample selection at the early paces of self-paced training. Yang *et al.* [47] defined a metric, named complexity of image category, to measure sample number and recognition difficult jointly, and adopted this measure for sample selection in SPL. In fact, the aforementioned two methods realize the lack of class diversity in SPL's sample selection may achieve biased solutions since visual data is often imbalanced. But what causes lack of class diversity is exactly the ranking unfairness as underrepresented examples may often have large loss (particular in DDMs). Not only that, [47,20] are only suitable for classification, but not regression (with continuous and high dimensional output). In this paper, we will go further along this direction, aiming to tackle the fundamental problem in SPL: ranking unfairness.

3 Preliminaries

In this section, we review the basic concepts of deep regression forests (DRFs) [44].

Deep Regression Tree. DRFs usually consist of a number of deep regression trees. A deep regression tree, given input-output pairs $\{\mathbf{x}_i, y_i\}_{n=1}^{N}$, where $\mathbf{x}_i \in \mathbb{R}^{D_x}$ and $y_i \in \mathbb{R}$, models the mapping from input to output through DNNs coupled with a regression tree. A regression tree $\mathcal{T}$ consists of split nodes $\mathcal{N}$ and leaf nodes $\mathcal{L}$ [44]. More specifically, each split node $n \in \mathcal{N}$ possesses a split to determine whether input $\mathbf{x}_i$ goes to the left or right subtree; each leaf node $\ell \in \mathcal{L}$ corresponds to a Gaussian distribution $p_\ell(y_i)$ with mean μ_l and variance σ_l^2.

Split Node. Split node has a split function, $s_n(\mathbf{x}_i; \boldsymbol{\Theta}) : \mathbf{x}_i \rightarrow [0, 1]$, which is parameterized by $\boldsymbol{\Theta}$—the parameters of DNNs. Conventionally, the split function is formulated as $s_n(\mathbf{x}_i; \boldsymbol{\Theta}) = \sigma\left(\mathbf{f}_{\varphi(n)}(\mathbf{x}_i; \boldsymbol{\Theta})\right)$, where $\sigma(\cdot)$ is the sigmoid function, $\varphi(\cdot)$ is an index function to specify the $\varphi(n)$-th element of $\mathbf{f}(\mathbf{x}_i; \boldsymbol{\Theta})$ in correspondence with a split node n, and $\mathbf{f}(\mathbf{x}_i; \boldsymbol{\Theta})$ denotes the learned deep features. An example to illustrate the sketch chart of the DRFs is shown in Fig. 1, where φ_1 and φ_2 are two index functions for two trees. The probability that $\mathbf{x}_i$ falls into the leaf node ℓ is given by:

$$\omega_\ell(\mathbf{x}_i|\boldsymbol{\Theta}) = \prod_{n \in \mathcal{N}} s_n(\mathbf{x}_i; \boldsymbol{\Theta})^{[\ell \in \mathcal{L}_{n_l}]} \left(1 - s_n(\mathbf{x}_i; \boldsymbol{\Theta})\right)^{[\ell \in \mathcal{L}_{n_r}]}, \tag{1}$$

where $[\mathcal{H}]$ denotes an indicator function conditioned on the argument $\mathcal{H}$. In addition, $\mathcal{L}_{n_l}$ and $\mathcal{L}_{n_r}$ correspond to the sets of leaf nodes owned by the subtrees $\mathcal{T}_{n_l}$ and $\mathcal{T}_{n_r}$ rooted at the left and right children n_l and n_r of node n, respectively.

Leaf Node. For tree $\mathcal{T}$, given $\mathbf{x}_i$, each leaf node $\ell \in \mathcal{L}$ defines a predictive distribution over age y_i, denoted by $p_\ell(y_i)$. To be specific, $p_\ell(y_i)$ is assumed to be a Gaussian distribution: $\mathcal{N}\left(y_i|\mu_l, \sigma_l^2\right)$. Thus, considering all leaf nodes, the final distribution of y_i conditioned on $\mathbf{x}_i$ is averaged by the probability of reaching each leaf:

$$p_{\mathcal{T}}(y_i|\mathbf{x}_i; \boldsymbol{\Theta}, \boldsymbol{\pi}) = \sum_{\ell \in \mathcal{L}} \omega_\ell(\mathbf{x}_i|\boldsymbol{\Theta}) p_\ell(y_i), \tag{2}$$

where $\boldsymbol{\Theta}$ and $\boldsymbol{\pi}$ represent the parameters of DNNs and the distribution parameters $\left\{\mu_l, \sigma_l^2\right\}$, respectively. It can be viewed as a mixture distribution, where $\omega_\ell(\mathbf{x}_i|\boldsymbol{\Theta})$ denotes mixing coefficients and $p_\ell(y_i)$ denotes the Gaussian distributions associated with the ℓ^{th} leaf node. Note that $\boldsymbol{\pi}$ varies along with tree $\mathcal{T}_k$, and thus we rewrite it as $\boldsymbol{\pi}_k$ below.

Forests of Regression Trees. Since a forest comprises a set of deep regression trees $\mathcal{F} = \{\mathcal{T}_1, ..., \mathcal{T}_k\}$, the predictive output distribution, given $\mathbf{x}_i$, is obtained by averaging over all trees:

$$p_{\mathcal{F}}\left(y_i|\mathbf{x}_i, \boldsymbol{\Theta}, \boldsymbol{\Pi}\right) = \frac{1}{K} \sum_{k=1}^{K} p_{\mathcal{T}_k}\left(y_i|\mathbf{x}_i, \boldsymbol{\Theta}, \boldsymbol{\pi}_k\right), \tag{3}$$

where K is the number of trees and $\boldsymbol{\Pi} = \{\boldsymbol{\pi}_1, ..., \boldsymbol{\pi}_K\}$. $p_{\mathcal{F}}\left(y_i|\mathbf{x}_i, \boldsymbol{\Theta}, \boldsymbol{\Pi}\right)$ can be viewed as the likelihood that the i^{th} sample has output y_i.

4 Self-Paced DRFs with Consideration on Underrepresented Examples

The problems in training DDMs for visual tasks arise from: (i) the noisy and confusing examples, and (ii) the imbalance of training data. Intuitively inspired by the gradual learning manner of humans, we resort to self-paced learning and explore whether the DDMs, by virtue of SPL, tend to achieve more robust solutions. Perhaps not easily, in existing SPL, we observe ranking unfairness, as shown in Fig. 1. Motivated by this observation, we propose SPUDRFs, which starts learning with easy yet underrepresented examples, and build up to complex ones. Such a paradigm avoids overlooking the "minority" of training samples, leading to less biased solutions.

4.1 Underrepresented Examples

Underrepresented examples mean "minority", as which the examples with similar or the same labels are scarce. Unsurprisingly, we observe that they may incur unfairness treatment in the early paces of SPL (see Fig. 1(b)), due to imbalanced data distribution. The underrepresented level could be measured by predictive uncertainty. Given the sample $\mathbf{x}_i$, its predictive uncertainty is formulated as the entropy of its predictive output distribution $p_{\mathcal{F}}\left(y_i|\mathbf{x}_i, \boldsymbol{\Theta}, \boldsymbol{\Pi}\right)$:

$$H\left[p_{\mathcal{F}}\left(y_i|\mathbf{x}_i, \boldsymbol{\Theta}, \boldsymbol{\Pi}\right)\right] = \frac{1}{K}\sum_{k=1}^{K} H\left[p_{\mathcal{T}_k}\left(y_i|\mathbf{x}_i, \boldsymbol{\Theta}, \boldsymbol{\pi}_k\right)\right], \tag{4}$$

where $H\left[\cdot\right]$ denotes entropy, and the entropy corresponds to the k^{th} tree is:

$$H\left[p_{\mathcal{T}_k}\left(y_i|\mathbf{x}_i, \boldsymbol{\Theta}, \boldsymbol{\pi}_k\right)\right] = -\int p_{\mathcal{T}_k}\left(y_i|\mathbf{x}_i, \boldsymbol{\Theta}, \boldsymbol{\pi}_k\right) \ln p_{\mathcal{T}_k}\left(y_i|\mathbf{x}_i, \boldsymbol{\Theta}, \boldsymbol{\pi}_k\right) dy_i, \tag{5}$$

The large the entropy is, the more uncertain the prediction should be, *i.e.*, the more underrepresented the sample is. Considering underrepresented samples can be interpreted as adequately utilizing the "information" inherent in such examples in SPL training.

As previously discussed, $p_{\mathcal{T}_k}\left(y_i|\mathbf{x}_i; \boldsymbol{\Theta}, \boldsymbol{\pi}_k\right)$ is a mixture distribution, taking the form $\sum_{\ell\in\mathcal{L}} \omega_\ell(\mathbf{x}_i|\boldsymbol{\Theta}) p_\ell(y_i)$, where $\omega_\ell(\mathbf{x}_i|\boldsymbol{\Theta})$ denotes mixing coefficients and $p_\ell(y_i)$ denotes the Gaussian distribution associated with the ℓ^{th} leaf node. In Eq. (5), the integral of mixture of Gaussians is non-trivial. Monte Carlo sampling provides a way to calculate it, but incurs large computational cost [17]. Here, we use the lower bound of this integral to approximate its true value:

$$H\left[p_{\mathcal{T}_k}\left(y_i|\mathbf{x}_i, \boldsymbol{\Theta}, \boldsymbol{\pi}_k\right)\right] \approx \frac{1}{2}\sum_{\ell\in\mathcal{L}} \omega_\ell(\mathbf{x}_i|\boldsymbol{\Theta}) \left[\ln\left(2\pi\sigma_\ell^2\right) + 1\right]. \tag{6}$$

The underrepresented examples are often scarce, and have not been treated fairly, resulting in large prediction uncertainty (*i.e.* entropy).

4.2 Objective Function

Rather than considering all the samples simultaneously, our proposed SPUDRFs are presented with the training data in a meaningful order, that is, easy and underrepresented examples first. Specifically, we define a latent variable v_i that indicates whether the i^{th} sample is selected ($v_i = 1$) or not ($v_i = 0$) depending on how easy and underrepresented it is for training. Our objective is to jointly maximize the log likelihood with respect to DRFs' parameters $\boldsymbol{\Theta}$ and $\boldsymbol{\Pi}$, and learn the latent selecting variables $\mathbf{v} = (v_1, ..., v_N)^T$. We prefer to select the underrepresented examples, which probably have higher predictive uncertainty (*i.e.* entropy), particularly in the early paces. It builds on the intuition that the underrepresented examples may incur neglect since they are the "minority" in training data. Therefore, we maximize a self-paced term regularized likelihood function, meanwhile considering predictive uncertainty,

$$\max_{\boldsymbol{\Theta},\boldsymbol{\Pi},\mathbf{v}} \sum_{i=1}^{N} v_i \left\{ \log p_{\mathcal{F}} \left(y_i | \mathbf{x}_i, \boldsymbol{\Theta}, \boldsymbol{\Pi} \right) + \gamma H_i \right\} + \lambda \sum_{i=1}^{N} v_i, \tag{7}$$

where λ is a parameter controlling the learning pace, $\lambda > 0$, γ is the parameter imposing on entropy, and H_i denotes the predictive uncertainty of the i^{th} sample, as previously discussed in Sec. 4.1. When γ decays to 0, the objective function is equivalent to the log likelihood function with respect to DRFs' parameters $\boldsymbol{\Theta}$ and $\boldsymbol{\Pi}$. Eq. (7) indicates each sample is weighted by v_i, and whether $\log p_{\mathcal{F}} \left(y_i | \mathbf{x}_i, \boldsymbol{\Theta}, \boldsymbol{\Pi} \right) + \gamma H_i > -\lambda$ determines the i^{th} sample is selected or not. That is, the sample with high likelihood value or high predictive uncertainty may be selected. The optimal v_i^* is:

$$v_i^* = \begin{cases} 1 \text{ if } \log p_{\mathcal{F}i} + \gamma H_i > -\lambda \\ 0 \text{ otherwise} \end{cases}, \tag{8}$$

where $p_{\mathcal{F}} \left(y_i | \mathbf{x}_i, \boldsymbol{\Theta}, \boldsymbol{\Pi} \right)$ is written as $p_{\mathcal{F}i}$ for simplicity.

One might argue the noisy and hard examples tend to have high predictive uncertainty also, rendering being selected in the early paces. In fact, from Eq. (8), we observe whether one sample is selected is determined by both its predictive uncertainty and the log likelihood of being predicted correctly. The noisy and hard examples probably have relatively large loss *i.e.* low log likelihood, avoiding being selected at the very start.

Iteratively increasing λ and decreasing γ, samples are dynamically involved in the training of DRFs, starting with easy and underrepresented examples and ending up with all samples. Note every time we retrain DRFs, that is, maximizing Eq. (7), our model is initialized to the result of the last iteration. As such, our model is initialized progressively by the result of the previous pace—adaptively calibrated by "good" examples. This also means we place more emphasis on easy and underrepresented examples rather than confusing and noisy ones. Thus, SPUDRFs are prone to have more robust and less biased solutions since we adequately consider the underrepresented examples.

Mixture Weighting. In the previous section, we adopt a hard weighting scheme to assign data points to paces, in which one sample is either selected ($v_i = 1$) or not ($v_i = 0$). Such a weighting scheme appears to be less accurate as it omits the importance of samples. Hence, we adopt a mixture weighting scheme [19], where the selected samples are weighted by its importance, ling in the range $0 \leq v_i \leq 1$. The objective function with mixture weighting is defined as:

$$\max_{\boldsymbol{\Theta},\boldsymbol{\Pi},\mathbf{v}} \sum_{i=1}^{N} v_i \left\{ \log p_{\mathcal{F}} \left(y_i | \mathbf{x}_i, \boldsymbol{\Theta}, \boldsymbol{\Pi} \right) + \gamma H_i \right\} + \zeta \sum_{i=1}^{N} \log \left(v_i + \zeta / \lambda \right), \tag{9}$$

where ζ is a parameter controlling the learning pace. We set $\zeta = \left(\frac{1}{\lambda'} - \frac{1}{\lambda} \right)^{-1}$, and $\lambda > \lambda' > 0$ to construct a reasonable soft weighting formulation. The self-paced regularizer in Eq. (9) is convex with respect to $v \in [0, 1]$. Then, setting the partial gradient of Eq. (9) with respect to v_i as zero will lead to the following:

$$\log p_{\mathcal{F}} \left(y_i | \mathbf{x}_i, \boldsymbol{\Theta}, \boldsymbol{\Pi} \right) + \gamma H_i + \frac{\zeta}{v_i + \zeta / \lambda} = 0. \tag{10}$$

Then, the optimal solution of v_i is given by:

$$v_i^* = \begin{cases} 1 & \text{if } \log p_{\mathcal{F}i} + \gamma H_i \geq -\lambda' \\ 0 & \text{if } \log p_{\mathcal{F}i} + \gamma H_i \leq -\lambda \\ \frac{-\zeta}{\log p_{\mathcal{F}i} + \gamma H_i} - \zeta / \lambda & \text{otherwise} \end{cases} \tag{11}$$

If either the log likelihood or the predictive uncertainty is too large, v_i^* equals to 1. In addition, if the likelihood and the predictive uncertainty are both too small, v_i^* equals to 0. Except the above two situations, the soft weighting calculation (*i.e.*, the last line of Eq. (11)) is adopted.

Curriculum Reconstruction. The underrepresented examples play an important role in our SPUDRFs algorithm. As previously mentioned, the proposed new self-paced regime coupled with a mixture weighting scheme emphasizes more on underrepresented examples, rendering better solutions. Since the intrinsic reason that causes predictive uncertainty is plausibly the imbalanced training data, we further re-balance data distribution via a curriculum reconstruction strategy. More specifically, we distinguish the underrepresented examples (whose H_i is lager than β) from regular ones at each pace, and augment them into the training data.

4.3 Optimization

We propose a two-step alternative search strategy (ASS) algorithm to solve SPUDRFs: (i) update $\mathbf{v}$ for sample selection with fixed $\boldsymbol{\Theta}$ and $\boldsymbol{\Pi}$, and (ii) update $\boldsymbol{\Theta}$ and $\boldsymbol{\Pi}$ with current fixed sample weights $\mathbf{v}$.

Optimizing $\boldsymbol{\Theta}$ and $\boldsymbol{\Pi}$. The parameters $\{\boldsymbol{\Theta}, \boldsymbol{\Pi}\}$ and weights $\mathbf{v}$ are optimized alternatively. With fixed $\mathbf{v}$, our DRFs is learned by alternatively updating $\boldsymbol{\Theta}$ and $\boldsymbol{\Pi}$. In [44], the parameters $\boldsymbol{\Theta}$ for split nodes (*i.e.* parameters for VGG) are

updated through gradient descent since the loss is differentiable with respect to $\boldsymbol{\Theta}$. While the parameters $\boldsymbol{\Pi}$ for leaf nodes are updated by virtue of variational bounding [44] when fixing $\boldsymbol{\Theta}$.

Optimizing v. As previously discussed, v_i is a binary variable or real variable ranged in $[0, 1]$. It indicates how to weight the i^{th} sample during training. The parameter λ could be initialized to obtain 50% samples to train the model, and is then progressively increased to involve 10% more data in each pace. The parameter γ could be initialized empirically and is progressively decayed to zero. The training stops when all the samples are selected, at $\gamma = 0$. Along with increasing λ and decreasing γ, DRFs are trained to be more "mature". This learning process is like how our human beings learn one thing from easy and uncertain to complex.

5 Experimental Results

5.1 Tasks and Benchmark Datasets

Age Estimation. The Morph II [40] dataset contains 55,134 unique face images of 13618 individuals with unbalanced gender and ethnicity distributions, and is the most popular publicly available real age dataset. The FG-NET [35] dataset includes 1,002 color or gray images of 82 people with each subject almost accompanied by more than 10 photos at different ages. Since all images were taken in a totally uncontrolled environment, there exists a large deviation on lighting, pose and expression (*i.e.* PIE) of faces inside the dataset.

Head Pose Estimation. The BIWI dataset [7] contains 20 subjects, of which 10 are male and 6 are female, besides, 4 males have been chosen twice with wearing glasses or not. It includes 15678 images collected by a Kinect sensor device for different persons and head poses with pitch, yaw and roll angles mainly ranging within $\pm 60°$, $\pm 75°$ and $\pm 50°$.

5.2 Experimental Setup

Dataset Setting. The settings of different datasets are given below.

- **Morph II.** Following the recent relevant work [44], the images in Morph II were divided into two sets: 80% for training and the rest 20% for testing. The random division was repeated 5 times and the reported performance was averaged over these 5 times. The VGG-Face [36] networks were chosen as the pre-trained model.
- **FG-NET.** The leave-one-person-out scheme [44] was adopted, where the images of one person were selected for testing and the remains for training. The VGG-16 networks were pre-trained on the IMDB-WIKI [42] dataset.
- **BIWI.** Similarly, 80% of the whole data was randomly chosen for training and the rest 20% for testing, and this operation was repeated 5 times. Moreover, the VGG-FACE networks were the pre-trained model.

Evaluation Metrics. The first evaluation metric is the mean absolute error (MAE), which is defined as the average absolute error between the ground truth and the predicted output: $\sum_{i=1}^{N} |\hat{y}_i - y_i| / N$, $\hat{y}_i$ represents the estimated output of the i^{th} sample, and N is the total number of testing images. The other evaluation metric is cumulative score (CS), which denotes the percentage of images sorted in the range of $[y_i - L, y_i + L]$: $CS(L) = \sum_{i=1}^{N} [|\hat{y}_i - y_i| \leq L] / N \cdot 100\%$, where $[\cdot]$ denotes an indicator function and L is the error range.

Preprocessing and Data Augmentation. On the Morph II and FG-NET datasets, MTCNN [49] was used for joint face detection and alignment. Furthermore, following [44], we augmented training images in three ways: (i) random cropping (5 times); (ii) adding Gaussian white noise with variance of 0.0001 (2 times); (iii) random horizontal flipping (2 times). The whole number of samples was increased by 20 times after augmentation. On the BIWI dataset, we utilized the depth images for training and did not augment training images.

Parameters Setting. The VGG-16 [45] was employed as the fundamental backbone networks of SPUDRFs. The hyper-parameters of VGG-16 were: training batch size (32 on Morph II and BIWI, 8 on FG-NET), drop out ratio (0.5), max iterations of each pace ($80k$ on Morph II, $20k$ on FG-NET, and $40k$ on BIWI), stochastic gradient descent (SGD), initial learning rate (0.2 on Morph II, 0.1 on BIWI, 0.02 on FG-NET) by reducing the learning rate ($\times 0.5$) per $10k$ iterations. The hyper-parameters of SPUDRFs were: tree number (5), tree depth (6), output unit number of feature learning (128), iterations to update leaf node predictions (20), number of mini-batches used to update leaf node predictions (50). In the first pace, 50% samples which are easy or underrepresented were selected for training. Here, λ was set to guarantee the first 50% samples with large $\log p_{\mathcal{F}i} + \gamma H_i$ values involved. λ' was set to ensure 10% of selected samples with soft weighting. γ was initialized to be 15 on the Morph II and BIWI datasets, and 5 on the FG-NET dataset. β was set to select 1180 and 2000 samples as the ones needed to be augmented twice at each pace on the Morph II and BIWI datasets. The number of paces was empirically set to be 10, 3 and 6 on the Morph II, FG-NET, and BIWI datasets, and except the first pace, an equal proportion of the rest data was gradually involved at each pace.

5.3 Validity of Our Proposed Method

Self-paced Learning Strategy. The validity of self-paced strategy in training DDMs is mainly demonstrated by the following experiments on the MorphII dataset. We first used all training images in the Morph II datasets to train DRFs so as to rank samples at the beginning pace. Retraining proceeded with progressively increasing λ such that every 1/9 of the rest data was gradually involved at each pace, where γ was decreased to the half of its previous value every time. In the last pace, the value γ was constrainedly set to be 0. The visualization of this process can be found in Fig. 2.

Fig. 2 illustrates the representative face images in each learning pace of SPUDRFs, along with increasing λ and decreasing γ. The two numbers below each

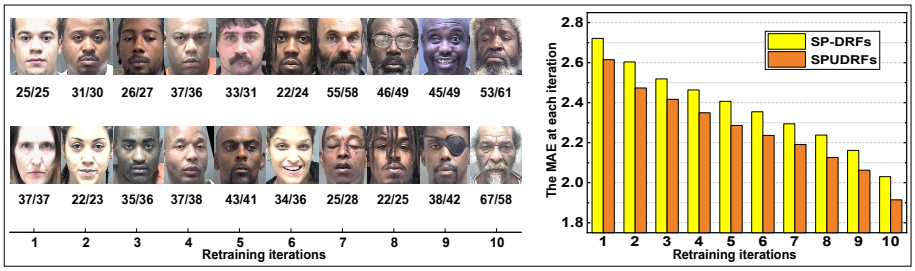


Fig. 2. The gradual learning process of SP-DRFs and SPUDRFs. **Left:** The typical worst cases at each iteration become more confusing and noisy along with iteratively increasing λ and decreasing γ. The two numbers below each image are the real age (left) and predicted age (right). **Right:** The MAEs of SP-DRFs and SPUDRFs at each pace descend gradually. The SPUDRFs show its superiority of taking predictive uncertainty into consideration, when compared with SP-DRFs.

image are the real age (left) and predicted age (right). We observe that the training images in the latter paces are obviously more confusing and noisy than the ones in the early paces. Since our model is initialized by the results of the previous retraining pace, meaning adaptively calibrated by "good" examples. As a result, it has improved performance than DRFs, where the MAE is improved from 2.17 to 1.91, and the CS is promoted from 92.79% to 93.31% (see Fig. 4(a)).

Fig. 2 also shows the comparison between SP-DRFs and SPUDRFs on the Morph II datasets. The yellow bar denotes the MAE of SP-DRFs, while the orange bar denotes for SPUDRFs. We find the MAE of SPUDRFs is lower than SP-DRFs at each pace, particularly the last pace (1.91 against 2.02). As we discussed previously, as in Fig. 1, SPUDRFs are prone to reach less biased solutions due to the wider covering range of leaf nodes, owing to considering underrepresented examples. This experiment could be regarded as an ablation study of considering ranking fairness in SPL.

Considering Underrepresented Examples. On the BIWI dataset, the necessity of considering ranking fairness in SPUDRFs is further demonstrated. In SP-DRFs, DRFs was first trained on the basis of all data, and the samples were ranked and selected for the first pace according to this result. Subsequently, every 10% of the rest samples were progressively involved for retraining. λ was progressively increased while γ was progressively decreased until zero. In SP-DRFs, the same self-paced strategy was adopted as in SPUDRFs, but without considering ranking fairness (*i.e.* underrepresented examples).

Fig. 3 visualizes the leaf node distributions of SP-DRFs and SPUDRFs in the progressive learning process. The Gaussian means μ_l associated with the 160 leaf nodes, where each 32 leaf nodes are defined for 5 trees, are plotted in each subfigures. Three paces, *i.e.* pace 1, 3, and 6, are randomly chosen for visualization. Only pitch and yaw angles are shown for clarity. Besides, the distribution of angle labels (*i.e.* pitch and yaw) are also shown, where the imbalance problem of data distribution is obvious.

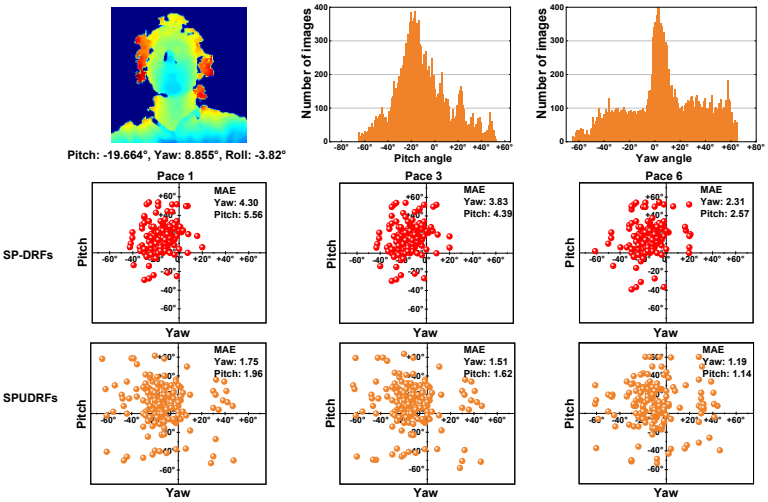


Fig. 3. The leaf node distribution of SP-DRFs and SPUDRFs in gradual learning process. Three paces, *i.e.* pace 1, 3, and 6, are randomly chosen for visualization. For SP-DRFs, the Gaussian means of leaf nodes (the red points in the second row) are concentrated in a small range, incurring seriously biased solutions. For SPUDRFs, the Gaussian means of leaf nodes (the orange points in the third row) distribute widely, leading to much better MAE performance.

In Fig. 3, the comparison results between SP-DRFs and SPUDRFs demonstrate the efficacy of considering ranking fairness in SPL. For SP-DRFs, the Gaussian means of leaf nodes (red points in the second row) are concentrated in a small range, incurring seriously biased solutions. That means the underrepresented examples have been neglected in SPL training. The poor MAEs are the evidence for this, which are even inferior to DRFs (see Fig. 6(a)). SPUDRFs rank samples by log likelihood coupled with entropy, and are prone to achieve less biased solutions, as shown in the third rows of Fig. 3. Such an experiment could be also regard as an ablation study of the proposed ranking algorithm.

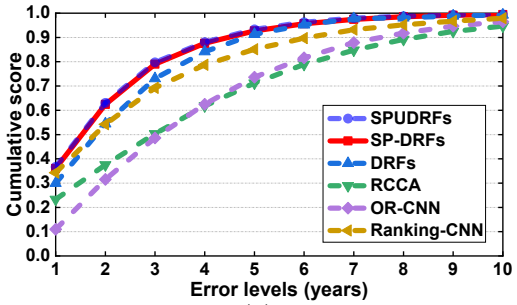
5.4 Comparison with State-of-the-art Methods

We compared our SPUDRFs with other state-of-the-art methods on the Morph II, FG-NET and BIWI datasets.

Results on Morph II. Fig. 4(a) compares SPUDRFs with other baseline methods: LSVR [11], RCCA [18], OHRank [2], OR-CNN [34], Ranking-CNN [4], DRFs [44], and DLDL-v2 [8]. Firstly, owing to the effective feature learning ability of DNNs, the SPUDRFs method is much superior to the shallow model based approaches, such as LSVR [11] and OHRank [2]. Secondly, duing to the valid self-paced regime, our SPUDRFs outperform other DDMs, and lead to more robust and less biased solutions. Thirdly, SPUDRFs outperform SP-DRFs on both

Method	MAE↓	CS↑
LSVR [11]	4.31	66.2%
RCCA [18]	4.25	71.2%
OHRank [2]	3.82	N/A
OR-CNN [34]	3.27	73.0%
Ranking-CNN [4]	2.96	85.0%
DRFs [44]	2.17	91.3%
DLDL-v2 [8]	1.97	N/A
SP-DRFs	**2.02**	**92.79%**
SPUDRFs	**1.91**	**93.31%**

(a)



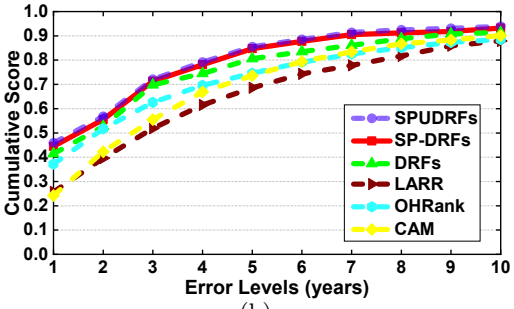
(b)

Fig. 4. The comparison results on the Morph II dataset. (a) The MAE comparison with the state-of-the-art methods, (b) the CS curves of the comparison methods.

MAE and CS, and achieve state-of-the-art performance. Fig. 4(b) shows the CS comparison on this dataset. We observe that the CS of SPUDRFs reachs 93.31% at error level $L = 5$, which is significantly better than DRFs and obtained 2.01% increment.

Method	MAE↓	CS↑
IIS-LDL [9]	5.77	N/A
LARR [10]	5.07	68.9%
MTWGP [50]	4.83	72.3%
DIF [12]	4.80	74.3%
OHRank [2]	4.48	74.4%
CAM [31]	4.12	73.5%
DRFs [44]	3.06	83.33%
SP-DRFs	**2.84**	**84.73%**
SPUDRFs	**2.77**	**85.53%**

(a)



(b)

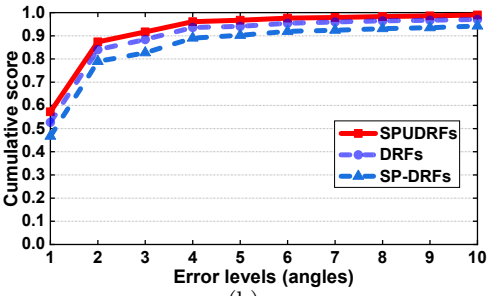
Fig. 5. The comparison results on the FGNET dataset. (a) The MAE comparison with the state-of-the-art methods, (b) the CS curves of the comparison methods.

Results on FG-NET. Fig. 5(a) shows the comparison results of SPUDRFs with the state-of-the-art approaches on FG-NET dataset. As can be seen, SPUDRFs reach an MAE of 2.77 years, which reduces the MAE of DRFs by 0.29 years. Besides, the CS comparison is shown in Fig. 5(b), SPUDRFs consistently outperform other recent proposed methods at different error levels, proving that our method is effective in enhancing the robustness of facial age estimation.

Results on BIWI. Fig. 6(a) shows the comparison results of our method with several state-of-the-art approaches. The experimental results reveal the proposed SPUDRFs method achieves the best performance with an MAE of 1.18, which is state-of-the-art. *Besides, we observe one important phenomenon: the MAE of*

Method	MAE↓
HF [41]	4.95
SVR [6]	3.14
RRF [30]	3.06
KPLS [1]	2.88
SAE [15]	1.94
MoDRN [16]	1.62
DRFs [44]	1.44
SP-DRFs	**2.08**
SPUDRFs	**1.18**

(a)



(b)

Fig. 6. The comparison results on the BIWI dataset. (a) The MAE comparison with the state-of-the-art methods, (b) the CS curves the comparison methods.

SP-DRFs is even much worse than DRFs. This further demonstrates the obvious drawback of the ranking and selecting algorithm in original SPL—incurring seriously biased solutions. In the first pace of the original SPL, as illustrated in Fig. 3, the Gaussian means of leaf nodes are concentrated in a small range, leading biased solutions. Incorporating underrepresented examples in the early pace of SPUDRFs renders to more reasonable distributions of the leaf nodes. Fig. 6(b) plots only three CS curves for brevity, *i.e.*, DRFs, SP-DRFs and SPUDRFs, which is the average of the three angles. SPUDRFs also outperform DRFs and SP-DRFs at different error levels.

6 Conclusion and Future Work

This paper explored how self-paced regime leads deep discriminative models (DDMs) to achieve more robust and less biased solutions on different computer vision tasks (*e.g.* facial age estimation and head pose estimation). Specifically, a novel self-paced paradigm, which considers ranking fairness, was proposed. The new ranking scheme jointly considers loss and predictive uncertainty. Such a paradigm was combined with deep regression forests (DRFs), and led to a new model, namely self-paced deep regression forests with consideration on underrepresented examples (SPUDRFs). Extensive experiments on two well-known computer vision tasks demonstrated the efficacy of the proposed paradigm.

We are currently applying self-paced DDMs for other computer vision tasks, *e.g.* viewpoint estimation, indoor scene classification, where the ability to handle ranking unfairness is fundamental to the success. Thus, investigating the causes of algorithm unfairness in DDMs is a worthy direction. Obviously, except data imbalance, there exist some other causing factors. In addition to this, exploring how to combine the new self-paced paradigm with other DDMs, including deep regressors and classifiers, will also be our future work.

Acknowledgement. The authors gratefully acknowledge the support of China Postdoctoral Science Foundation No.2017M623007.

References

1. Al Haj, M., Gonzalez, J., Davis, L.S.: On partial least squares in head pose estimation: How to simultaneously deal with misalignment. In: CVPR. pp. 2602–2609 (2012)
2. Chan, K., Chen, C., Hung, Y.: Ordinal hyperplanes ranker with cost sensitivities for age estimation. In: CVPR. pp. 585–592 (2011)
3. Chen, S., Zhang, C., Dong, M.: Deep age estimation: from classification to ranking. IEEE Transactions on Multimedia **20**(8), 2209–2222 (2018)
4. Chen, S., Zhang, C., Dong, M., Le, J., Rao, M.: Using ranking-CNN for age estimation. In: CVPR. pp. 742–751 (2017)
5. Cui, Y., Jia, M., Lin, T.Y., Song, Y., Belongie, S.: Class-balanced loss based on effective number of samples. In: CVPR. pp. 9268–9277 (2019)
6. Drucker, H., Burges, C.J., Kaufman, L., Smola, A.J., Vapnik, V.: Support vector regression machines. In: NeurIPS. pp. 155–161 (1997)
7. Fanelli, G., Dantone, M., Gall, J., Fossati, A., Van Gool, L.: Random forests for real time 3D face analysis. International journal of computer vision **101**(3), 437–458 (2013)
8. Gao, B., Zhou, H., Wu, J., Geng, X.: Age estimation using expectation of label distribution learning. In: IJCAI. pp. 712–718 (2018)
9. Geng, X., Yin, C., Zhou, Z.H.: Facial age estimation by learning from label distributions. IEEE Transactions on Pattern Analysis and Machine Intelligence **35**(10), 2401–2412 (2013)
10. Guo, G., Fu, Y., Dyer, C., Huang, T.: Image-based human age estimation by manifold learning and locally adjusted robust regression. IEEE Transactions on Image Processing **17**(7), 1178–1188 (2008)
11. Guo, G., Guowang, M., Fu, Y., Huang, T.S.: Human age estimation using bio-inspired features. In: CVPR. pp. 112–119 (2009)
12. Han, H., Otto, C., Liu, X., Jain, A.K.: Demographic estimation from face images: Human vs. machine performance. IEEE Transactions on Pattern Analysis and Machine Intelligence **37**(6), 1148–1161 (2015)
13. Han, L., Zhang, D., Huang, D., Chang, X., Ren, J., Luo, S., Han, J.: Self-paced mixture of regressions. In: IJCAI. pp. 1816–1822 (2017)
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016)
15. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science **313**(5786), 504–507 (2006)
16. Huang, Y., Pan, L., Zheng, Y., Xie, M.: Mixture of deep regression networks for head pose estimation. In: ICIP. pp. 4093–4097 (2018)
17. Huber, M.F., Bailey, T., Hugh, D.W., Hanebeck, U.D.: On entropy approximation for gaussian mixture random vectors. In: IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems. pp. 181–188 (2008)
18. Huerta, I., Fernèdez, C., Prati, A.: Facial age estimation through the fusion of texture and local appearance descriptors. In: ECCV. pp. 667–681 (2014)
19. Jiang, L., Meng, D., Mitamura, T., Hauptmann, A.G.: Easy samples first: self-paced reranking for zero-example multimedia search. In: ACM MM. pp. 547–556 (2014)
20. Jiang, L., Meng, D., Yu, S.I., Lan, Z., Shan, S., Hauptmann, A.: Self-paced learning with diversity. In: NeurIPS. pp. 2078–2086 (2014)

21. Jiang, L., Meng, D., Zhao, Q., Shan, S., Hauptmann, A.G.: Self-paced curriculum learning. In: AAAI. pp. 2694–2700 (2015)
22. Khan, S., Hayat, M., Zamir, S.W., Shen, J., Shao, L.: Striking the right balance with uncertainty. In: CVPR. pp. 103–112 (2019)
23. Kontschieder, P., Fiterau, M., Criminisi, A., Bulo, S.R.: Deep neural decision forests. In: ICCV (2015)
24. Kortylewski, A., Egger, B., Schneider, A., Gerig, T., Morel-Forster, A., Vetter, T.: Analyzing and reducing the damage of dataset bias to face recognition with synthetic data. In: CVPR Workshops (2019)
25. Kuhnke, F., Ostermann, J.: Deep head pose estimation using synthetic images and partial adversarial domain adaption for continuous label spaces. In: ICCV. pp. 10164–10173 (2019)
26. Kumar, M.P., Packer, B., Koller, D.: Self-paced learning for latent variable models. In: NeurIPS. pp. 1189–1197 (2010)
27. Li, H., Gong, M.: Self-paced convolutional neural networks. In: IJCAI. pp. 2110–2116 (2017)
28. Li, K., Xing, J., Su, C., Hu, W., Zhang, Y., Maybank, S.: Deep cost-sensitive and order-preserving feature learning for cross-population age estimation. In: CVPR (2018)
29. Li, W., Lu, J., Feng, J., Xu, C., Zhou, J., Tian, Q.: Bridgenet: A continuity-aware probabilistic network for age estimation. In: CVPR. pp. 1145–1154 (2019)
30. Liaw, A., Wiener, M., et al.: Classification and regression by random forest. R News **2**(3), 18–22 (2002)
31. Luu, K., Seshadri, K., Savvides, M., Bui, T.D., Suen, C.Y.: Contourlet appearance model for facial age estimation. In: ICB. pp. 1–8 (2013)
32. Ma, F., Meng, D., Xie, Q., Li, Z., Dong, X.: Self-paced co-training. In: ICML. pp. 2275–2284 (2017)
33. Meng, D., Zhao, Q., Jiang, L.: A theoretical understanding of self-paced learning. Information Sciences **414**, 319–328 (2017)
34. Niu, Z., Zhou, M., Wang, L., Gao, X., Hua, G.: Ordinal regression with multiple output CNN for age estimation. In: CVPR. pp. 4920–4928 (2016)
35. Panis, G., Lanitis, A., Tsapatsoulis, N., Cootes, T.F.: Overview of research on facial ageing using the FG-NET ageing database. IET Biometrics **5**(2), 37–46 (2016)
36. Parkhi, O.M., Vedaldi, A., Zisserman, A.: Deep face recognition (2015)
37. Ren, M., Zeng, W., Yang, B., Urtasun, R.: Learning to reweight examples for robust deep learning. arXiv:1803.09050 (2018)
38. Ren, Y., Huang, S., Zhao, P., Han, M., Xu, Z.: Self-paced and auto-weighted multi-view clustering. Neurocomputing **383**, 248–256 (2020)
39. Ren, Y., Zhao, P., Sheng, Y., Yao, D., Xu, Z.: Robust softmax regression for multi-class classification with self-paced learning. In: IJCAI. pp. 2641–2647 (2017)
40. Ricanek, K., Tesafaye, T.: Morph: A longitudinal image database of normal adult age-progression. In: FG. pp. 341–345 (2006)
41. Riegler, G., Ferstl, D., Rüther, M., Bischof, H.: Hough networks for head pose estimation and facial feature localization. International Journal of Computer Vision **101**(3), 437–458 (2013)
42. Rothe, R., Timofte, R., Van Gool, L.: Deep expectation of real and apparent age from a single image without facial landmarks. International Journal of Computer Vision **126**(2-4), 144–157 (2018)
43. Ruiz, N., Chong, E., Rehg, J.M.: Fine-grained head pose estimation without keypoints. In: CVPR workshops. pp. 2074–2083 (2018)

44. Shen, W., Guo, Y., Wang, Y., Zhao, K., Wang, B., Yuille, A.: Deep regression forests for age estimation. In: CVPR. pp. 2304–2313 (2018)
45. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv 1409.1556 (2014)
46. Wang, Y., Liang, W., Shen, J., Jia, Y., Yu, L.F.: A deep coarse-to-fine network for head pose estimation from synthetic data. Pattern Recognition **94**, 196–206 (2019)
47. Yang, J., Wu, X., Liang, J., Sun, X., Cheng, M.M., Rosin, P.L., Wang, L.: Self-paced balance learning for clinical skin disease recognition. IEEE Transactions on Neural Networks and Learning Systems (2019)
48. Zeng, X., Ding, C., Wen, Y., Tao, D.: Soft-ranking label encoding for robust facial age estimation. arXiv 1906.03625 (2019)
49. Zhang, K., Zhang, Z., Li, Z., Qiao, Y.: Joint face detection and alignment using multi-task cascaded convolutional networks. IEEE Signal Processing Letters **23**(10), 1499–1503 (2016)
50. Zhang, Y., Yeung, D.Y.: Multi-task warped gaussian process for personalized age estimation. In: CVPR. pp. 2622–2629 (2010)