Abousaleh *et al. EURASIP Journal on Image and Video Processing* (2016) 2016:47
DOI 10.1186/s13640-016-0151-4

EURASIP Journal on Image and Video Processing

RESEARCH Open Access

A novel comparative deep learning framework for facial age estimation

Fatma S. Abousaleh[1,2,3†], Tekoing Lim[2†], Wen-Huang Cheng[2], Neng-Hao Yu[3*], M. Anwar Hossain[4] and Mohammed F. Alhamid[4]

Abstract

Developing automatic facial age estimation algorithms that are comparable or even superior to the human ability in age estimation becomes an attractive yet challenging topic emerging in recent years. The conventional methods estimate one person's age directly from the given facial image. In contrast, motivated by human cognitive processes, we proposed a comparative deep learning framework, called Comparative Region Convolutional Neural Network (CRCNN), by first comparing the input face with reference faces of known age to generate a set of hints (comparative relations, i.e., the input face is younger or older than each reference). Then, an estimation stage aggregates all the hints to estimate the person's age. Our approach has several advantages: first, the age estimation task is split into several comparative stages, which is simpler than directly computing the person's age; secondly, in addition to the input face itself, side information (comparative relations) can be explicitly involved to benefit the estimation task; finally, few incorrect comparisons will not influence much the accuracy of the result, making this approach more robust than the conventional approach. To the best of our knowledge, the proposed approach is the first comparative deep learning framework for facial age estimation. Furthermore, we proposed to incorporate the Method of Auxiliary Coordinates (MAC) for training, which reduces the ill-conditioning problem of the deep network and affords an efficient and distributed optimization. In comparison to the best results from the state-of-the-art methods, the CRCNN showed a significant outperformance on all the benchmarks, with a relative improvement of 13.24% (on FG-NET), 23.20% (on MORPH), and 4.74% (IoG).

Keywords: Deep learning, Facial age estimation, Region convolutional neural network, Comparative framework

1 Introduction

With the progress of aging, the appearance of human faces exhibits changes. The facial appearance is thus a very important trait when estimating the age of a person and facial age estimation is an essential component in a number of mobile and social media applications [1–6]. However, the estimation of age by humans is usually not as easy as for determining other 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 becomes an attractive yet challenging topic emerging in recent years [7–11].

In the literature, the conventional way for facial age estimation is a direct method to estimate the age of a person by analysing his/her facial information (e.g., eyes, nose and so forth) directly from the facial image of the person, cf. Fig. 1a, c. In particular, only the input image is taken to estimate the person's age. However, telling someone's precise age at a glance without any reference information is essentially difficult even for humans [10]. In responding to the above challenges, our idea is to develop a facial age estimation algorithm inspired by human cognitive processes [12]. In practice, humans commonly use several judgements to estimate one person's age, cf. Fig. 1b. First, they learn to establish connections between a known age and the corresponding facial cues of a person (the direct method) and second, they take the learnt knowledge as reference to judge if an unseen face is younger or older than the reference (the comparative method). The larger

*Correspondence: jonesyu@nccu.edu.tw
†Equal contributors
[3]Department of Computer Science, National Chengchi University, ZhiNan Road, Taipei, 11605, Taiwan
Full list of author information is available at the end of the article

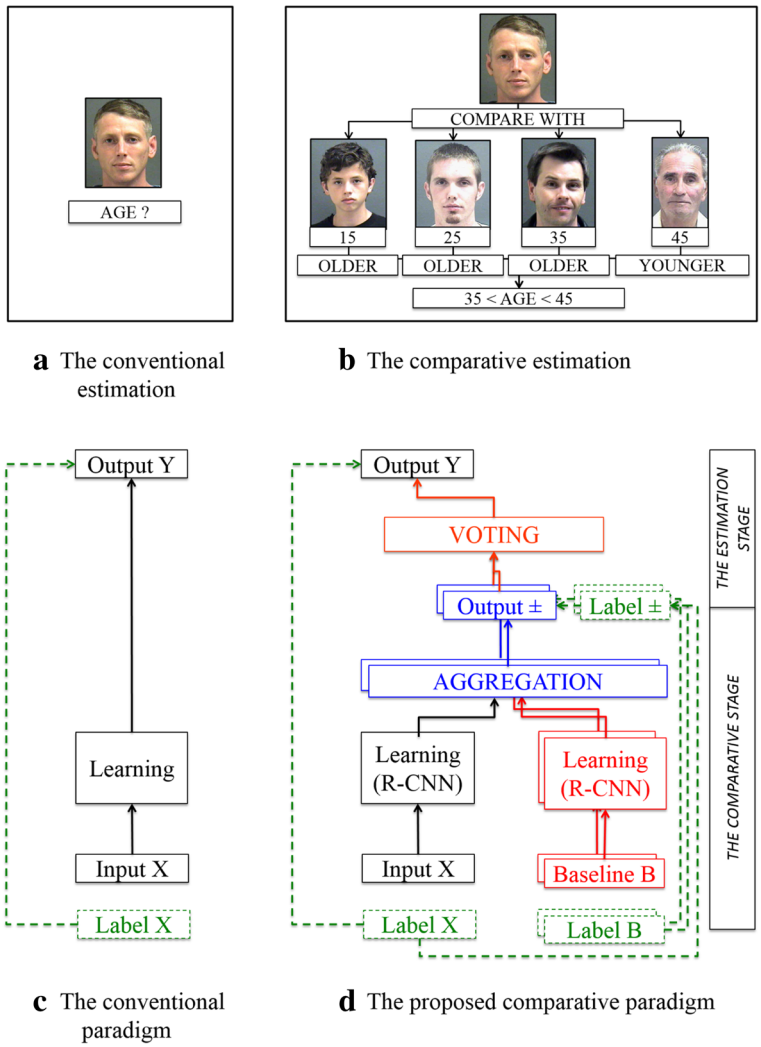

Fig. 1 Schematic diagram of **a**, **c** the conventional paradigm for facial age estimation by learning the age information from a facial image directly, and **b**,**d** the proposed paradigm by aggregating the comparisons of a facial image with baseline samples to determine the age in a comparative manner

the number of available references are, the more precise the age of an unseen face can be estimated.

Therefore, a general mathematical framework, namely *Comparative Region-Convolutional Neural Network (CRCNN)*, is proposed for facial age estimation, cf. Fig. 1d. Conceptually, we compare an unseen face with a set of selected references (labelled baseline samples) to determine if the person of the unseen face is younger or older than each of the baseline persons. We couple this comparative scheme with a specific deep learning architecture, namely Region-Convolutional Neural Network (R-CNN) [13]. The R-CNN is exploited to extract the most "iconic" local region from each facial image, where the spatial context (geometrical interrelation) of the extracted local regions can be also accounted for robust classification. In the proposed CRCNN framework, not only the input image is used, but also several other reference images are taken as baseline samples to be compared with the input. The comparison is equivalent to estimate if the input person is younger or older than the other ones. In comparison to the conventional paradigm, the first advantage of this approach is to reformulate the estimation task into sequentially independent sub-problems. Each sub-problem represents a comparison (younger/older decision) between two images, which is much simpler than the initial task, i.e., guessing the exact age of an observed face. The second advantage is, by simply increasing the number of baseline samples, more side information (comparisons) can be exploited to benefit the estimation task, leading to a more robust estimation. Last but not least, one more advantage by leveraging many baseline samples is that few incorrect comparisons will not influence much the accuracy of the age estimation.

Further, the traditional way to learn the parameters of a deep architecture is to minimize an objective function by computing the gradient over all the parameters using the backpropagation algorithm [14] with a nonlinear optimizer. However, the deep learning method has been observed to be very difficult to train especially due to the ill-conditioning problem and local minima issue [15]. These difficulties also complicate the manual tuning of deep learning parameters as well as the convergence. In this work, we propose to incorporate the recent Method of Auxiliary Coordinates (MAC) [16] into our framework for training, which appears to open an interesting door toward more efficient training of deep architecture. The method introduces a set of variables to break the objective function dependency, which makes the problem much better conditioned without nesting, affording an efficient and distributed optimization.

Our main contributions are multifold: first, to the best of our knowledge, our CRCNN framework is the first comparative deep learning approach for facial age estimation and has demonstrated its outperformance over the state-of-the-art methods by experimenting with well-known face datasets. In addition, instead of using the classical deep learning techniques, e.g., Convolutional Neural Network (CNN) [17], we proposed the use of R-CNN to account for the spatial context of facial regions; secondly, we improved the training efficiency of deep architecture by incorporating the MAC techinique. The notorious ill-conditioning problem of deep learning can be alleviated; thirdly, we implemented our mathematical framework with CAFFE [18], a popular deep learning platform which exploits the parallelization over multiple GPUs. The compatibility with CAFFE makes all the components of our mathematical implementation readily available to be used by other researchers; fourthly, observing the fact that the sensitivity of deep learning parameters makes it a non-trivial task to obtain an appropriate setting, the systematic investigation on parametric optimization provides a guidance to users who would extend our approach for their future researches.

This paper is organized as follows. Section 2 describes the related work. Section 3 presents our algorithm, and Section 4 gives experimental results to demonstrate the optimization and the various advantages of our approach. Section 5 draws the conclusions and gives directions for future work.

2 Related work

Many researchers have developed techniques for facial age estimation. Most of the previous works focus on the extraction and fusion of different types of facial features: the extraction of local features by using various methods [9]; the combination of hybrid features (e.g., Gabor filters and local binary patterns) by using hierarchical classifiers based on support vector machines (SVMs) and support vector regression (SVR) [8, 19]; the fusion of textural and local appearance based descriptors to achieve faster and more accurate results [20]; the use of canonical correlation analysis (CCA) for jointly estimating the age with other facial information like gender [21]. Recently, the deep learning has been applied for facial age estimation, e.g., a multilayered neural network is integrated with the adapted retinal sampling mechanism [22]; the convolutional neural network based methods [23, 24] have been studied as well; a constructive probabilistic neural network based on learning from label distributions was also presented [10]. In summary, the previous works all followed the conventional paradigm, i.e., learning direct mappings between the extracted facial features and the associated age labels. These observations motivated the development of our comparative approach with the deep learning method.

Motivated by human cognitive processes [12], a more robust way to estimate a facial age is arguably to be in a comparative manner, i.e., learning from a number of comparative relations (a given face is younger or older than another face of known age). The development of our approach was also inspired by other ranking-based approaches, such as Ranking SVM [25], RankBoost [26], and RankNet [27]. Ranking SVM [25] formalizes the learning to rank as a problem of classifying instance pairs into two categories (correctly ranked and incorrectly ranked). Experimental results from this approach showed that the algorithm performs well in practice, successfully adapting the retrieval function of a meta-search engine to the preferences of a group of users. However, the losses (penalties) of incorrect ranking between higher ranks and lower ranks and incorrect ranking among lower ranks are defined the same. This remark will cause troubles for facial age estimation as the youngest and oldest persons provide totally different facial information. RankBoost [26] is another ranking algorithm that is trained on pairs, which is close in spirit to our work since it attempts to solve the preference learning problem directly, rather than solving an ordinal regression problem. Results are given using decision stumps as the weak learners. RankNet [27] is simple to train and gives good performance on a real world ranking problem with large amounts of data. RankNet explored the use of a neural network formulation. A probabilistic cost for training systems is also proposed to learn ranking functions using pairs of training examples. In this paper, we propose a novel ranking approach through our comparative framework for facial age estimation. First, a set of selected references, i.e., baseline samples, is introduced into the framework to make each rank more robust. Secondly, our age estimation model will be generated with the deep learning technique, providing efficient features to rank each age from facial information. Finally,

the younger/older comparison will provide robust ranking by leaning similar facial information to estimate similar ranks, thus the ranking will be better structured.

3 The proposed method: a CRCNN framework

The proposed *Comparative Region-Convolutional Neural Network (CRCNN)*, a general mathematical framework for facial age estimation, is developed by comparing an input face with a number of baseline samples to determine its age. We compare the input face with each baseline sample and determine if the input face is older or younger than the baseline person. A set of hints (comparative relations) is therefore collected. The estimation stage aggregates the set of hints to obtain the age of the input person. In this section, we first explain some preliminary definitions (Section 3.1). Then we give an overview of our CRCNN framework (Section 3.2). Finally each algorithmic component in our approach is explained in details (Section 3.3).

3.1 Preliminary definitions

Before explaining our CRCNN framework, we first define two terminologies: the baseline and the set of hints.

a) Baseline: The objective is to compare the age of an input image with those of a set of reference images, where the ages of these references are known. We define these references as the baseline. A baseline is composed by a set of reference samples, as many as possible to thoroughly cover the value range of possible ages (e.g., labels). In other words, each baseline sample represents an age label. In a minimum, we take one baseline sample per label, therefore, if we have M labels, then we have M baseline samples in total. And if we have K baseline samples per label, we will have totally MK baseline samples.

b) Set of hints: To understand the exploitation of the set of hints, we follow the example in Fig. 2. To estimate the age of an input $\mathbf{X}$ (the ground-truth age is 62), we first compare the input with the baseline samples $\mathbf{B} = \{\mathbf{B}_1, \ldots, \mathbf{B}_5\}$. A hint can be in two categorical types: "younger" or "older".[1] For each baseline sample, if the age of the input is estimated to be larger than the age of the baseline sample (i.e., the input person is estimated to be "older" than the baseline one), we add a hint for the corresponding label of every baseline sample with its age larger than (or equal to) the comparing one. For example, we consider the comparison between $\mathbf{X}$ and $\mathbf{B}_2$. Since $\mathbf{X}$ is older than $\mathbf{B}_2$, we thus add a hint for the labels of $\mathbf{B}_2$, $\mathbf{B}_3$, $\mathbf{B}_4$, and $\mathbf{B}_5$ to indicate that they are all possible labels for $\mathbf{X}$. Similarly, if the input person is estimated to be "younger" than the baseline person, then we add a hint for the corresponding label of every baseline sample with its age smaller than (or equal to) the comparing one. In this way, the obtained hints of each label in number is proportional to the likelihood that a label is the true label to the input, e.g., in Fig. 2, $\mathbf{B}_4$ is the most likely label to $\mathbf{X}$ and $\mathbf{B}_1$ is the most unlikely one.

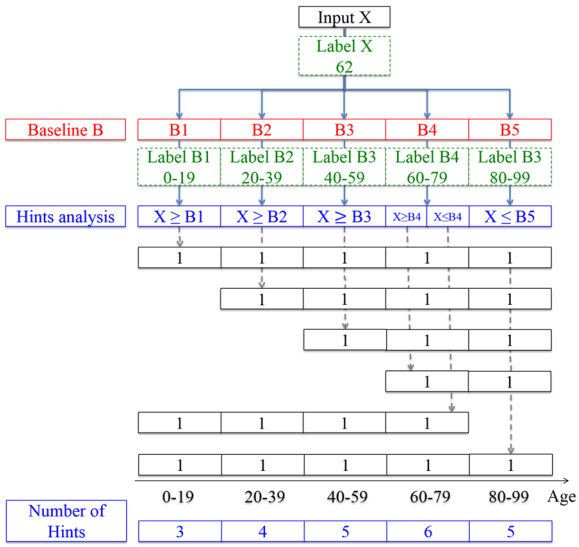

Fig. 2 Generation of a set of the hints (for simplicity, five labels are presented)

3.2 An overview of our CRCNN framework

Our CRCNN framework can be decomposed into two main stages, as presented in Fig. 1(d):

3.2.1 The comparative stage (collecting the hints)

After building up a baseline, the input image is compared with each of the baseline samples. We use the R-CNN deep architecture to extract facial information from the images and then apply an energy function-based aggregation to generate the comparisons (Section 3.3.1). Therefore, a set of hints is collected. Each hint represents a comparative relation (younger or older) which provides information to compute the estimated age at the next stage.

3.2.2 The estimation stage (voting the hints)

This stage votes by the results from the set of hints to compute the estimated age (Section 3.3.2).

3.3 The CRCNN formulations

Considering $\mathcal{I}$ as a universal set of facial images and $\mathcal{L}$ be the corresponding label set of possible ages of a human being, we are given a training set of N facial images $\mathbf{X} \in \mathcal{I}$ and its label $\mathbf{Y} \in \mathcal{L}$. Let F denotes the deep architecture function. Instead of computing $\mathbf{Y}$ with F as usual in the conventional paradigm:

$$\begin{array}{rcl} F : \mathcal{I} & \rightarrow & \mathcal{L} \\ \mathbf{X} & \mapsto & \mathbf{Y} = F(\mathbf{X}). \end{array} \quad (1)$$

The idea is to introduce a baseline $\mathbf{B} = \{\mathbf{B}_1, \ldots, \mathbf{B}_M\}$ from $\mathcal{I}$ with a composition function Ψ and Φ in order to decompose the task into two main parts. Note that $\mathbf{X}$ and $\mathbf{B}$ are usually disjoint. First, in the comparative stage, the comparison of $\mathbf{X}$ and the baseline $\mathbf{B}$ with Ψ provides the set of hints $\mathcal{H}$ (Section 3.3.1). Second, in the estimation stage, the vote of hints from the set of hints $\mathcal{H}$ is to obtain the final label $\mathcal{L}$ with Φ (Section 3.3.2). Therefore, the proposed CRCNN approach is formulated as follows:

$$\begin{array}{ccccccc} (\mathcal{I}\times\mathcal{I}) & \overset{\Psi}{\rightarrow} & \mathcal{H} & & \overset{\Phi}{\rightarrow} & & \mathcal{L} \\ (\mathbf{X},\mathbf{B}) & \mapsto & \mathbf{Z}=\Psi(\mathbf{X},\mathbf{B}) & & \mapsto & & \mathbf{Y}=\Phi(\mathbf{Z}). \end{array}$$

3.3.1 The comparative stage

The set of hints $\mathbf{Z} \in \mathcal{H}$ is computed from $\mathbf{X} \in \mathcal{I}$ and $\mathbf{B} \in \mathcal{I}$ with the function Ψ which is decomposed into:

$$\Psi = \Psi^R \circ \Psi^C \circ \Psi^L \circ \Psi^F \circ \Psi^A.$$

The first operator Ψ^R detects all the regions where the facial information is selected by R-CNN to be the most relevant. The second operator Ψ^C is the convolutional step (including sub-sampling layers) that extracts a fixed-length feature vector from each region. The third and fourth operators (Ψ^L and Ψ^F) are the locally and fully-connected steps [17]. Finally, the features of both the input image and baseline samples are aggregated into the last operator Ψ^A where an energy function approximates the age comparison with a distance metric.

Region-detection layer: Consider $\mathbf{X}_i \in \mathcal{I}$, an input image, a set of candidate regions $\{X_{i,j}\}_{j=1\ldots J}$ is detected from $\mathbf{X}_i$ in order to extract more efficient facial information features. Each region $X_{i,j}$ is detected by the algorithm in [13]. The same region-detection operator Ψ^R is applied to each baseline sample $\mathbf{B}_m$ providing a set of candidate regions $\{B_{m,j'}\}_{j'=1\ldots J'}$. Therefore, we denote by $\mathbf{H}_1$ the first hidden layer of our deep architecture, formed with the region-detection layer. Notice that, if no region detection is used (Ψ^R is equivalent to an identical function), then we set the output as the input image itself ($\{X_i\} = \{\mathbf{X}_i\}$).

Convolutional layers: The convolutional operator Ψ^C extracts features from the first hidden layer $\mathbf{H}_1$. Specifically, features are computed by forward propagating through a convolutional structure of $|C|$ layers with

$$\Psi^C = \Psi_1^C \circ \Psi_2^C \circ \cdots \circ \Psi_{|C|}^C.$$

These steps expand the input into a set of simple local features. We denote $\mathbf{H}_k = \Psi_k^C(\mathbf{H}_{k-1})$ as the output of a convolutional layer for $k = 2, 3, \ldots, |C| + 1$. More details of the convolutional layer can be referred to [17]. We interpret these convolutional steps as an adaptive preprocessing step. The purpose of these convolutional steps is to extract low-level features, like simple edges and textures. Notice that the sub-sampling layers make the output of convolution networks more robust to local translations and small registrational errors, which is important in facial recognition problem.

Locally-connected layers: After extracting features with Ψ^C, applied independently to $\mathbf{X}_i$ and $\mathbf{B}_m$, we first combine locally extracted features through $|L|$ locally-connected layers with

$$\Psi^L = \Psi_1^L \circ \Psi_2^L \circ \cdots \circ \Psi_{|L|}^L,$$

resulting to $\mathbf{H}_k = \Psi_k^L(\mathbf{H}_{k-1})$ for $k = |C| + 2, |C| + 3, \ldots, |C|+|L|+1$. Like in the convolutional deep learning, the locally-connected layers apply a filter bank, but every location in the feature map learns a different set of filters. For example, information from an area between the eyes and the eyebrows will be combined with the one between the nose and the mouth, but the two pieces of information will be processed differently in the convolutional operation.

Fully-connected layers: Then, the fully-connected operation Ψ^F computes all the weights together with

$$\Psi^F = \Psi_1^F \circ \Psi_2^F \circ \cdots \circ \Psi_{|F|}^F$$

and $\mathbf{H}_k = \Psi_k^F(\mathbf{H}_{k-1})$ for $k = |C| + |L| + 2, |C| + |L| + 3, \ldots, |C| + |L| + |F| + 1$. Unlike in the locally-connected operation where the inputs are locally combined, each output unit in the fully connected layers is connected to all inputs. These layers are able to capture correlations between features captured in distant parts of the face images, e.g., the position and shape of eyes and the position and shape of mouths.

Aggregation: An EBM energy function [28] is exploited to aggregate both information of $\mathbf{X}_i$ and $\mathbf{B}_m$ from the fully-connected operation in order to estimate if $\mathbf{X}_i$ is younger or older than $\mathbf{B}_m$. The advantage of the adopted energy function is that there is no need for estimating normalized probability distributions over the input space. The scalar energy function E measures the compatibility between $\mathbf{X}_i$ and $\mathbf{B}_m$ and leads to a set of hints associated with the in-between comparative relation, cf. Fig. 2. This real-valued energy function is thus defined as $E(\mathbf{X}_i, \mathbf{B}_m) = ||G_W(\mathbf{X}_i) - G_W(\mathbf{B}_m)||$, where G_W is a mapping (subject to learning) to produce output vectors that are nearby for images from the same person, and far away for images from different persons [28].

Learning is then performed by finding the deep architecture parameters that minimize a suitably designed loss function, evaluated over a training set. Consider L^- (or L^+) the partial loss function if $\mathbf{X}_i$ is younger (or older) than $\mathbf{B}_m$, our loss function is of the form

$$L = \left(1 - \bar{\mathbf{Z}}_l\right) L^- \left(E\left(\mathbf{X}_i, \mathbf{B}_m\right)\right) + \left(\bar{\mathbf{Z}}_l\right) L^+ \left(E\left(\mathbf{X}_i, \mathbf{B}_m\right)\right),$$

where $\bar{\mathbf{Z}}_l$ is the ground truth of the hint $\mathbf{Z}_l$. The partial loss function L^- (or L^+) is designed in such a way that the minimization of L will decrease (or increase) the energy when $\mathbf{X}$ is younger (or older) than $\mathbf{B}_i$. A simple way to achieve that is to make L^- monotonically decreasing, and L^+ monotonically increasing.

3.3.2 The estimation stage

Once the set of hints have been generated, the estimation stage is applied to vote by the output information of the previous comparative stage in order to estimate the person's age. The representation of the set of hints in Fig. 2 includes the number of hints for each label. This result is computed by applying a summation at each label. Therefore, the age of the input person could be estimated by taking the label with the most votes in a naive way. In practice, to avoid the case where the most votes appears in more than one label, we choose to use the real value outputted from the energy function E instead of the number of hints $\mathbf{Z}_i$, since the confidence of a vote is also embedded. That is, a larger value indicates the higher confidence of a vote, and vice versa.

3.4 Learning method for the comparative stage

In this work, we propose to incorporate the recent Method of Auxiliary Coordinates (MAC) [16] for training the comparative stage. The MAC method decouples the typical learning problem of the comparative stage, which typically has an objective function in the form

$$\min \parallel \mathbf{Z} - \Psi\left(\mathbf{X}, \mathbf{B}\right) \parallel^2$$

into the following one:

$$\begin{aligned} &\min \parallel \mathbf{H}_{k+1} - \Psi_{k+1}\left(\mathbf{H}_k\right) \parallel^2 \\ &\min \quad \parallel \mathbf{Z} - \Psi_K\left(\mathbf{H}_K\right) \parallel^2 \end{aligned}$$

for $k = 1, 2, \ldots, |C| + |L| + |F|$ and $K = |C| + |L| + |F| + 1$. Note that the MAC is applied only to the convolutional, locally- and fully-connected layers, such that $\Psi_k \in \{\Psi_k^C, \Psi_k^L, \Psi_k^F\}$. The problem becomes a set of small, independent minimization subproblems, each of which can be easily solved, and without back-propagating any gradients. The objective function is optimized over the hidden layer $\mathbf{H}$ and over the weights $\mathbf{W}$ (of the function Ψ) with the two functions below alternatively:

$$\mathbf{H}_{k-1} \xrightarrow{\Psi_{k-1}} \mathbf{H}_k \xrightarrow{\Psi_k} \mathbf{H}_{k+1}$$

$$\mathbf{H}_k \xrightarrow{\Psi_k} \mathbf{H}_{k+1}$$

Specifically, optimizing the objective function over the hidden layer $\mathbf{H}_k$ means optimizing the following nonlinear, least-squares regression of the form

$$\min_{\mathbf{H}_k} \parallel \mathbf{H}_k - \Psi_{k-1}(\mathbf{H}_{k-1}) \parallel^2 + \parallel \mathbf{H}_{k+1} - \Psi_k(\mathbf{H}_k) \parallel^2$$

and alternatively, optimizing over the weight $\mathbf{W}^k$ (of the function Ψ_k) with

$$\min_{\mathbf{W}^k} \parallel \mathbf{H}_{k+1} - \Psi_k\left(\mathbf{W}^k, \mathbf{H}_k\right) \parallel^2 .$$

Notice that optimizing over the hidden layer $\mathbf{H}_k$ has fixed weights $\mathbf{W}^k$ and optimizing over the weight $\mathbf{W}^k$ has fixed hidden layer $\mathbf{H}_k$. This minimization problem results in several independent, single-layer single-unit problems that can be solved with existing algorithms, without extra programming cost. We solve this nonlinear least-squares fitting problem with a Gauss-Newton approach [29].

4 Experimental results and discussions

In this section, we present the results from a series of experiments designed to optimize and to test the effectiveness of our CRCNN framework. We implemented our experiments using CAFFE in a machine with Intel CPU duo-cores (at 3.40 GHz). Firstly, we present the general setting of our experiments. Secondly, we optimize the setting (i.e., try our best to search for the best setting empirically) of our CRCNN approach. Finally, we compare our CRCNN approach with the state-of-the-art methods in facial age estimation.

4.1 Experimental setup

4.1.1 Datasets

We used three public datasets in the experiments and they are also common benchmarks adopted in the related literature [10, 21, 30, 31]. The first one is the FG-NET Aging Database [32]. There are 1002 face images from 82 subjects in this database. Each subject has 6-18 face images at different ages. Each image is labelled by its real age. The ages are distributed in a wide range from 0 to 69. The dataset images exhibit large facial variations, such as significant changes in pose, illumination, expression, etc. The second dataset is the MORPH Database [33]. 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. Finally, the last one is the Images of Groups (IoG) dataset [34]. The dataset consists of 5080 images with a total of 28,231 labeled faces. The images were acquired through searches on the photo-sharing website Flickr, and each face is assigned to one of seven age groups: 0–2, 3–7, 8–12, 13–19, 20–36, 37–65, and 66+. As the images were collected from searches, there is an extremely uneven distribution of images across age and pose.

4.1.2 Implementation platform

CAFFE [18] is a BSD-licensed C++ library with Python and MATLAB bindings for training and deploying general purpose convolutional neural networks and other deep models efficiently on commodity architectures. It is now a very popular deep learning platform and we chose to implement our CRCNN framework based on it to give high extendibility for future practitioners to integrate their own implementations with our CRCNN framework.

4.1.3 Early and late fusion schemes

We perform our mathematical comparative method with two different schemes: the early fusion and the late fusion [35]. The framework described in this paper first adopts the late fusion scheme, i.e., we extract features from the input image and each baseline sample separately and then fully connect all the information into a final layer of the deep architecture. Alternately, the early fusion scheme first combines the input image with the baseline samples and then extracts information from the both type of images together in the same time. Both fusion schemes will be optimized, tested, and compared to the state-of-the-art results.

4.2 Optimization of our CRCNN framework

In this section, we present the optimization of our deep architecture. The purpose is to provide insights on the sensitivity of the parameters associated with our CRCNN framework. First, the performance of the comparative stage with different settings of the deep architecture's parameters (e.g., fusion strategy, baseline, region detection, etc.) is presented in Fig. 3. Each sub-figure represents the performance of a parameter when in different values (or choices). The empirically optimal values of our CRCNN parameters from the experiments are summarized in Table 1. Secondly, the sensitivity between

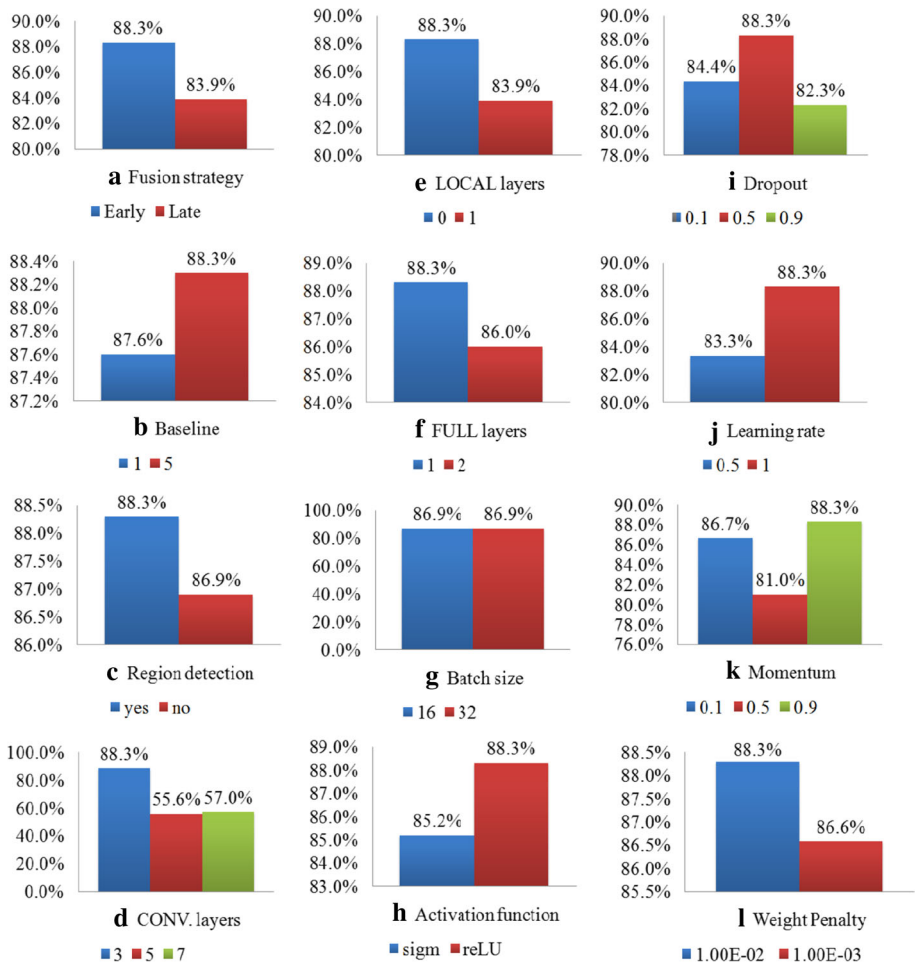

Fig. 3 Optimization of our CRCNN approach: performance by different setting of the deep architecture's parameters. **a** Fusion strategy. **b** Baseline. **c** Region detection. **d** CONV. layers. **e** LOCAL layers. **f** FULL layers. **g** Batch size. **h** Activation function. **i** Dropout. **j** Learning rate. **k** Momentum. **l** Weight Penalty

Table 1 The optimized setting of our CRCNN method

Deep architecture's parameters	Optimized value
Fusion	Early
Number of baseline samples	5
Region detection	Yes
Number of convolutional layers	3
Number of locally-connected layers	0
Number of fully-connected layers	1
Batch size	32
Activation function	reLU
Dropout	0.5
Learning rate	1
Momentum	0.9
Weight penalty	1e-2

parameters is presented in Fig. 4. Each sub-figure represents the correlation coefficient of a parameter and the others based on the obtained performances (of the comparative stage). The lower the correlation coefficient is (close to 0), the more independent the two parameters are; the higher the correlation coefficient is (close to 1), the more dependency between them on the performance of the comparative stage will be. For example, in Fig. 4g, the correlation coefficient of BS (batch size) and D (dropout) is less than 0.5 (weakly related) and the correlation coefficient of BS and itself is naturally 1 (perfectly related). Note that raw image pixels are taken as the extracted features.

4.2.1 CRCNN parameters:

Fusion strategy (F): The early and the late fusion are different in the way of sharing weights. In the early fusion, both types of images (the input one and the baseline ones) share the same set of weights, and in the late fusion, each image has its own weight. As can be seen in Fig. 3a, the first value (88.3%) represents the accuracy when the early fusion is applied to our CRCNN framework, and the second value (83.9%) represents the accuracy when the late fusion is applied. In other words, Fig. 3a shows a better accuracy when the early fusion is applied. This observation intuitively corresponds to the fact that learning shared weights improves the inner relation between the input image and the baseline. We observe in Fig. 4a that the optimization of each fusion strategy depends on the whole deep architecture (i.e., convolutional layers, locally connected layers, and fully connected layers) and the value of dropout.

Baseline (B): Each baseline sample is taken as a reference to represent a range of possible ages (e.g., labels). In our optimization, we take M baseline samples per label, with $M = 1, 5$. As expected and observed in Fig. 3b, a more robust computation is provided when $M > 1$ baseline sample to represent each label. Correlations exist between this parameter and the region detection, and also with several deep learning parameters, such as the momentum and the weight penalty (Fig. 4b).

Region detection (R): We optimized our method with and without the region detection. In other words, this optimization is equivalent to optimize our CRCNN method by combining the R-CNN [13] or the classical CNN [17]. Figure 3c shows the results of this optimization and it is clear that region detection Ψ^R can extract more robust features for improving the performance. The performance of applying this detection depends on the setting of its input (e.g., baseline) and output (e.g., convolutional layers) as observed in Fig. 4c.

Convolutional layers (CL): We optimized the convolutional layers Ψ^C relating to the influence of the number of layers. Several numbers of layers have been experimented and the results are shown in Fig. 3d. We observe that three convolutional layers provide the best results and the number of layer is logically correlates with its previous and following layers (the region detector and the locally-connected layer Ψ^C), also with the value of dropout and as mentioned previously, the early/late fusion choice (Fig. 4d).

Locally-connected layers (LL): We optimized the locally-connected layers Ψ^L. Figure 3e shows the results for different numbers of layers. The most accurate result is provided when the convolutional layer Ψ^C is directly connected with the fully-connected layer Ψ^F. Its influence between other parameters is the same as the convolutional layers (Fig. 4e).

Fully-connected layers (FL): The optimization of the fully-connected layers Ψ^F is shown in Fig. 3f. We observe that only one fully-connected layers is enough to provide the best results. Notice that the optimization of the number of fully connected layer can be set independently (Fig. 4f).

Batch size (BS): The "batch" learning accumulates contributions for all data points, then updates the parameters. We use the "mini-batches" learning [36], where the parameters are updated after every n data points (i.e., this approach divides the dataset into piles and learns each pile separately). The computation time of learning the deep architecture depends on the number of epoches and the size of batches. Fig. 3g shows two different sizes of batches. Empirically, we take `batchsize = 32` and the batch size can be optimized independently (Fig. 4g).

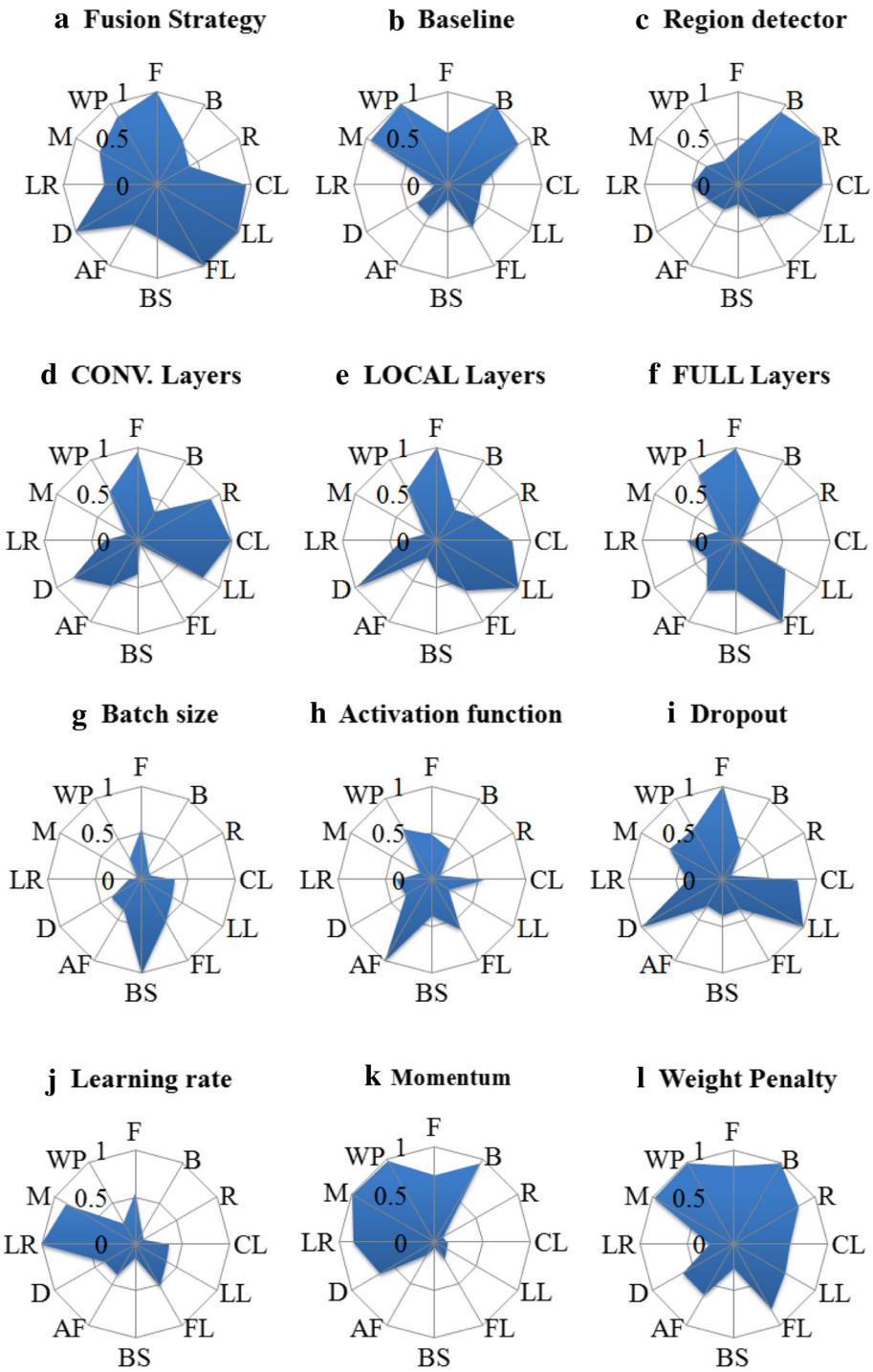

Fig. 4 Optimization of our CRCNN approach: sensitivity of the deep architecture's parameters. **a** Fusion Strategy. **b** Baseline. **c** Region detector. **d** CONV. Layers. **e** LOCAL Layers. **f** FULL Layers. **g** Batch size. **h** Activation function. **i** Dropout. **j** Learning rate. **k** Momentum. **l** Weight Penalty

Activation function (AF): The type of non-linear activation function is typically chosen to be the logistic sigmoid function `sigm` and `reLU`. We observe in Fig. 3h that `reLU` has better accuracy than `sigm`. Usually, `reLU` trains faster and outperforms the other activation functions. This parameter can be also set independently (Fig. 4h).

Dropout (D): The dropout process is that each hidden unit is randomly omitted from the deep architecture with

a probability such that a hidden unit cannot rely on other hidden units being presented, based on the observation that this parameter is correlating with the deep architecture (Fig. 4i). Previously, we observed the dependency between the influence of this parameter and the early/late fusion choice. Therefore, each fusion strategy leads to its own setting: `dropout = 0.5` for the early fusion (Fig. 3i) and `dropout = 0` for the late fusion.

Learning rate (LR) and momentum (M): We continue the analysis with the learning rate and momentum. Each iteration sees an update of the weight by the computed gradient. The learning rate represents the convergence speed and the momentum parameter introduces a *damping* effect on the search procedure, thus avoiding oscillations in irregular areas of the error surface by averaging gradient components with opposite signs and accelerating the convergence in long flat areas. In our experiments, we observed that in Fig. 3j, k the unit step and the momentum both near to 1 converges better. As a result, we take `learning rate = 1` and `momentum = 0.9`, which have to be set dependently (Fig. 4j, k). That is, it has been shown that the use of the momentum in the age estimation task can avoid the search procedure from being stopped in a local minimum and improves the convergence of the back propagation algorithm in general.

Weight penalty (WP): The last parameter is a constraint on the updating weight and we observe in Figs. 3l and 4l that the penalty can be set as `penalty = 1e-2` and will influence the setting of several parameters, such as the momentum, the baseline and the fully-connected layers.

In summary, the architecture of our CNN consists of three convolutional layers (CL), each of which is followed by the rectification, max-pooling and normalization. In addition, one fully connected layer (FL) is used. The network architecture is detailed as follows:

1. CL: The kernel size is 5×5, 1 stride - ReLU - Pool 3×3, 2 stride - Local Response Normalization (LRN).
2. CL: The kernel size is 5×5, 1 stride - ReLU - Pool 3×3, 2 stride - Local Response Normalization (LRN).
3. CL: The kernel size is 5×5, 1 stride - ReLU - Pool 3×3, 2 stride - Local Response Normalization (LRN).
4. FL.
5. Softmax Loss Layer.

4.2.2 Computational cost

Given an input image, our comparative approach compares it with all the k baseline samples, but not with all the N training samples. For example, in our experiments, each age label is represented by one baseline sample, and totally we have 9 labels, making $k = 9$. In other words, we only need to compute the comparative relation of the input image for k times, where k can be a small number and much less than N. Therefore, the computational cost of our approach is reasonable.

4.3 Discussions and comparisons with state-of-the-art methods

We compare our approach with others recent facial age estimation techniques such as rKCCA [21], IIS-LLD [10], CPNN [10], OHRank [31], AGES [37] and two *aging function regression* based methods, i.e. WAS [38] and AAS [39]. In addition, several conventional general-purpose classification methods, *k*-Nearest Neighbors (*k*NN) [40], Back Propagation neural network (BP) [41], C4.5 decision tree [42], Support Vector Machine (SVM) [43], Adaptive Network based Fuzzy Inference System (ANFIS) [44], as well as ranking based approaches are included, such as Ranking SVM [25], RankBoost [26], and RankNet [27]. We trained by using Leave-One-Person-Out (LOPO) test strategy [45], a popular test strategy, as suggested in the related benchmarks [10, 21, 31, 37]. Specifically, we split the used datasets (FG-NET and MORPH) by adopting the same training/testing protocol for all the comparing methods. For example, the LOPO is used on the FG-NET dataset as follows: in each fold, the images of one person are used as the testing set and those of the others are used as the training set. After 82 folds (the FG-NET dataset has a total of 82 subjects), each subject has been used as the testing set in turn, and the average results are computed from all of the estimates. However, since there are more than 13,000 subjects in the MORPH dataset, the LOPO test will be too time-consuming. Thus, we adopted the 10-fold cross validation instead on the MORPH dataset.

Our CRCNN method is configured with the deep learning parameters optimized in Section 4.2.1 and detailed in Table 1. Here the human tests are included for reference, which were performed on 5 percent samples from the FG-NET database and 60 samples from the MORPH database [10]. The performance of the age estimation is evaluated by the Mean Absolute Error (MAE) metric. In statistics, MAE is a metric used to measure how close a prediction is to the ground truth. In our case, the MAE is a mean of the absolute errors between the estimates and the true ages, $\text{MAE} = \sum_{k=1}^{N} |\widehat{a}_k - a_k| / N$, where $\widehat{a}_k$ and a_k are the estimate and the true age of the sample image k, and N is the total number of samples. The standard deviations on the MORPH dataset are also given in Table 2. For example, a number in the format $a \pm b$ means that the mean absolute error is a with a standard deviation b. Note that some of the comparing methods (e.g., rKCCA and rKCCA+SVM) do not show the standard deviations because they do not report the standard deviation values in the experiments

Table 2 Comparison with state-of-the-art methods on FG-NET and MORPH databases

Method	Database (FG-NET)	Database (MORPH)
CRCNN (early fusion) (RCNN)	**4.13**	**3.74 ± 0.29**
CRCNN (early fusion) (CNN)	4.72	4.33 ± 0.27
CRCNN (late fusion) (RCNN)	4.20	3.81 ± 0.32
CRCNN (late fusion) (CNN)	4.81	4.52 ± 0.23
Ranking SVM [25]	5.24	6.49 ± 0.17
RankBoost [26]	5.67	6.83 ± 0.25
RankNet [27]	5.46	6.71 ± 0.24
rKCCA [21]	-	3.98
rKCCA + SVM [21]	-	3.92
IIS-LLD [10] (Gaussian)	5.77	5.67 ± 0.15
IIS-LLD [10] (Triangle)	5.90	6.09 ± 0.14
IIS-LLD [10] (Single)	6.27	6.35 ± 0.17
CPNN [10] (Gaussian)	4.76	4.87 ± 0.31
CPNN [10] (Triangle)	5.07	4.91 ± 0.29
CPNN [10] (Single)	5.31	6.59 ± 0.31
OHRank [31]	6.27	6.28 ± 0.18
AGES [37]	6.77	6.61 ± 0.11
WAS [38]	8.06	9.21 ± 0.16
AAS [39]	14.83	10.10 ± 0.26
kNN [40]	8.24	9.64 ± 0.24
BP [41]	11.85	12.59 ± 1.38
C4.5 [42]	9.34	7.48 ± 0.12
SVM [43]	7.25	7.34 ± 0.17
ANFIS [44]	8.86	9.24 ± 0.17
Human Tests (HumanA)	8.13	8.24
Human Tests (HumanB)	6.23	7.23

The data in boldface means the best results of FG-NET and MORPH database are both from our CRCNN approach (with the early fusion scheme)

of their papers. For the results of the FG-NET dataset, we follow the common practice of the previous work (e.g., [10]) and do not show the standard deviations. For example, as mentioned in [10], "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". In other words, for the FG-NET database, the values of standard deviation are not so statistically meaningful and thus these values are not shown. The statistics are tabulated in Table 2. As can be seen, the best results (boldfaced) are both from our CRCNN approach (with the early fusion scheme). The second best results are also from our CRCNN approach (with the late fusion scheme). The overall performance of CRCNN is very encouraging. Our results are significantly better than all of the state-of-the-art methods. In comparison to the deep learning based method, i.e. CPNN [10], we also achieved a better performance, with a relative improvement of 13.24% (from 4.76 to 4.13 on FG-NET) and 23.20% (from 4.87 to 3.74 on MORPH). These facts validate the robustness of the newly proposed comparative approach.

We further performed an evaluation on the IoG database. It consists of 28,231 facial images collected from the Flicker. Each face is labeled in one of the defined seven age groups: 0–2, 3–7, 8–12, 13–19, 20–36, 37–65, and 66+. In our evaluation, we considered only faces having an interocular distance more than 40 pixels, resulting in a subset of 1495 face images. We further reorganized the age labels into the child, teen, and adult classes with the age range of 0–12, 13–19, and 20+, respectively. The setting yielded the following amount of samples per each age group: 546, 250, and 699. Finally, we performed the same normalizations as in the previous experiments on all of the IoG faces. We compare our results with the ranking based methods, including [25–27], and Local Binary Pattern Kernel Density Estimation (LBP-KDE) [30]. The age group classification performance is represented in Table 3. We can observe better performances of our approach over the state-of-the-art methods, with a relative improvement from 4.74% (in LBP-KDE) to 13.74% (in RankBoost). We believed the outperformance of our CRCNN approach on all the datasets demonstrated its effectiveness for practical applications.

5 Conclusions

This paper proposed a novel comparative deep learning framework for facial age estimation, namely Comparative Region Convolutional Neural Network (CRCNN). Motivated by human cognitive processes, we use a comparative approach to determine the age of an unseen person. To the best of our knowledge, it is the first comparative approach in deep learning for facial age estimation and the experimental results validate the outperformance of our CRCNN approach over state-of-the-art methods. One of our future work is to further improve the baseline selection, since obtaining an effective baseline is crucial in our

Table 3 Comparison with state-of-the-art methods on IoG database

Method	Database (IoG)
CRCNN (early fusion) (RCNN)	**66.41%**
CRCNN (early fusion) (CNN)	63.16%
CRCNN (late fusion) (RCNN)	65.48%
CRCNN (late fusion) (CNN)	62.19%
LBP-KDE [30]	61.67%
Ranking SVM [25]	56.17%
RankBoost [26]	52.67%
RankNet [27]	55.08%

The data in boldface shows that the performance of our approach on IoG database is better comparing to the state-of-the-art methods

comparative approach. As aging procedures are quite different from person to person, especially from different social groups, we also plan to build a "baseline bank" (constituted by a set of baselines, with each corresponds to a computed group of social consistency), instead of using a single and global baseline. Further research on CRCNN in these directions will be attractive future work.

Endnote

[1] Note that, in this paper, the comparative relations of "younger" and "older" are actually defined to be "younger than or equal to" and "older than or equal to", respectively. The "same age" relation thus exists when the two relations hold simultaneously.

Acknowledgements
The authors extend their appreciation to the Deanship of Scientific Research at King Saud University for funding this work through the research group project No. RGP-049.

Authors' contributions
FA and TL collected the datasets and carried out the experiments. WC and NY constructed the main ideas of the research. AH and MA took part in the examination of the study. All authors read and approved the final manuscript.

Competing interests
The authors declare that they have no competing interests.

Author details
[1]Social Networks and Human-Centered Computing Program, Taiwan International Graduate Program, Institute of Information Science (IIS), Academia Sinica, Academia Road, Taipei, 11529, Taiwan. [2]Research Center for Information Technology Innovation (CITI), Academia Sinica, Academia Road, Taipei, 11529, Taiwan. [3]Department of Computer Science, National Chengchi University, ZhiNan Road, Taipei, 11605, Taiwan. [4]Department of Software Engineering, College of Computer and Information Sciences, King Saud University, Riyadh, 11362, Saudi Arabia.

Received: 31 July 2016 Accepted: 2 December 2016
Published online: 19 December 2016

References

1. Microsoft Corp, How-Old.net. (2015). https://how-old.net. Accessed 10 July 2016
2. T-H Tsai, W-C Jhou, W-H Cheng, M-C Hu, I-C Shen, T Lim, K-L Hua, A Ghoneim, MA Hossain, SC Hidayati, Photo sundial: estimating the time of capture in consumer photos. Neurocomputing. **177**, 529–542 (2016)
3. C-W You, Y-L Chen, W-H Cheng, Socialcrc: enabling socially-consensual rendezvous coordination by mobile phones. Pervasive Mobile Comput. **25**, 67–87 (2016)
4. W-H Cheng, C-W Wang, J-L Wu, Video adaptation for small display based on content recomposition. IEEE Trans. Circ. Sys. Video Technol. **17**(1), 43–58 (2007)
5. B Wu, W-H Cheng, Y Zhang, T Mei, in *Proceedings of the ACM International Conference on Multimedia*. Time matters: Multi-scale temporalization of social media popularity, (2016), pp. 1336–1344
6. B Wu, T Mei, W-H Cheng, Y Zhang, in *Proceedings of the AAAI Conference on Artificial Intelligence*. Unfolding temporal dynamics: Predicting social media popularity using multi-scale temporal decomposition, (2016), pp. 272-278
7. Y Fu, G Guo, TS Huang, Age synthesis and estimation via faces: a survey. IEEE Trans. Pattern Anal. Mach. Intell. **32**(11), 1955–1976 (2010)
8. SE Choi, YJ Lee, SJ Lee, KR Park, J Kim, Age estimation using a hierarchical classifier based on global and local facial features. J. Pattern Recognit. **44**(6), 1262–1281 (2011)
9. SE Choi, YJ Lee, SJ Lee, KR Park, J Kim, in *Control Automation Robotics and Vision (ICARCV), 2010 11th International Conference on*. A comparative study of local feature extraction for age estimation (IEEE, 2010), pp. 1280–1284
10. X Geng, C Yin, Z-H Zhou, Facial age estimation by learning from label distributions. IEEE Trans. Pattern Anal. Mach. Intell. **35**(10), 2401–2412 (2013)
11. T Lim, K-L Hua, H-C Wang, K-W Zhao, M-C Hu, W-H Cheng, in *Proceedings of the IEEE International Workshop on Multimedia Signal Processing*. Vrank: Voting system on ranking model for human age estimation, (2015), pp. 1–6
12. JB Carroll, *Human cognitive abilities: A survey of factor-analytic studies*. (Cambridge University Press, New York, 1993)
13. R Girshick, J Donahue, T Darrell, J Malik, Rich feature hierarchies for accurate object detection and semantic segmentation. IEEE Conf. Comput. Vis. Pattern Recognit, 580–587 (2014)
14. T Serre, L Wolf, S Bileschi, M Riesenhuber, T Poggio, Robust object recognition with cortex-like mechanisms. IEEE Trans. Pattern Anal. Mach. Intell. **29**(3), 411–426 (2007)
15. Y Bengio, in *International Conference on Statistical Language and Speech Processing*. Deep learning of representations: Looking forward (Springer Berlin Heidelberg, 2013), pp. 1–37
16. MA Carreira-Perpinan, W Wang, Distributed optimization of deeply nested systems. Int. Conf. Artif. Intell. Stat. **33**, 10–19 (2014)
17. A Krizhevsky, I Sutskever, G Hinton, Imagenet classification with deep convolutional neural networks. Conf. Neural Inf. Process. Syst, 1097–1105 (2012)
18. Y Jia, E Shelhamer, J Donahue, S Karayev, J Long, R Girshick, S Guadarrama, T Darrell, in *Proceedings of the 22nd ACM international conference on Multimedia*. Caffe: Convolutional architecture for fast feature embedding (ACM, 2014), pp. 675–678
19. JK Pontes, AS Britto, C Fookes, AL Koerich, A flexible hierarchical approach for facial age estimation based on multiple features. Pattern Recognit. **54**, 34–51 (2016)
20. I Huerta, C Fernandez, A Prati, Facial age estimation through the fusion of texture and local appearance descriptors. Eur. Conf. Comput. Vis. Workshop, 667–681 (2014)
21. G Guo, G Mu, A framework for joint estimation of age, gender and ethnicity on a large database. Image Vis. Comput. **32**(10), 761–770 (2014)
22. H Takimoto, Y Mitsukura, M Fukumi, N Akamatsu, Robust gender and age estimation under varying facial pose. Electronics Commun. Japan. **91**(7), 32–40 (2008)
23. C Yan, C Lang, T Wang, X Du, C Zhang, Age estimation based on convolutional neural network. Adv. Multimedia Inf. Process. **8879**, 211–220 (2014)
24. F Gurpinar, H Kaya, H Dibeklioglu, A Salah, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. Kernel elm and cnn based facial age estimation, (2016), pp. 80–86
25. T Joachims, Optimizing search engines using clickthrough data. International Conference on Knowledge Discovery and Data Mining, 133–142 (2002)
26. Y Freund, R Iyer, RE Schapire, Y Singer, An efficient boosting algorithm for combining preferences. Journal of Machine Learning Research, 4(Nov), 933-969 (2003)
27. C Burges, T Shaked, E Renshaw, A Lazier, M Deeds, N Hamilton, G Hullender, Learning to rank using gradient descent. International Conference on Machine Learning, 89-96 (2005)
28. S Chopra, R Hadsell, Y LeCun, Learning a similarity metric discriminatively, with application to face verification. IEEE Conf. Comput. Vis. Pattern Recognit. **1**, 539–546 (2005)
29. J Nocedal, SJ Wright, Numerical optimization. Springer Series in Operations Research and Financial Engineering (2006)
30. J Ylioinas, A Hadid, X Hong, M Pietikäinen, Age estimation using local binary pattern kernel density estimate. Int. Conf. Image Anal. Process. **8156**, 141–150 (2013)
31. K-Y Chang, C-S Chen, Y-P Hung, in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. Ordinal hyperplanes ranker with cost sensitivities for age estimation, (2011), pp. 585–592
32. A Lanitis, CJ Taylor, T Cootes, Toward automatic simulation of aging effects on face images. IEEE Trans. Pattern Anal. Mach. Intell. **24**(4), 442–455 (2002)

33. K Ricanek, T Tesafaye, Morph: a longitudinal image database of normal adult age-progression. International Conference on Automatic Face and Gesture Recognition, 341–345 (2006)
34. AC Gallagher, T Chen, Using group prior to identify people in consumer images. IEEE Conference on Computer Vision and Pattern Recognition, 1-8 (2007)
35. J Sanchez-Riera, K-L Hua, Y-S Hsiao, T Lim, SC Hidayati, W-H Cheng, A comparative study of data fusion for rgb-d based visual recognition. Pattern Recognit. Lett. **73**, 1–6 (2016)
36. M Li, T Zhang, Y Chen, A Smola, in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. Efficient mini-batch training for stochastic optimization, (2014), pp. 661–670
37. X Geng, W-H Zhou, K Smith-Miles, Automatic age estimation based on facial aging patterns. IEEE Trans. Pattern Anal. Mach. Intell. **29**(12), 2234–2240 (2007)
38. A Lanitis, CJ Taylor, T Cootes, Toward automatic simulation of aging effects on face images. IEEE Trans. Pattern Anal. Mach. Intell. **24**(4), 442–455 (2002)
39. A Lanitis, C Draganova, C Christodoulou, Comparing different classifiers for automatic age estimation. IEEE Trans. Syst. Man Cybernet. **34**(1), 621–628 (2004)
40. EA Patrick, FP Fischer, A generalized k-nearest neighbor rule. Inf. Control. **16**(2), 128–152 (1970)
41. DE Rumelhart, GE Hinton, RJ Williams, Learning representations by backpropagating errors. Nature. **323**(9), 533–536 (1986)
42. JR Quinlan, *C4.5: Programs for machine learning*. (Morgan Kaufmann, San Francisco, 1993)
43. V Vapnik, *Statistical learning theory*. (Wiley, New York, 1998)
44. R Jang, Anfis: Adaptive network based fuzzy inference system. IEEE Trans. Syst. Man Cybernet. **23**(3), 665–685 (1993)
45. X Geng, Z-H Zhou, K Smith-Miles, Automatic age estimation based on facial aging patterns. IEEE Trans. Pattern Anal. Mach. Intell. **29**(12), 2234–2240 (2007)