

## A Study on Apparent Age Estimation

Yu Zhu<sup>\* 1</sup>, Yan Li<sup>\* 1,2</sup>, Guowang Mu<sup>1</sup>, and Guodong Guo<sup>1</sup>

<sup>1</sup>Lane Department of CSEE, West Virginia University, Morgantown WV 26506, USA,

<sup>2</sup>Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing, 100190, China

yzhu4@mix.wvu.edu, yan.li@vip1.ict.ac.cn, guowang.mu@mail.wvu.edu ,  
Guodong.Guo@mail.wvu.edu (corresponding author)

### Abstract

*Age estimation from facial images is an important problem in computer vision and pattern recognition. Typically the goal is to predict the chronological age of a person given his or her face picture. It is seldom to study a related problem, that is, how old does a person look like from the face photo? It is called apparent age estimation. A key difference between apparent age estimation and the traditional age estimation is that the age labels are annotated by human assessors rather than the real chronological age. The challenge for apparent age estimation is that there are not many face images available with annotated age labels. Further, the annotated age labels for each face photo may not be consistent among different assessors. We study the problem of apparent age estimation by addressing the issues from different aspects, such as how to utilize a large number of face images without apparent age labels to learn a face representation using the deep neural networks, how to tune the deep networks using a limited number of examples with apparent age labels, and how well the machine learning methods can perform to estimate apparent ages. The apparent age data is from the ChaLearn Looking At People (LAP) challenge 2015. Using the protocol and time frame given by the challenge competition, we have achieved an error of 0.294835 on the final evaluation, and our result has been ranked the 3rd place in this competition.*

### 1. Introduction

Age estimation is an important problem in computer vision and pattern recognition. Estimating the age from facial images has received great interests in recent years [9, 8, 15]. Typically the goal of age estimation is to predict the chrono-

logical age of a person given his or her face picture. An age estimation system usually involves two components, i.e., aging pattern representation and aging function learning. Representative approaches to age estimation can be found in some survey papers, such as [9, 8, 15].

A related but different problem is apparent age estimation, where the focus is on predicting “how old does the person look like?” rather than “what is the real age of this person?” It is relatively new to study the problem of apparent age estimation. A key difference between apparent age estimation and the traditional age estimation is that the age labels are annotated by human assessors rather than the real chronological age. In reality, some people may look younger than the real chronological age, while some may look older. As a result, the apparent age may be quite different from the real age for each subject.

The challenge for apparent age estimation is that there are not many face images available with annotated age labels. Further, the annotated age labels for each face photo may not be consistent among different assessors.

The problem of apparent age estimation is studied in this work. Particularly, we firstly utilize a large number of face images without apparent age labels to learn a face representation using the deep neural networks, then we study how to fine-tune the deep networks using a limited number of data with apparent age labels. We also explore how well the machine learning methods can perform to estimate apparent ages.

In our study, the apparent age data is from the ChaLearn Looking At People (LAP) challenge 2015 [17]. Using the protocol and time frame given by the challenge competition, we can achieve an error of 0.294835 on the final evaluation, and our result is ranked the 3rd place in this competition. Figure 1 illustrates our proposed method for this challenging problem.

The rest of paper is organized as follows. In Section 2,

<sup>\*</sup>The first two authors contributed equally to this work.

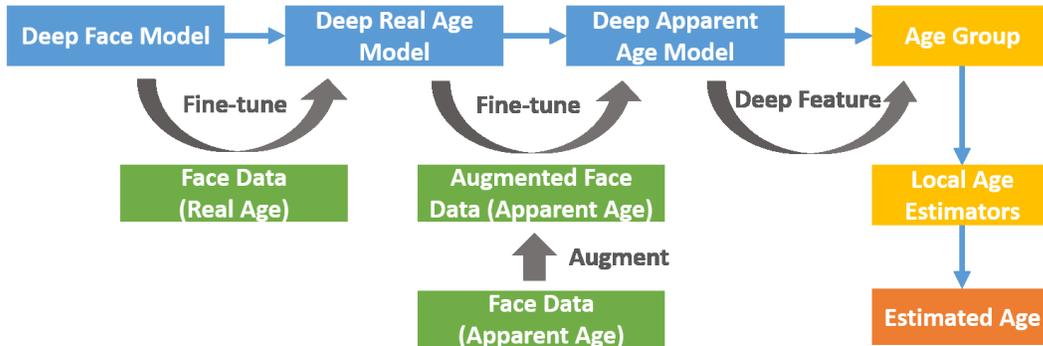


Figure 1. Illustration of our method for training deep model for apparent age estimation.

we describe the details of our proposed architecture for apparent age estimation. In Section 3, the databases that are used for training the deep networks are presented. Experiments are conducted and experimental results are shown in Section 4. Finally, we draw some conclusions in Section 5.

## 2. Approach

In this section, we present our method. First, a data pre-processing procedure is applied. Second, we describe the design of our architecture of deep neural network in a cascade way. Then a coarse-to-fine scheme is proposed, which consists of age grouping and local age estimators within age groups. Finally we perform fusing of multiple predictors to obtain a better estimation of the apparent ages.

### 2.1. Preprocessing

#### 2.1.1 Face Detection and Landmark Detection

Given the image data, we first applied face detection and landmark localization using Microsoft Project Oxford API [2] and Face++ API [1]. Images are rotated every 10 degrees for further detection if no face is detected in the original image. The rest of images that are still not detected by rotation are not used in our approach. During the final submission, all the undetected images are set to the average age from the overall predictions as their age label. All the detected face images are cropped and aligned by the eye locations with an image size of  $256 \times 256$ , which is kept for all face images.

#### 2.1.2 Data Augmentation

The face images from the age estimation challenge are collected in the wild and the number of training images is very limited (about 2500 images). There are various poses, illumination, and image quality issues in the dataset. To handle this issue, we apply data augmentation to create new training samples from the given training data before fine-tuning

the deep network.

Each training example (after cropping and alignment) was perturbed before presenting it to the network by randomly rotating, translating, scaling, adding noise and optionally flipping. By doing this data augmentation, for each training sample, multiple new training samples can be generated so that the total number of apparent age data is increased at the same time the deep networks are expected to “see” more data with more variations in learning.

### 2.2. Deep Neural Network

Recently, deep learning methods especially the convolutional neural networks (CNN) have shown promising performance in age estimation. An early work using CNN for age estimation is [22]. A more recent one is [23], where a multi-scale CNN was proposed for estimating age, as well as gender and race from a facial image. However, the performance of deep learning on apparent age estimation has not been studied yet. In our proposed method, we utilize the nice properties of deep representation and exploit the performance of deep learning method for apparent age estimation. Based on the deep features, we further propose to utilize age grouping and local age estimators to improve the overall performance.

We hope that the deep neural networks are capable of capturing the useful information from face images to help age estimation. Considering the number of training data provided by the challenge is relatively small (about 2500 images for training and 1200 images for validation), we design our apparent age estimation architecture accordingly and propose a three-step training procedure. The first step is to pre-train the network using a large number of face images, e.g., from the WebFace dataset [24]. In order to learn more robust and representative features by deep neural networks, in our method we expect to utilize a much deeper neural network rather than previous work [23] for real age estimation. Particularly, we chose the GoogLeNet, [19] which is 22-layer deep neural network, to train our deep

models. We hope that through this pre-training step, the deep network is able to capture general facial representations.

The second step for training our deep model is fine-tuning the network parameters using a large number of data with biological age labels. We merged multiple age databases into one training dataset, details of the databases we used is presented in Section 3. In this fine-tuning process, we consider the age estimation as a regression problem, therefore we change the loss function from softmax to Euclidean loss for this real-valued regression task.

Euclidean loss function  $E$  computes the sum of squares of differences of its two inputs, which can be written as:

$$E = \frac{1}{2N} \sum_{i=1}^N \|\hat{y}_i - y_i\|^2, \quad (1)$$

where  $N$  is the number of samples,  $\hat{y}_i$  is the output from the network and  $y_i$  is the ground truth age labels.

Based on the pre-trained deep face model, now we are aiming at fine-tuning the deep model for age estimation. Since this is a different task and the loss function is also changed to Euclidean loss, we initialize a relatively large value for the learning rate. After the above steps, the learned deep network is expected to have the capability to predict real age given input facial image. However, such deep model may not perform well for apparent age estimation since the real age usually differs from the apparent age. Therefore, the third step to train our deep age model is a further fine-tuning step, which utilizes the apparent age data. In this fine-tuning, the learning rate is set to a relatively small number for better convergence. We boost the learning rate for the feature layers, so that most of the age models change slowly but let the feature layers learn fast. After this fine-tuning the deep model is considered to have more capability to estimate the apparent age. Details about the fine-tuning of network parameters can be found in Section 4.2.

### 2.3. Age Grouping and Age Estimation

Given the deep features, we propose a coarse-to-fine strategy for apparent age estimation. First step is to categorize an input sample into one of the 10 age groups, the second step is to predict the apparent age value within some age groups. In this subsection, we describe the details of this procedure.

Inspired by the work in [10], for age grouping, we use all the training samples to learn the Support Vector Machine (SVM) [18] classifier, where the age labels are assigned to corresponding age group labels. Table 1 shows the lower bound and upper bound of the 10 age groups in our method. Our idea is to classify input samples into a specific age group coarsely at the first place, and then some local age estimators are trained within each age group for better

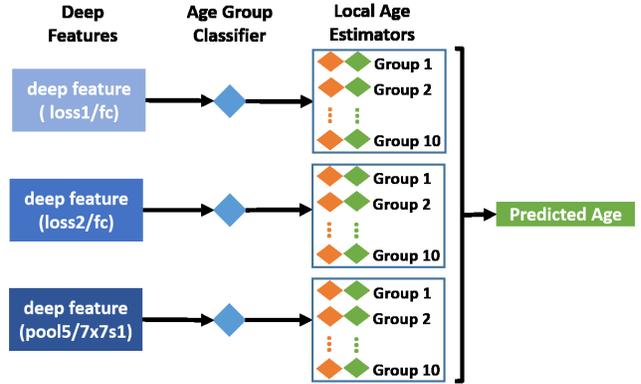


Figure 2. Illustration of our method for age grouping using deep features.  $loss1/fc$ , and  $loss2/fc$  are feature layers on top of two inception layers in the middle of the GoogLeNet, and  $pool5/7 \times 7_s1$  is the last feature layer.

Table 1. Ranges of 10 Age Groups that used in our method. The first row shows the group number, the second row shows the lower bound of each age group and the third row shows the upper bound of each group.

Group	1	2	3	4	5	6	7	8	9	10
Lower	0	7	14	21	28	35	42	49	56	63
Upper	6	13	20	27	34	41	48	55	62	93

age estimation. Practically for age grouping, the classification accuracy is relatively low ( 50%) when the test sample is required to be classified into the correct age group. However, the classification accuracy increases significantly ( 99%) when test samples are considered correct if they drop into some adjacent age groups. We found that it is appropriate to use 5 adjacent age groups around the predicted age group, therefore in each local estimator, data from 5 adjacent age groups are used for training the local age estimators.

For each age group, the Support Vector Regression (SVR) [18] and Random Forests (RF) [3] are trained for age estimation. In each group  $k$ , data from age group  $\max(1, k - 2)$  to  $\min(k + 2, 10)$  are used for training. In this way, each data sample is firstly classified into a specific age group, and then age estimation is done using the SVR and RF corresponding to the selected age groups. Figure 2 illustrates our method for age grouping using deep features.

### 2.4. Fusing Multiple Age Estimators

In order to further improve the age estimation performance, we propose to combine the estimation results using multiple features with different methods. The idea is to fully utilize the deep representations not only from the last fully-connected layer but from different layers of the deep network, and also combine different age estimators. Particularly, we use the fusion result from both RF and SVR after

age grouping as the estimation result for each deep feature. The estimation of three layers (last feature layer and layers in the middle of the network) in each deep model are then fused as one result. We also trained multiple deep models and their fused result is considered as the final estimation. The fusion is applied on score level, which takes the average of multiple results.

### 3. Databases

In this section, we present the databases that are used to train the deep models in a cascaded way. We firstly pre-train a GoogLeNet model using CASIA-WebFace database. Then multiple databases with real-age labels are merged into one dataset for fine-tuning the deep model, for the real age estimation. Finally, the apparent age data from the challenge are used to fine-tune the deep model parameters for apparent age estimation.

#### 3.1. CASIA-WebFace Database

CASIA-WebFace [24] is a large scale database including about 10,595 subjects and 494,414 images. To the best of our knowledge, the size of this database ranks second in the literature, only smaller than the private database of Facebook (SCF) [20]. It is originally collected to train deep convolutional neural network to learn discriminative representation and obtain state-of-the-art accuracy on LFW [12] and YTF [21]. In this work, we use them as the training data of the first step, network pre-training, i.e., identity discriminant learning.

#### 3.2. Adience Age Database

Adience Age Database [6] is another large scale database, which was collected to capture all the variations in appearance, noise, pose, lighting and more, and can be expected to serve as images taken without careful preparation or posing. There are 18,300 images in this dataset. The sources of the images included in Adience Age Database are Flickr albums, assembled by automatic upload from iPhone5 (or later) smart-phone devices, and released by their authors to the general public under the Creative Commons (CC) license. However, different from the CACD, this database doesn't have the accurate age annotation, i.e., age is annotated in a range form like 0-2, 4-6, 8-13, 15-20, 25-32, 38-43, 48-53, 60+. As we use Euclidean loss for the deep networks, in this work we transfer the range annotation into traditional single label by just taking the mean value, e.g., 28-32 is converted to 30.

#### 3.3. Morph Age Database

The Morph database [16] was also used for our network training. It is a large database containing two sections, I and II. Since Morph-I is too small, we used Morph-II that

contains about 55,000 face images. The Morph is a multi-ethnic database. It has about 77% Black faces and 19% White, while the remaining 4% includes Hispanic, Asian, Indian, and Others. Although Morph has a great number of faces, it is like mug-shot images, which is quite different from the faces in the wild.

#### 3.4. FGNET Age Database

FGNET [7] consists of 1,002 images of 82 subjects, labeled with accurate chronological age. Some of these photos were acquired under controlled conditions. One characteristic of this database is that it contains a lot of samples of young ages, i.e., from 1-13 years.

#### 3.5. Lifespan Age Database

Lifespan [14] is a database of 575 individual faces ranging from ages 18 to 93, and it was developed to be more representative of age groups across the lifespan, with a special emphasis on recruiting older adults. The database has faces of 218 adults age 18-29, 76 adults age 30-49, 123 adults age 50-69, and 158 adults age 70 and older. In addition, this database also contains the information of facial expressions, like neutral, happy, annoyed, grumpy, and surprised expressions. The database was originally developed in the psychology society, and was introduced to the computer vision community for computational age estimation in [11].

#### 3.6. CACD Age Database

CACD is short for Cross-Age Celebrity Dataset [5], a large scale age database. It contains more than 160,000 images of 2,000 celebrities with age ranging from 16 to 62 years. It is originally designed for investigating the problem of age-invariant face recognition and retrieval. Nevertheless, annotation in this database cannot be guaranteed to be correct [5].

#### 3.7. Other Age Database

Besides the above public databases, in this work we also used a private age database. This database is composed of controlled face images which were captured in studio environment and in the wild. Totally, we have about 8,941 faces with age labels in this dataset.

## 4. Experiments

### 4.1. Evaluation Protocol

The age estimation challenge is a new track at the ChaLearn LAP challenge 2015. A dataset of 4699 images is provided, each image is labeled a real number from 0 to 100 indicating the apparent age. The images are collected from two web-bases application and labeled by at least 10 different users. The ground truth is calculated with all the

users’ votes, where the mean age from all the votes is considered as the apparent age label. Images in this dataset are taken in the wild, so there exist various poses, illumination and quality changes. The age estimation challenge aims at investigating the performance of estimation methods on apparent age rather than real chronological age.

There are 4699 images in this dataset which are divided into training, validation and test sets. In the training set there are 2476 images with apparent age labels (mean age and standard deviation), while in the validation set there are 1136 images (labels are provided in the final evaluation phase). The rest 1087 images are used as the test set (labels are not public). Some example images of the dataset are shown in Figure 3. During the development phase of the challenge we evaluate our methods on the validation set. In the final evaluation phase we merge the training and validation set into one training set and all the models are retrained for the final evaluation on the test set.

The estimation result is evaluated by fitting a normal distribution with the mean and standard deviation of the votes for each image. The error over all the images is computed as:

$$\epsilon = \frac{1}{N} \sum_{i=1}^N 1 - e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \quad (2)$$

where  $\epsilon$  is the error,  $N$  is the number of test samples,  $x$  is the predicted age,  $\mu$  is the mean age (labeled apparent age) and  $\sigma$  is the standard deviation.

## 4.2. Implementation Details

For data augmentation, we used predefined ranges for the parameter random selection to generate new training samples from apparent age data. Specifically, we chose (1) rotation: random with angle between -10 and 10 degrees; (2) translation: random with shift between -5 and 5 pixels in both x and y directions; (3) scaling: random with the scale factor between 1.0 and 1.2; (4) adding noise: add random gaussian noise with sigma between 0 and 2.0; (5) flip: yes or no. For each training sample, 40 new training samples are generated by this procedure and the total training samples become 40 times more than the original data size.

The training of GoogLeNet is implemented using the Caffe framework [13]. As described in Section 2.2, we pre-train a GoogLeNet using WebFace dataset, which contains about 500,000 face images. We then fine-tune the network parameters using a large number of face images with real age labels for the age estimation task. Since we consider age estimation as a regression problem, the loss function of GoogLeNet is changed from Softmax to Euclidean loss. In this fine-tuning step, the dataset consists multiple real age databases which are described in Section 3. The total number of images is about 240,000. We used mini-batch SDG (Stochastic gradient descent) with momentum settings. The mini-batch size is set to 32 and momentum

is set to 0.9. We initialize the learning rate to 0.01, and 0.00001 for the first and second stage of fine-tuning, respectively. The learning rate of feature layers (*loss1/fc*, *loss2/fc*, and *pool5/7 × 7\_s1*) is initially set to 100 times of the hidden layers. The learning rate decreases in polynomial decay with power of 0.5. The weights of the loss in the three layers are set to 0.3, 0.3 and 1.0. The training procedure stops after 30 epochs.

For fine-tuning the GoogLeNet using the apparent age data, same settings are utilized as previous procedure except that the iteration number is reduced to 15,000. We also trained multiple GoogLeNet (5 for the challenge final evaluation) with the same initialization and learning rate policies, the only difference lies in sampling methodologies and the random order for the input images. For each deep model, three feature vectors (each of 1024 dimension) are extracted according to three feature layers in the GoogLeNet: *loss1/fc*, *loss2/fc* and *pool5/7 × 7\_s1*, respectively.

In the next step, we train the SVM classifiers for age grouping. In development stage, only training images are used to train age group classifier, and validation set is used for tuning the parameters. While on the final evaluation stage, we merge the training and validation set into one dataset for training classifiers. In our implementation, SVM [4] is trained with RBF kernel, and the parameter C is set to 100, gamma is set to  $1/\# \text{ of features}$ .

Within each age group, we use SVR and RF to train the local age estimators using the data from adjacent age groups (as described in Section 2.3). In SVR, the value of C is set to 100 and gamma is set to  $1/\# \text{ of features}$ . In RF, number of tree is set to 200, and number of features in each split is set to  $\# \text{ of features}/3$ .

During the test phase, face detection is applied first to each test image and the detected face images are cropped and aligned according to the procedure mentioned in Section 2.1. The undetected images are left and the average age is set as the finally predicted age. The detected facial images in the test set are used in our method for age estimation. The finally predicted age is computed by averaging all the methods for each layer feature in all the deep models.

## 4.3. Experimental Results

We measure the performance using different features from different layers of GoogLeNet. Since the apparent age labels of the test data are not known, all the experiments are tested on the validation set. The experimental results are shown in Table 2. From the table, one can see that, features from different layers perform differently: the *pool5/7x7\_s1* layer performs better than the other layers. Considering different methods, the result shows that RF performs slightly better than SVR. Meanwhile, by using age grouping the estimation error can be further decreased.



Figure 3. Example images in the dataset of ChaLearn LAP challenge apparent age estimation track.

Table 2. Experimental Results of Age Estimation on Validation set

Feature	Method	Error
loss1/fc	RF	0.3383
loss1/fc	SVR	0.3493
loss1/fc	Group + RF	0.3325
loss1/fc	Group + SVR	0.3467
loss2/fc	RF	0.3285
loss2/fc	SVR	0.3253
loss2/fc	Group + RF	0.3265
loss2/fc	Group + SVR	0.3242
pool5/7x7_s1	RF	0.3215
pool5/7x7_s1	SVR	0.3229
pool5/7x7_s1	Group + RF	0.3217
pool5/7x7_s1	Group + SVR	0.3236

Table 3. Estimation Error using different fusion methods on validation set.

Method	Error
Fusion RF	0.3172
Fusion SVR	0.3122
Fusion Group + RF	0.3150
Fusion Group + SVR	0.3117
Fusion RF and SVR	0.3096
Fusion Group + RF, Group + SVR	0.3086

In Table 3, we show the experimental results of fusing different methods at the score level. The estimation result is calculated by averaging each of the age estimator. We first show the results by fusing results from different deep features. When RF is used as the age estimator, the fusion of three features (from one deep model) gets the result 0.3172, which is better than using single feature (Rows 1,5 and 9 in Table 2). When SVR is used as the age estimator, by fusion of the three features the error drops to 0.3122. It is better compared with the single layer errors 0.3493, 0.3253, and 0.3229. The best result is achieved by fusing RF with age group and SVR with age group, the error obtained is 0.3086.

Based on the above analysis and practical results, we found that: (1) Age grouping that applied before RF and

Table 4. Age Estimation Challenge Final Rank on the test set.

Rank	Team	Error (Test phase)
1	CVL_ETHZ	0.264975
2	ICT-VIPL	0.270685
3	AgeSeer (no code)	0.287266
3	WVU_CVL (Ours)	0.294835
4	SEU-NJU	0.305763
5	UMD	0.373352
6	Enjuto	0.374390
7	Sungbin Choi	0.420554
8	Lab219A	0.499181
9	Bogazici	0.524055
10	Notts CVLab	0.594248

SVR can help improve the performance than directly using RF or SVR. (2) Fusion of features from three layers performs better than each individual feature from a single layer. (3) Fusing RF and SVR under age grouping can further improve the overall performance. Therefore, according to these observations, our solution for the challenge contains 5 different GoogLeNet models. For each deep model, the features from three layers are extracted, respectively. For each layer of feature, age grouping is applied, RF and SVR are then used for age estimator. The fusion of RF and SVR with age grouping for all the three layers are considered as the estimation results for one deep network. The final result is to fuse the results from 5 deep networks. The final results of the age estimation challenge given by the competition organizer, are shown in Table 4.

## 5. Conclusions

We have studied the problem of apparent age estimation, and presented an effective approach to address the problem. Our method has utilized the deep representations that are trained in a cascaded way on deep neural networks. Firstly face images without age labels are used to pre-train the GoogLeNet model, then data with chronological age labels are used to fine-tune the network parameters. Finally, the

apparent-age data are used to do the fine-tuning for apparent age estimation. After extracting the deep representations, we have performed apparent age estimation in a coarse to fine manner. Age grouping is used to firstly classify the sample into age groups and then local age estimators are applied for specific age estimation. Furthermore, we have applied score level fusion to further improve the performance. Finally, in the ChaLearn LAP challenge 2015 age estimation track, our method has shown promising results and achieved the 3rd place. In the future, we will explore more deep representations to improve the performance further.

## References

- [1] Face++ api. <http://www.faceplusplus.com>.
- [2] Microsoft project oxford api. <https://www.projectoxford.ai>.
- [3] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.
- [4] C.-C. Chang and C.-J. Lin. Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.
- [5] B.-C. Chen, C.-S. Chen, and W. H. Hsu. Cross-age reference coding for age-invariant face recognition and retrieval. In *European Conference on Computer Vision*, pages 768–783. Springer, 2014.
- [6] E. Eidinger, R. Enbar, and T. Hassner. Age and gender estimation of unfiltered faces. *Information Forensics and Security, IEEE Transactions on*, 9(12):2170–2179, 2014.
- [7] FGNET. The fg-net aging database. In <http://www.fgnet.rsunit.com/>, 2002.
- [8] Y. Fu, G.-D. Guo, and T. S. Huang. Age synthesis and estimation via faces: A survey. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 32(11):1955–1976, 2010.
- [9] G.-D. Guo. Human age estimation and sex classification. In C. Shan, F. Porikli, T. Xiang, and S. Gong, editors, *Video Analytics for Business Intelligence*, pages 101–131. Springer, 2012.
- [10] G.-D. Guo, G. Mu, Y. Fu, C. Dyer, and T. S. Huang. A study on automatic age estimation on a large database. In *IEEE International Conference on Computer Vision*, pages 1986–1991, 2009.
- [11] G.-D. Guo and X. Wang. A study on human age estimation under facial expression changes. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 2012.
- [12] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical report, Technical Report 07-49, University of Massachusetts, Amherst, 2007.
- [13] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the ACM International Conference on Multimedia*, pages 675–678. ACM, 2014.
- [14] M. Minear and D. C. Park. A lifespan database of adult facial stimuli. *Behavior Research Methods, Instruments, & Computers*, 36(4):630–633, 2004.
- [15] N. Ramanathan, R. Chellappa, and S. Biswas. Age progression in human faces : A survey. *Visual Languages and Computing*, 2009.
- [16] K. Ricanek Jr and T. Tesafaye. Morph: A longitudinal image database of normal adult age-progression. In *7th International Conference on Automatic Face and Gesture Recognition*, pages 341–345. IEEE, 2006.
- [17] E. Sergio, F. Junior, P. Pablo, B. Xavier, G. Jordi, J. E. Hugo, Isabelle, and Guyon. Chalearn 2015 apparent age and cultural event recognition: datasets and results. In *ICCV ChaLearn Looking at People workshop*, 2015.
- [18] A. J. Smola and B. Schölkopf. A tutorial on support vector regression. *Statistics and computing*, 14(3):199–222, 2004.
- [19] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015.
- [20] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1701–1708. IEEE, 2014.
- [21] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 529–534. IEEE, 2011.
- [22] M. Yang, S. Zhu, F. Lv, and K. Yu. Correspondence driven adaptation for human profile recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 505–512, Colorado Springs, CO, 2011.
- [23] D. Yi, Z. Lei, and S. Z. Li. Age estimation by multi-scale convolutional network. In *Computer Vision-ACCV 2014*, pages 144–158. Springer, 2015.
- [24] D. Yi, Z. Lei, S. Liao, and S. Z. Li. Learning face representation from scratch. *arXiv preprint arXiv:1411.7923*, 2014.