Recurrent Face Aging

Wei Wang[1], Zhen Cui[3], Yan Yan[1], Jiashi Feng[2*], Shuicheng Yan[4,2], Xiangbo Shu[2], and Nicu Sebe[1]

[1]Department of Information Engineering and Computer Science, University of Trento, Italy
[2]Department of Electrical and Computer Engineering, National University of Singapore
[3]Research Center for Learning Science, Southeast University, Nanjing, China
[4]360 Artificial Intelligence Institute, China
{wei.wang,yan.yan,niculae.sebe}@unitn.it {elefjia,eleyans}@nus.edu.sg
zhen.cui@seu.edu.cn shuxb104@gmail.com

Abstract

Modeling the aging process of human face is important for cross-age face verification and recognition. In this paper, we introduce a recurrent face aging (RFA) framework based on a recurrent neural network which can identify the ages of people from 0 to 80. Due to the lack of labeled face data of the same person captured in a long range of ages, traditional face aging models usually split the ages into discrete groups and learn a one-step face feature transformation for each pair of adjacent age groups. However, those methods neglect the in-between evolving states between the adjacent age groups and the synthesized faces often suffer from severe ghosting artifacts. Since human face aging is a smooth progression, it is more appropriate to age the face by going through smooth transition states. In this way, the ghosting artifacts can be effectively eliminated and the intermediate aged faces between two discrete age groups can also be obtained. Towards this target, we employ a two-layer gated recurrent unit as the basic recurrent module whose bottom layer encodes a young face to a latent representation and the top layer decodes the representation to a corresponding older face. The experimental results demonstrate our proposed RFA provides better aging faces over other state-of-the-art age progression methods.

1. Introduction

Face aging, also known as age progression, is attracting more and more research interest. It has wide applications in various domains including cross-age face verification [22] and finding lost children. In recent years, face aging has witnessed various breakthroughs and a number of face aging models have been proposed [7]. Face aging, however, is still a very challenging task in practice for various reasons. First, faces may have many different expressions and lighting conditions, which pose great challenges to modeling the aging patterns. Besides, the training data are usually very limited and the face images for the same person only cover a narrow range of ages. Moreover, the face aging process also depends on the environment and genes which are hard to model.

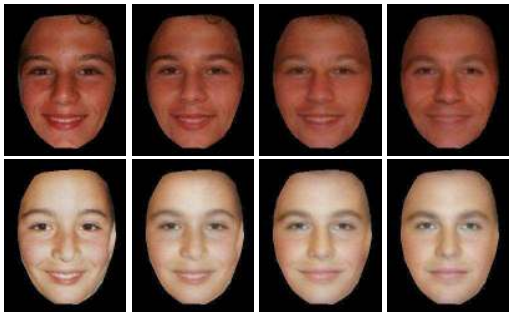


Figure 1. Exemplars of face aging from teenager to adult generated by our model. The left most column shows the input, and the other columns show the generated aged faces.

Generally, face aging follows some common patterns of the human aging process. For kids, the main appearance change is the shape change caused by cranium growth. For adults, the appearance change is mainly reflected by wrinkles [30]. Various face aging approaches were proposed to model such dynamic aging patterns, which can be roughly divided into two types [30], namely, prototype approaches [17, 34] and physical model approaches [31, 33]. The physical model approaches employ parametric models to simulate face aging by modeling the aging mechanisms of muscles, skins, or cranium. However, those approaches are very complex and computationally expensive, and they require a large number of face sequences of the same person

*J. Feng is supported by NUS startup grant R–263–000–C08–133.

covering a wide range of ages. However, few of the current face aging datasets can provide sufficient data. In contrast, the prototype approach [17] does not require face sequences of the same person with continuous ages. The prototype approach models face aging using a non-parametric model. First, all the available faces are divided into discrete age groups, and an average face within each age group is computed as a prior. The difference between the average faces is treated as the aging pattern and the pattern is transferred to each individual face to produce an aged face. However, the prototype approaches totally discard the personalized information and all the people share the same aging pattern. Moreover, regardless of the model type, all those methods perform a one-step transformation from one age group to another by learning a single mapping function. Thus, the one-step mapping function typically fails to capture the dynamics of the in-between face sequence between adjacent age groups.

To model the complex yet smooth dynamics of face aging, we propose a recurrent face aging (RFA) framework. Our RFA is based on a recurrent neural network (RNN), and it transforms a face smoothly across different ages by modeling the intermediate transition states. Different from the one-step mapping function used in the previous prototype approaches, our RFA framework can generate the fine-grained in-between faces. Similar to the prototype approaches, our model only requires the short-term faces of the same person which can cover two adjacent groups. This setting effectively alleviates the issue caused by data insufficiency of the long-term face sequences. Similar to the prototype approaches, we divide the faces of each gender into 9 age groups (*e.g.*, $1-5$, $5-10$, ...), as shown in Fig. 2. We adopt a recurrent neural network (RNN) to learn the aging pattern for every two adjacent age groups. Then all the RNNs are concatenated to form the complete face aging framework. One of the most attractive advantage of RNN [26] is its ability to memorize the previous states and allows information to persist. Thus, the RFA framework can age the face gradually while preserving the identity of the face by memorizing the previous faces.

The collected face images, however, usually have various expressions. Mild expressions can have a drastic effect in face analysis methods [35, 40], specifically in the position of the landmarks. Thus, we need to normalize the faces before aging them. Another concern in aging real faces is the presence of various lighting conditions. Such lighting inconsistency can manipulate a large portion of pixels. It is likely for a face aging model to learn the lighting change instead of the aging patterns if we learn the pixel-to-pixel mappings. Besides, there is possibly much noise in an image. For example, it is very common that forehead is occluded by hair. The shapes of the wrinkles are also diverse for different people. Thus, the noise (*e.g.*, the small and detailed wrinkles) should be filtered out while the regular shading (*e.g.*, the shading around the mouth) and texture information should be kept in the training phase. The eigenfaces [37] are very robust to noise as the they capture the global structure information. Thus, after obtaining the normalized images, we project the images to the egienface space and take their low rank coefficients as the image representation input to the RFA framework. After obtaining the predicted low rank aged face, we synthesize the textures by transferring the textures from its nearest neighbor in the eigenface space. The textures of the in-between faces are synthesized by combining the textures of the young face and the textures of the nearest neighbour. As shown in Fig. 1, we can generate realistic old faces with detailed textures.

To summarize, our paper makes following contributions: (1) We propose a smooth face aging process between each neighbouring groups with RNN network; (2) Our method can generate smooth intermediate faces and it handles the ghosting artifacts properly.

2. Related Work

2.1. Face aging

Face aging models can be roughly divided into prototype approaches and physical model approaches [7]. The prototype approaches [34] aim at constructing an average face as prototypes for the young and old groups, and transferring the texture difference between the prototypes to the test image. The state-of-the-art prototype method [17] improves the result by replacing the average face with a relighted average face whose lighting condition can be tuned to be the same with the input. Apart from the lighting considerations, the geometry transformation is implemented using optical flow.

However, the limitation of the prototype models still exists: face texture changes are the same for different inputs. Besides, the detailed texture information (*e.g.*, wrinkles) is averaged out. Recently, a coupled dictionary learning (CDL) model [27] was proposed. Similar to [39], the CDL model learns a dictionary for each age group. This model assumes that the sparse coefficients of the same person remain the same across the dictionaries. Thus, the aging patterns are encoded by the dictionary bases. Every two neighbouring dictionaries are learned jointly. Besides, the reconstruction error is regarded as the personalized information and it is added into the synthesized aged face directly. However, this method still has ghost artifacts as the reconstruction residual does not evolve over time.

2.2. Recurrent neural network

Traditional RNN can learn complex dynamics by mapping the input sequence to a sequence of hidden variables. By passing the hidden variables recursively to the repeat-

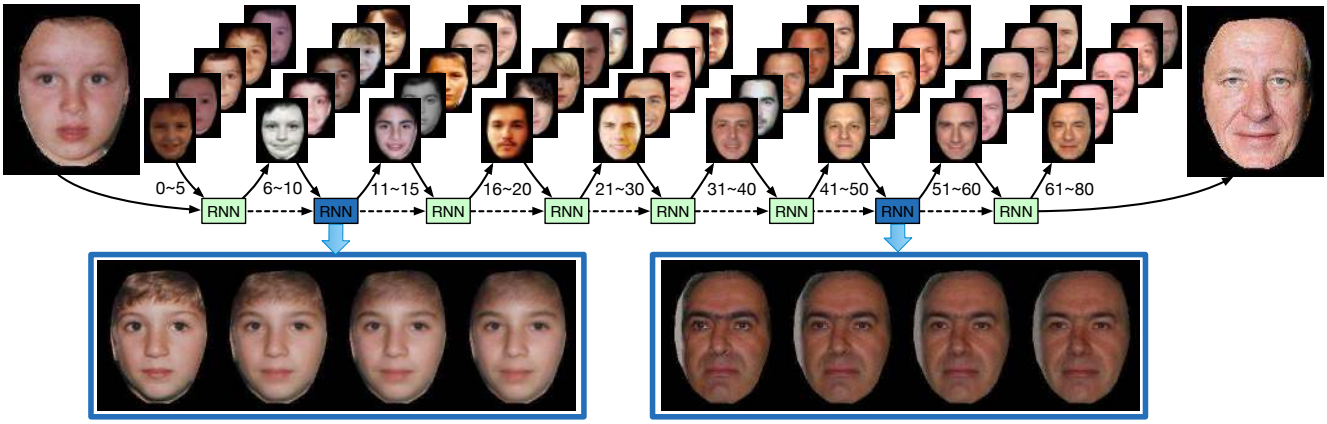


Figure 2. Our recurrent face aging (RFA) framework exploits a RNN to model the aging pattern between the neighbouring age groups. Our RFA generates the aged face by referring to the memory of the previous faces. The intermediate transition faces between every two neighbouring age groups can also be synthesized. Finally, all these networks are concatenated to form RFA framework.

ing module in the network, RNN is able to memorize the previous information. Thus, RNN performs well in dealing with sequential data which have dependencies. In the past few years, RNN has been successfully applied to a variety of natural language processing and image processing tasks, such as speech recognition [9], machine translation [1], hand-writing recognition [10], image caption generation [14] and video to text description [38].

As a special type of RNN, Long Short Term Memory (LSTM) networks are explicitly designed to tackle the long-term dependency problem. LSTM [8] was firstly introduced for speech recognition problem [12], where the memory cells enable LSTM network to process sequential data with dependencies. The key idea behind LSTM is its memory state and four gates which can control the information flow inside the unit adaptively. Given the success of LSTM, many LSTM variants are explored, such as the Gated Recurrent Unit (GRU) introduced by Cho *et al*. [5]. As shown in Fig. 5, GRU is a simplified version of LSTM. Given the various RNN variants, Greff *et al*. conducted thorough search in order to find out the optimal RNN structure [11]. The research revealed that GRU outperforms LSTM on almost all the tasks.

3. Recurrent Face Aging

3.1. Overview

Our model conducts face aging in the following two steps. The first step is face normalization and the second step is aging pattern learning. We crawl the face images from the Web, as well as the available databases. The details of image collection are in Section 4.1. As the face images are collected in the wild, they have various expressions.

There are various ways to normalize the faces, such as warping a face to the average face by matching the detected landmarks [36, 40] through interpolation, or utilizing optical flow [16]. As shown in Fig. 3, aligning a face to the

Figure 3. Landmark matching method VS optical flow method [16]

average position of the landmarks via interpolation twist the face. To avoid this undesirable effect, we use optical flow for face normalization, which preserves wrinkles well as shown in Fig. 3. In the eigenface space, the face images can be normalized as desired and important details can be well preserved. More details of face normalization are given in Section 3.2. Based on the learned eigenfaces, a robust image representation can be obtained by projecting the image to the eigenfaces. These image representations are then fed into our RFA framework.

We exploit RNN to learn the aging patterns between the neighbouring groups. Although many models can be used to learn the smooth transformation, such as LSTM [8], GRU [11], as well as their variants, we employ GRU to learn the aging patterns because of its simple structure and superior performance [11]. As shown in Fig. 5, a two-layer GRU is built as the basic recurrent module. The top GRU has the same structure as the bottom GRU with the difference in the dimensions of the hidden state $\widehat{h}_t$.

3.2. Intrinsic face normalization

The most important factor to be considered in face normalization is to preserve the intrinsic age information, such that one can normalize the face group-wisely by leveraging the faces within the same age group. In this way, the age-specific information can be maintained. For instance, the eyes of children are usually larger than the eyes of old people, as shown in Fig. 2. These characteristics could be well preserved by the eigenfaces. Another factor that needs to be considered is the smoothness of age progression between the adjacent age groups. Instead of normalizing each

face group independently, we normalize the faces from every two adjacent age groups jointly. A shared egienface space can be learned for each pair of groups. Then a smooth transformation can be learned in the shared eigenface space.

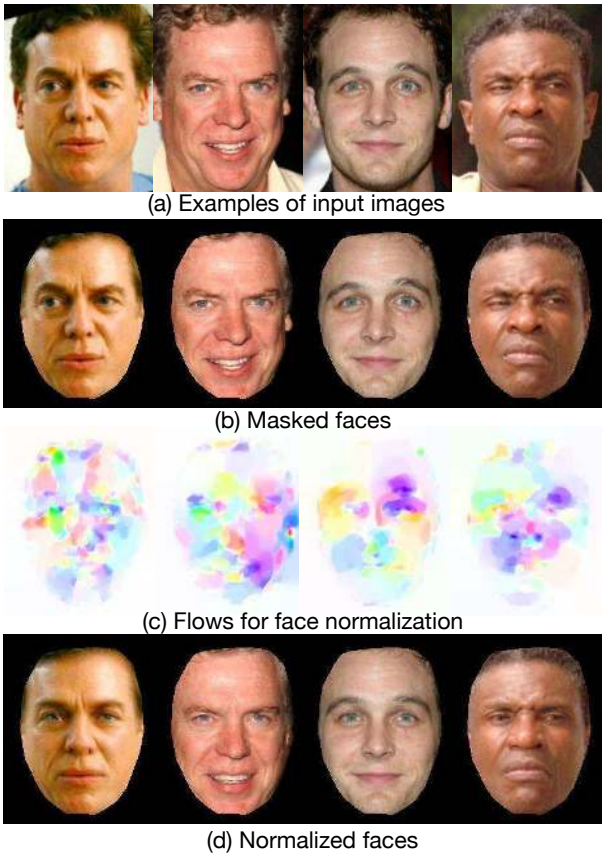


Figure 4. Intrinsic face normalization. (a) Examples of input images. (b) Masked images. (c) Estimated flow for face normalization. (d) Normalized faces with the estimated optical flow.

We optimize the eigenfaces and optical flow estimation iteratively. First, we stack the images column-wisely into the matrix $\mathbf{M}=[\mathbf{I}_1,...,\mathbf{I}_n]$. Here $\mathbf{I}$ denotes an image. Then we implement singular value decomposition on $\mathbf{M}$: $\mathbf{M}=\mathbf{U}\mathbf{S}\mathbf{V}^T$. We keep the top k eigenvectors in $\mathbf{U}$ and denote them as $\mathbf{H}=\mathbf{U}(:,1{:}k)$. We reconstruct image $\mathbf{I}$ in the low rank eigenface space as $\mathbf{I}'=\mathbf{H}(\mathbf{H}^T\mathbf{I})$ where $\mathbf{H}^T\mathbf{I}$ means projecting the image $\mathbf{I}$ to the eigenface space $\mathbf{H}$. Then the optical flow from $\mathbf{I}'$ to $\mathbf{I}$ can be calculated, and we can get $\widehat{\mathbf{I}}'$ by warping $\mathbf{I}$ to $\mathbf{I}'$ reversely using the optical flow. As the optical flow can not recover the images perfectly, $\widehat{\mathbf{I}}'$ and $\mathbf{I}'$ are not exactly the same and $\widehat{\mathbf{I}}'$ has ghost artifacts. To remove the ghost artifacts, we reset $\mathbf{M}=[\widehat{\mathbf{I}}_1,...,\widehat{\mathbf{I}}_n]$ and repeat the process above until convergence. In each new face normalization process, we progressively increase the number of eigenvectors.

We start the process from $k=4$ and terminate the process when $k=80$. Fig. 4 shows the face normalization results. It is worth noting that the expressions of the 4 people are normalized (*e.g.*, the eyeballs of the first image come to the middle; the mouth of the second image becomes horizontal; the wide-open eyes of the third image become smaller, and the closed eyes of the forth image are open).

3.3. Problem formulation

Let $\mathbf{H}^{(k)}$ represent the shared egienfaces for age group k and $k{+}1$, where each column in $\mathbf{H}^{(k)}$ denotes one eigenface. The columns in $\mathbf{H}^{(k)}$ are unit vectors and they are orthogonal to each other. Let $\mathbf{\Lambda}_k=(\lambda_1,\lambda_2,...,\lambda_n)^T$ denote the eigenvalues of the eigenfaces. Let $\mathbf{I}_y$ be the low rank young face, $\mathbf{I}_o$ be the low rank image of the ground truth of the old face, $\mathbf{I}'_o$ be the predicted low rank old face, and $\mathbf{x}_y$, $\mathbf{x}_o$, $\mathbf{x}'_o$ be their coefficients in the eigenface space. We expect the predicted image to be as similar as possible to the ground truth image. Therefore we define the following loss function:

$$\|\mathbf{I}_o-\mathbf{I}'_o\|_F^2=\|\mathbf{H}^{(k)}\mathbf{x}_o-\mathbf{H}^{(k)}\mathbf{x}'_o\|_F^2=\|\mathbf{x}_o-\mathbf{x}'_o\|_F^2. \quad (1)$$

During the face normalization process, we observe that the first 4 eigenvalues occupy more than 60% of the total energy. The previous studies [16] revealed that the first 4 eigenfaces correspond to the lighting effect of the face while the others correspond to the texture. Thus, in order to keep the illumination consistency between the source and target images, we transfer the first 4 coefficients directly from the young image to the predicted old image. The high rank coefficients mainly preserve the texture information. We rely on the high rank coefficients to learn the aging patterns. We visualize the high rank eigenfaces and find that these eigenfaces capture different texture information (*e.g.*, beard, open mouth with teeth and closed mouth). Here we normalize the distribution over these eigenfaces and propose the following loss function:

$$J=\|(\mathbf{x}_o-\mathbf{x}'_o)\odot(\frac{1}{\lambda_1},...,\frac{1}{\lambda_n})^T\|_F^2. \quad (2)$$

To optimize the objective function, we adopt RNN to learn the aging patterns as follows:

$$\mathbf{x}_y\odot(\frac{1}{\lambda_1},...,\frac{1}{\lambda_n})^T \xrightarrow{RNN} \mathbf{x}'_o\odot(\frac{1}{\lambda_1},...,\frac{1}{\lambda_n})^T. \quad (3)$$

The $\odot$ in Eq. 2 and Eq. 3 is an element-wise multiplication to scale the samples to the interval $[-1,1]$ by dividing the i-th element of $\mathbf{x}$ by λ_i.

3.4. Recurrent age progression

The two-layer GRU is more flexible compared with the single GRU. As shown in Fig. 5, the bottom GRU first encodes the input face to a hidden high dimensional variable. Then the top GRU decodes the hidden high dimensional state to an aged face. The hidden states are initialized with zeros. Using high dimension could boost its capability to encode complex high dimensional signals. The difference between the output and the ground truth aged face is calculated as the loss. Different weights are assigned for the loss.

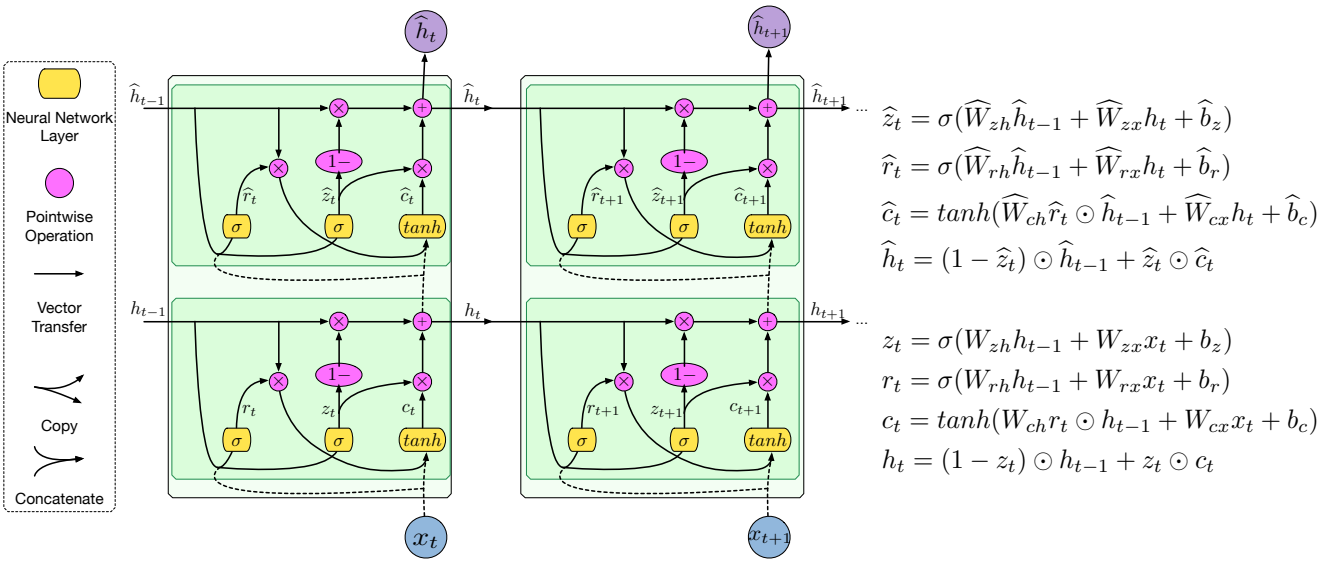


Figure 5. Recurrent face aging framework with two-layer gated recurrent unit

The loss in the last recurrence step has the largest weights as we take the output of the last step as the aged face. Smaller weights are set for the first several recurrence steps. These losses will guide the system to age the face slowly.

The equation shown in Fig. 5 is the model of the two-layer GRU, where σ is the logistic sigmoid function. $[\mathbf{W}_z, \mathbf{W}_r, \mathbf{W}]$ are weight matrices and $[\mathbf{b}_z, \mathbf{b}_r, \mathbf{b}_c]$ are bias terms which need to be learned. Each GRU has two gates and one hidden state.

The *reset* gate $\mathbf{r}_t$ decides whether the information of previous faces should be ignored. If $\mathbf{r}_t$ is close to 0, the previous face information is forced to be discarded, and the unit will focus on the current input face only. This gate allows the unit to remember or drop the irrelevant face information.

The *update* gate $\mathbf{z}_t$ controls the amount of face information that could be transferred from the previous state to the current state. This update gate works like the forget and input gates. Instead of calculating the value of forget and input gates separately like LSTM, the update gate in GRU calculates them together with $\mathbf{z}_t$ and $1 - \mathbf{z}_t$. This setting means that the unit only accepts the new input face when it forgets something of the previous faces. The update gate acts similarly to the memory cell in the LSTM.

c_t is the new face candidate created by a *tanh* layer that could be added to the current face. Then the face candidate is merged with previous face information to form a new face (*hidden state*) with the wights generated by the update gate:

$$\mathbf{h}_t{=}(1{-}\mathbf{z}_t){\odot}\mathbf{h}_{t-1}{+}\mathbf{z}_t{\odot}\mathbf{c}_t. \quad (4)$$

The system has short-term memory and ignores the previous faces if the reset gate is activated all the time. If the update gate is always inactivated, the system can have long-term memory and all the previous faces will be memorized.

In our RFA framework, RNN acts as a refinement process which transforms the young face *slowly* to the aged face. In our settings, each basic unit will iterate for 3 times. The **input series** $[\mathbf{x}_1, ..., \mathbf{x}_n]$ is the replicates of the young face. The **loss** is calculated after each recurrence. We expect to age the face gradually. In other words, the in-between faces should become more similar to the target face after each iteration. Thus, the loss between the in-between faces and the target faces should become smaller gradually in the training process. To meet this requirement, we set a series of weights $\mathbf{w}{=}[0.1, 1, 10]$ for the loss. The weight increases monotonically as we expect that the faces could be transformed to the target face gradually. We assign the largest weight for the loss of the output face. For the transformation from group k to $k{+}1$, we obtain the following loss function:

$$J = \sum_{t=1}^{n} \mathbf{w}_t \|(\mathbf{x}_{k+1} - \widehat{\mathbf{h}}_t) \odot (1/\mathbf{\Lambda}_k)\|_F^2, \quad (5)$$

where $\mathbf{x}_{k+1}$ represents the target image in group $k{+}1$, $\widehat{\mathbf{h}}_t$ is the predicted in-between states during the recurrent training process, and $\mathbf{w}_t$ is the weight for recurrence step t.

3.5. Follow-up operations

For each pair of neighbouring RNNs, their corresponding low-rank egienface spaces are different. Thus, the output of the previous RNN can not be used as input to the following RNN directly. We rely on the following formula to transform the output $(\mathbf{x}_{k+1})$ of k-th RNN to the input $(\overline{\mathbf{x}}_{k+1})$ of $(k{+}1)$-th RNN.

$$\overline{\mathbf{x}}_{k+1}{=}\mathbf{U}_{k+1}(\mathbf{U}_k\mathbf{x}_{k+1}){=}(\mathbf{U}_{k+1}\mathbf{U}_k)\mathbf{x}_{k+1}{=}\overline{\mathbf{U}}\mathbf{x}_{k+1} \quad (6)$$

where $\mathbf{x}_{k+1}$ is a column vector and it is the output of the k-th RNN. $\mathbf{U}_k\mathbf{x}_{k+1}$ is its low rank image. Then the operation $\mathbf{U}_{k+1}(\mathbf{U}_k\mathbf{x}_{k+1})$ reprojects the image to the eigenface space of $k{+}1$-th RNN. This transformation can be integrated into

the RNN framework. As shown in Fig. 5, the term $\mathbf{W}\mathbf{x}_t$ in the first three equations can be transformed as following:

$$\mathbf{W}\mathbf{x}_t = \mathbf{W}(\overline{\mathbf{U}}\mathbf{x}_k) = (\mathbf{W}\overline{\mathbf{U}})\mathbf{x}_k = \overline{\mathbf{W}}\mathbf{x}_k. \quad (7)$$

After obtaining the low rank face $\mathbf{I}'$. The next step is to transfer the detailed features from its nearest neighbour. We find the nearest neighbour of $\mathbf{I}'$ in the eigenface space which is denoted as $\mathbf{J}$. Feature transfer from its nearest neighbour $\mathbf{J}$ to $\mathbf{I}'$ is illustrated in Fig. 6.

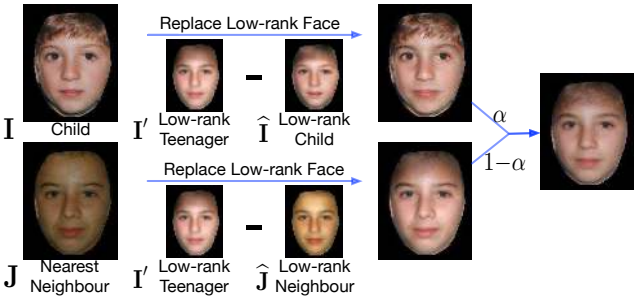


Figure 6. Feature transfer from the nearest neighbour.

First, the input face is warped to the coordinates of $\mathbf{I}'$, and obtain $\mathbf{I}$. Then we replace the low rank face of the $\mathbf{I}$ with $\mathbf{I}'$. Thus, the detailed texture of the young face is preserved. After replacing the low rank face of $\mathbf{J}$ with $\mathbf{I}'$, the aged face can be synthesized by linearly combining the two faces.

4. Experiments

In this section, we first describe the implementation details of data collection and image pre-processing, followed by introducing the implementation details of the RFA framework. Then we will show the qualitative experimental results, as well as the quantitative evaluations.

4.1. Data collection

We collect face images according to a celebrity list which contains 3,561 celebrities from the dataset of Labeled Faces in the Wild (LFW) [13]. We collect 163,810 images from Google and Bing image search engine where 3,240 celebrities have the photos which cover different age groups. We also use the images from the Morph Aging Dataset [23] and Cross-Age Celebrity Dataset (CACD) [4]. The Morph Aging Dataset contains 13,000 people with 55,134 images. The CACD dataset contains 163,446 photos of 2,000 people. Both datasets contain multiple images for each person which cover different age groups. In order to ensure the high quality of the data, we remove the images which have large poses (greater than 30 degrees in yaw and pitch angles). For each crawled image, the groundtruth of its age is estimated by an off-the-shell age estimator [19]. Then we manually checked the accuracy of the estimated age.

Finally, we have 4,371 photos for male and 6,264 photos for female in total. After dividing the data into 9 age groups for both male and female: 0-5, 6-10, 11-15, 16-20, 21-30, 31-40, 41-50, 51-60, 61-80, we obtain 2,611 image pairs which covers two neighbour age groups for male and 3,821 pairs for female in total.

4.2. Implementation

We follow a similar pipeline as [15] for image pre-processing which includes face landmark detection, pose estimation, and masking the images. After detecting the 66 facial landmarks using the model provided in [40], the faces are aligned according to the centers of eyes and mouth. In the intrinsic face normalization process, we revise the optical flow implementation introduced by Liu [20] to calculate the flow because of its superior performance [16, 17]. We follow [16] to set the parameters of the flow algorithm.

After performing face normalization, we calculate 80 low rank eigenfaces for every two neighbouring age groups with respect to each channel of RGB. Then we concatenate the coefficients and get d=240 dimensional representations as the input for the two-layer GRU. As shown in Fig. 5, the dimension of the hidden unit of the bottom GRU can be set to different values (*e.g.*, $d\times k$) from the input dimension d. To strengthen the encoding capability for various faces, we set k=15, which can generate satisfied in-between faces and also consumes less training time (around half an hour for each RNN). The hidden unit dimension of top GRU is set to be the same dimension as the input since it needs to decode the signal to the same dimension of the input signal. The RNN is implemented in Theano [2].

4.3. Qualitative comparison

We compare the performance of our method with another two face aging models, which are the coupled dictionary learning (CDL) model [27] and Face Transformer (FT) demo (`http://cherry.dcs.aber.ac.uk/Transformer/`). CDL defines the same 9 age groups as ours. The FT Demo has fewer age groups: Baby, Child, Teenage, Young Adult and Older Adult. For fair comparison with FT Demo, we select the pairs from our dataset which have large age gaps such that the ages of the images can be consistent with the age groups in FT Demo. Some experimental results are shown in Fig. 7. We further compare our method with CDL on fine-grained age groups as shown in Fig. 8. The images from FG-NET database [18] are used as the test data. In Fig. 7, the aged faces in the green boxes are generated by our method. One can observe that the images generated by CDL and FT Demo suffer from the ghost artifacts. Our method can generate images with more realistic appearance. There are also some failure cases for our method. The images in the red boxes show the cases where the better results are generated by the other methods.

Our RFA remembers all the previous faces and generates the aged face by referring to its memory. After each iteration, the shape of the face changes slightly. This change

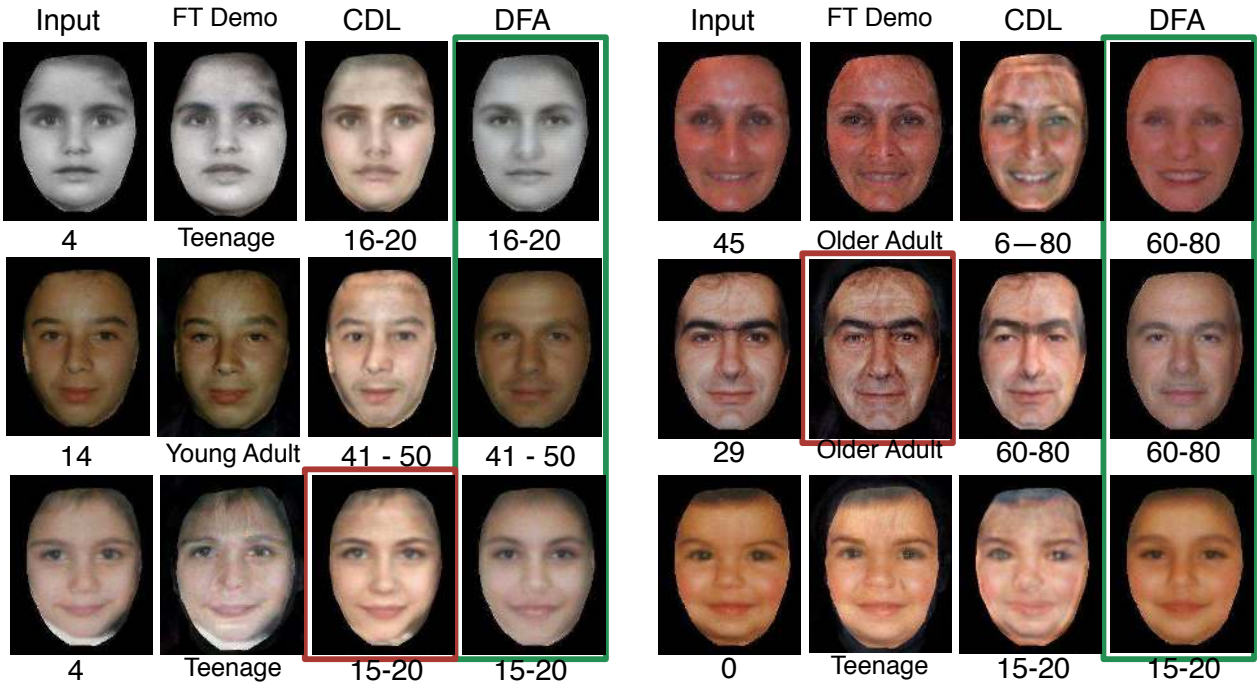


Figure 7. Face aging results comparison between FT Demo, CDL and our RFA method. The images in the green boxes are aged faces generated by our method. The images in the red boxes are the examples which are better than our method.

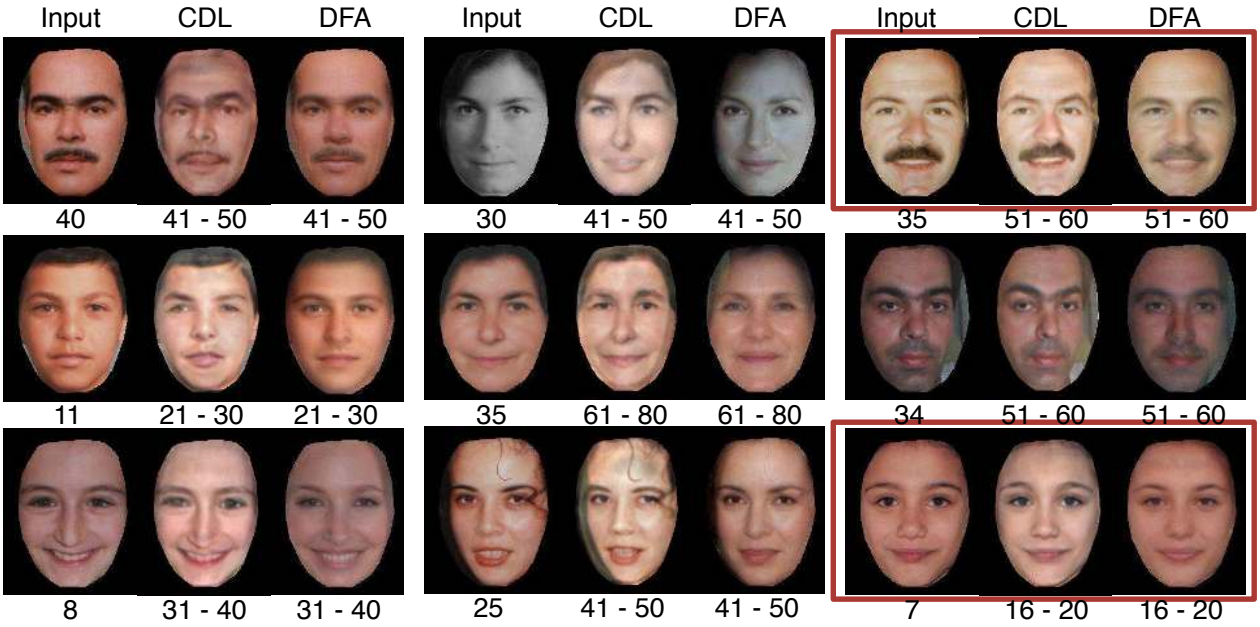
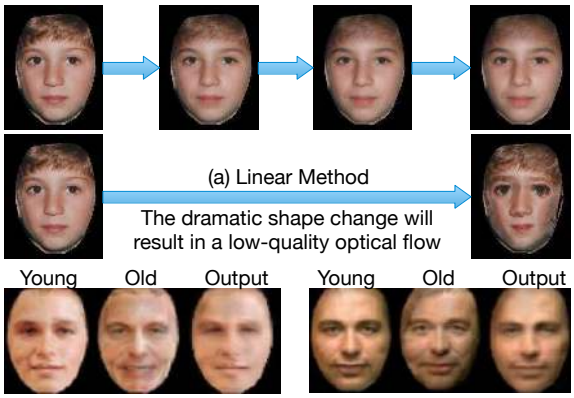


Figure 8. Comparison between Coupled Dictionary Learning (CDL) method [27] and our RFA method . Here we plot the aged results of 9 people. Each person has three images: the masked image which is the input, aged face generated with RFA, and aged face generated with CDL. The number shown below each image is the age or target age range of the person. For example, for the first group, the input image is of age 40, and the target age range is 41-50. We can observe that the aged face generated by our method matches the characteristics of the target age group well. For instance, the aged face in the first group (row 1, column1) gets some wrinkles, and his eyes become smaller during the aging process. But for some cases our aged faces are not so clear as the ones generated by CDL, such as the examples in the red boxes.



(b) VGG-S (c) Super-Resolution CNN

Figure 9. Comparison with one-step method.

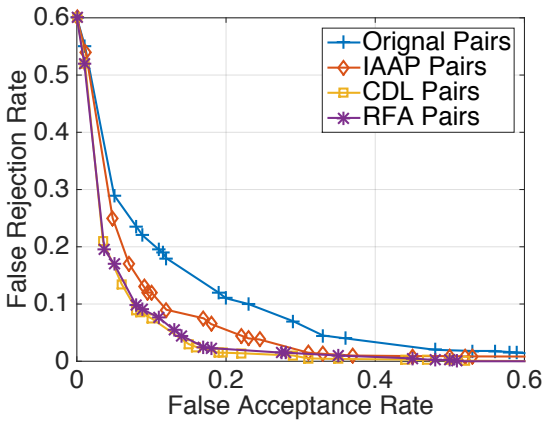


Figure 10. False acceptance rate vs false rejection rate curve

with well-controlled magnitude will result in a high-quality flow which makes the image look more realistic (getting rid of ghost artifacts). For the one-step method (*e.g.*, linear regression method), the shape and texture change is too dramatic. This leads to unnatural images with severe artifacts, as shown in Fig. 9 (a). We also compare our method with two popular Convolution Neural Networks (VGG-S [3], Super-Resolution CNN [6]). For VGG-S, we remove all the pooling layers and set the output of the last convolution layer to the aged face. However, these methods failed learning the wrinkles in the training phase, and their output are smoothed faces, as shown in Fig. 9 (b)(c).

4.4. Quantitative comparison with prior works

Several prior works released their best face aging results [17], [21], [24], [27], [31]. Shu *et al*. [27] summarized all the posted images, and found that there were 246 aged faces with 72 input images in total. We synthesize the aged face with the same age range of these methods for each input image. Similar to prior works, we evaluate our results through user study.

In user study, each subject views three images: the young image C, and the aged images B & A which are generated by other methods and our method respectively. We set two metrics for the evaluation, age accuracy and identity accuracy. Each subject is asked to evaluate the images based on these two metrics. Three types of scores are provided. If A is better, it gets a score of 1. If B is better, A gets a score of 0. If A and B are similar, then both of them get the score of 0.5. We invite 40 people to evaluate our results and get 9840 scores in total. The statistics of the scores are as follows: 58.67% of the votings think our result is better, 10.40% think the these results are equivalently good, and 30.92% think other results are better.

4.5. Evaluation on cross-age face verification

Many groups have made breakthroughs for face verification [25], [28], [29], [32]. We employ the deep Convolutional Neural Network model introduced in [28] for face verification. To evaluate the performance of our method for the cross-age face verification, we exploit FG-NET dataset which consists of 1,002 photos of 82 people as our input. We select the image pair whose the age gap is larger than 20 years. 916 image pairs are obtained in total. We further select 916 image pairs randomly from different people as negative pairs. We name these pairs as "Original Pairs". For each image pair, we synthesize the aged face of the younger one. Thus we get our synthesized image pair by replacing the young face with our synthesized aged face. We name these pairs as "RFA Pairs". We also synthesize the aged faces with CDL [27] method and illumination aware age progression (IAAP) method [17], and we name the synthesized pairs as "CDL Pairs" and "IAAP Pairs" respectively.

The false acceptance rate-false rejection rate (FAR-FRR) curve is available in Fig. 10. As shown in Fig. 10, our method has competitive performance compared with CDL and outperforms the other two methods.

5. Conclusion and Future Work

In this paper, we proposed a recurrent framework for face aging. By going through the smooth intermediate faces, the shape of the face evolves slowly and this leads to a high-quality optical flow. Then the synthesized faces derived from the optical flow are more realistic compared with the one-step methods. We exploit a very powerful two-layer GRU as our recurrent module. The bottom layer works as an encoder which can project the image to a high-dimension space and the top layer works as a decoder which decode the hidden variables to an aged face. This powerful structure can model very complex dynamic appearance changes. However, during the testing phase, the system requires the age of the input face which might be unavailable. In the future, we will integrate the age estimation model into our framework. Another consideration is that the edge of the generated aged face is not very sharp. We will explore other texture synthesizing methods to generate clear textures.

References

[1] M. Auli, M. Galley, C. Quirk, and G. Zweig. Joint language and translation modeling with recurrent neural networks. In *EMNLP*, volume 3, pages 1044–1054, 2013.

[2] F. Bastien, P. Lamblin, R. Pascanu, J. Bergstra, I. J. Goodfellow, A. Bergeron, N. Bouchard, and Y. Bengio. Theano: new features and speed improvements. Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop, 2012.

[3] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In *British Machine Vision Conference*, 2014.

[4] B.-C. Chen, C.-S. Chen, and W. H. Hsu. Cross-age reference coding for age-invariant face recognition and retrieval. In *ECCV*. 2014.

[5] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[6] C. Dong, C. C. Loy, K. He, and X. Tang. Learning a deep convolutional network for image super-resolution. In *ECCV*, pages 184–199. Springer, 2014.

[7] Y. Fu, G. Guo, and T. S. Huang. Age synthesis and estimation via faces: A survey. *TPAMI*, 32(11):1955–1976, 2010.

[8] A. Graves et al. *Supervised sequence labelling with recurrent neural networks*. Springer, 2012.

[9] A. Graves, A.-R. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *ICASSP*, pages 6645–6649. IEEE, 2013.

[10] A. Graves and J. Schmidhuber. Offline handwriting recognition with multidimensional recurrent neural networks. In *NIPS*, 2009.

[11] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber. Lstm: A search space odyssey. *arXiv preprint arXiv:1503.04069*, 2015.

[12] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[13] 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, University of Massachusetts, Amherst, 2007.

[14] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015.

[15] I. Kemelmacher-Shlizerman and S. M. Seitz. Face reconstruction in the wild. In *ICCV*, 2011.

[16] I. Kemelmacher-Shlizerman and S. M. Seitz. Collection flow. In *CVPR*, 2012.

[17] I. Kemelmacher-Shlizerman, S. Suwajanakorn, and S. M. Seitz. Illumination-aware age progression. In *CVPR*, 2014.

[18] A. Lanitis, C. J. Taylor, and T. F. Cootes. Toward automatic simulation of aging effects on face images. *TPAMI*, 24(4):442–455, 2002.

[19] C. Li, Q. Liu, J. Liu, and H. Lu. Learning ordinal discriminative features for age estimation. In *CVPR*, 2012.

[20] C. Liu. *Beyond pixels: exploring new representations and applications for motion analysis*. PhD thesis, Citeseer, 2009.

[21] U. Park, Y. Tong, and A. K. Jain. Face recognition with temporal invariance: A 3d aging model. In *Automatic Face & Gesture Recognition*, pages 1–7. IEEE, 2008.

[22] U. Park, Y. Tong, and A. K. Jain. Age-invariant face recognition. *TPAMI*, 32(5):947–954, 2010.

[23] K. Ricanek Jr and T. Tesafaye. Morph: A longitudinal image database of normal adult age-progression. In *Automatic Face and Gesture Recognition*, pages 341–345. IEEE, 2006.

[24] K. Scherbaum, M. Sunkel, H.-P. Seidel, and V. Blanz. Prediction of individual non-linear aging trajectories of faces. In *Computer Graphics Forum*, volume 26, pages 285–294. Wiley Online Library, 2007.

[25] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, 2015.

[26] M. Schuster and K. K. Paliwal. Bidirectional recurrent neural networks. *Signal Processing*, 45(11):2673–2681, 1997.

[27] X. Shu, J. Tang, H. Lai, L. Liu, and S. Yan. Personalized age progression with aging dictionary. In *ICCV*, 2015.

[28] Y. Sun, Y. Chen, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. In *NIPS*, 2014.

[29] Y. Sun, D. Liang, X. Wang, and X. Tang. Deepid3: Face recognition with very deep neural networks. *arXiv preprint arXiv:1502.00873*, 2015.

[30] J. Suo, X. Chen, S. Shan, W. Gao, and Q. Dai. A concatenational graph evolution aging model. *TPAMI*, 34(11):2083–2096, 2012.

[31] J. Suo, S.-C. Zhu, S. Shan, and X. Chen. A compositional and dynamic model for face aging. *TPAMI*, 32(3):385–401, 2010.

[32] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *CVPR*, 2014.

[33] Y. Tazoe, H. Gohara, A. Maejima, and S. Morishima. Facial aging simulator considering geometry and patch-tiled texture. In *ACM SIGGRAPH 2012 Posters*, 2012.

[34] B. Tiddeman, M. Burt, and D. Perrett. Prototyping and transforming facial textures for perception research. *Computer Graphics and Applications*, 21(5):42–50, 2001.

[35] S. Tulyakov, X. Alameda-Pineda, E. Ricci, L. Yin, J. F. Cohn, and N. Sebe. Self-Adaptive Matrix Completion for Heart Rate Estimation from Face Videos under Realistic Condition. In *CVPR*, 2016.

[36] S. Tulyakov and N. Sebe. Regressing a 3d face shape from a single image. In *ICCV*, 2015.

[37] M. A. Turk and A. P. Pentland. Face recognition using eigenfaces. In *CVPR*, 1991.

[38] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko. Sequence to sequence-video to text. In *ICCV*, 2015.

[39] W. Wang, Y. Yan, S. Winkler, and N. Sebe. Category specific dictionary learning for attribute specific feature selection. *TIP*, 25(3):1465–1478, 2016.

[40] X. Xiong and F. De la Torre. Supervised descent method and its applications to face alignment. In *CVPR*, 2013.