Cycle-GAN based Aging Progression/Regression Model

Shuying Guan
University of Massachusetts Amherst
sguan@umass.edu

Zhipeng Tang
University of Massachusetts Amherst
zhipengtang@umass.edu

Abstract

If you were given a young person's picture, would it be possible to generate the picture of the same person after she/he gets old? This is the classic problem named Age Progression, whose goal is to generate facial images with aging effect while preserve the facial recognition features. Similarly, Age Regression is the reverse problem which translates old faces to young faces. Both of them are difficult problems since it is hard to trace people's facial changes over a long time span. Most of models using traditional methods suffer from paired facial image shortage. However, the recent development of Generative Adversarial Networks (GANs) has paved a new way for addressing this problem. With the help of GANs, paired facial data becomes less important. We can just pour images from youth and old age groups into the GANs and let the battle of generators and discriminators take care of the mapping problem. In 2018, a new variant of GANs called cycle-GANs was proposed. It takes advantage of the property of Cycle Consistency to maintain main facial features between mapping results in the case of lacking paired data. In this project, we propose a new model based on cycle-GAN to address age progression. Because of the dual feature of cycle-GANs, we also get a by-product model for age regression.

1. Introduction

Time never stops and waits for anybody. Aging is a natural and innate process of our bodies, which nobody can avoid. Although being old is not pleasant, it is still useful and interesting to predict how our faces will be like after we getting old. This problem can be named as *Age Progression*. Formally speaking, the goal of age progression is to generate facial images with aging effect while preserve the facial features [4, 22]. This problem can result in many practical applications, such as finding missing children and entertainment. A typical example is Oldify, an application in the Apple App Store, which can age your face and snap a video. This application by far is rated with 4 stars and ranked 43 in the entertainment sector. The reverse of Age Progression is called *Age Regression*, which removes the aging features from faces without changing facial features.

Thus far, there are already many models for the age progression problem. They can be divided into three major categories: physical-based, data-driven prototype-based, and deep learning-based [21]. Physical-based models put efforts in modeling anatomical structures of human faces across ages and use these structures to simulate the changes of facial appearance [12]. This kind of models dig the insight of face aging deeply but cannot generalize well. Data-driven prototype-based models often rely on averaging the faces across age groups and then synthesizing the average faces with personal facial features, which may lose facial personality and cause ghosting artifacts [21]. With the popularity of deep learning, more models based on neural network methods are proposed and become state-of-the-art. In traditional neural network driven methods, images over a long age span for each person are required to be collected as training data [22]. However, they are hard to be collected, and therefore their use are limited. However, the development of Generative Adversarial Networks (GANs), a system of two neural networks contesting with each other in a zero-sum game framework, bypasses this problem [20].

In this project, we propose a Cycle Generative Adversarial Networks (cycle-GAN) based Model for the Age Progression problem. Cycle-GAN[23], proposed by Zhu et al., is an variant of GAN which can convert one image from a source domain X to a target domain Y without paired training data. This is perfect for age progression, since it is very difficult to obtain aging images for one subject due to long time span. Cycle-GAN exploits the principle of "cycle consistency". It uses two dual mappings $G : X \rightarrow Y$ and $F : Y \rightarrow X$. The first mapping G maps image $x \in X$ to $\hat{y}$, i.e. $\hat{y} = G(x)$, where $\hat{y}$ is indistinguishable from $y \in Y$. Therefore, G can convert images from domain X to domain $\hat{Y}$ which has the identical distribution with Y. It seems that we can just rely on this mapping to translate a facial image to an aging facial image. However, using G solely cannot guarantee individual input and output are meaningfully paired. The networks may not realize they are supposed to maintain facial features for the generated images. Another fatal problem which may be caused by using G only is Mode Collapse [23], where generators only generate limited types of samples regardless of the input. To solve these two problems, translator $F : Y \rightarrow X$ is introduced to be the inverse function of G. In other words, $F : Y \rightarrow X$ is a mapping whose output $\hat{x} = F(\hat{y})$ should have the same distribution to X. We train G and F simultaneously to achieve $F(G(x)) \approx x$ and $G(F(y)) \approx y$. We define a *cycle consistency loss* to quantify these two optimization targets. Obviously, we also need two adversarial discriminators to help the distribution of outputs of G and F to be nearly identical to Y and X, respectively. We quantify these two goals as adversarial losses. Combining the cycle consistency loss and the adversarial losses yields the optimization target of cycle-GAN. We use two facial datasets, the Labelled Faces in the Wild [10] and the CACD2000 [2, 3], to train the model with the optimization target. After training, generator G becomes the neural network for age progression. Simultaneously, we get a by-product generator F which can deal with age regression.

The major contribution of our project is that we propose a new cycle-GAN-based model to tackle Age Progression and Age Regression simultaneously.

The rest of the paper is organized as follows: In section 2, the formal definition of the problem we address is described and the related work of the approach. In section 3, we describe the technical details of our model. In section 4, the experiment details and results of this model are described. In section 5, we conclude the paper and outlook the future work.

2. Related Work

The Age Progression problem we solve can be formally defined as follows [22]: Given a series of facial images X of a certain youth group, and another series of facial images Y of a certain old age group, we want to find a mapping G which can translate any image $x \in X$ to image $\hat{y} \in \hat{Y}$ where $\hat{Y}$ has the identical distribution with Y. Besides, the generated $\hat{y}$ should maintain the facial recognition features of the original x. In other words, if we have a facial recognition function $H(a, b)$ which takes in two facial images a and b and outputs the probability $p \in [0, 1]$ that x and y are the same person, we require that H(x, $\hat{y}$) is as large as possible, i.e. the output should approach to 1.

Similarly, the Age Regression problem can be formally defined as the reverse of the age progression problem: Given a series of facial images Y of a certain old age group, and another series of facial images X of a certain youth group, we want to find a mapping F which can translate any image $y \in Y$ to image $\hat{x} \in \hat{X}$ where $\hat{X}$ has the identical distribution with X. Besides, the generated $\hat{x}$ should maintain the facial recognition features of the original x. We can also use $H(a, b)$ to define the property of facial feature preservation.

Currently, the algorithms for Age Progression can be divided into three types: physical-based, data-driven prototype-based, and deep learning-based as described in section 1. [6] is a comprehensive survey on this topic. Physical-based model usually explores the change of the anatomy structures of human faces caused by aging, such as skin and facial muscle. The drawback of this type of model is complexity. They usually need to be fed by a large amount of training data [16, 17]. For data-driven prototype-based model, they usually average the faces in different age groups, and try to utilize the difference between these average faces [9]. However, this method may cause unreality of photos because it may ignore the different facial features of different people [6]. Recently, the deep learning based models become prevalent. In 2016, [18] proposed a recurrent face aging framework based on RNNs, a time sequence neural network model. This framework can provide a smooth transformation of faces across ages. In 2017, conditional GAN was introduced to age progression problem [1]. However, this model may be slow at test time, because it needs to solve a LBFGS optimization problem for each image [19]. [22] proposed an auto-encoder conditional GANs to encode the images to manifolds and then reconstruct them.

As for Age Regression, the current results are mostly based on physical models [13, 7]. They basically remove textures according to learned transformation over facial surfaces [22].

As mentioned in the previous paragraphs, researchers now focus on how to combine GANs with age progression. Generative Adversarial Nets(GANs) proposed by *Goodfellow et al.* contain generator and discriminator as two major components [8]. These two networks compete with each other. The generator aims to generate the fake images to fool the discriminator, while the discriminator gradually gains its capability of differentiating between true images and fake ones. They uses the dynamical loss from the feedback of discriminator, instead of using the naive pixel-wise loss function to indicate how real a generated image is (e.g. mean square error).

Mirza et al.[14] extended GAN to Conditional GAN (cGANs) by adding extra input Y into both discriminator and generator as conditions of the images to constrain the image. By inputting the conditional label Y in the network, the discriminator would distinguish not only fake images from real images, but also the images without certain feature from the images with regularized features. With the dynamically feedback loss from the discriminator, the generator could also learn to generate the images based on the input conditional labels. But when using conditional GAN, facial features would not be preserved. Thus, this is not an ideal model for age progression and age regression.

Based on cGAN(conditional GAN)[14], Invertible Conditional GANs (icGAN) was introduced by *Perarnau*[15]to achieve image transformation based on specific conditions. It proposed an encoder to map an image to its specific compressed representation Z and Y which we feed to the generator to get a similar image to the input. In cGANs, even though the sample Z was randomly generated, the label Y is given as an input. Therefore, the generated images would be randomly distributed with constrained label Y. Invertible Conditional GANs proposed mapping images to (Z,Y) pairs to constrain output images of the generator. However, IcGANs still have risk to mix people's face up, especially when the training data is large. Another shortcoming of icGAN is the facial features depend heavily on the training set. If the majority face in training set is people with white skin, the test result on people with other skin color could not be nicely generalized.

To use GANs on age progression, we need to change GANs to support unpaired image translation. *Liu et al.*[11] proposed Coupled Generative Adversarial Networks(CoGANs) which contain two shared weight generator and two shared weight discriminator to perform unpaired image to image translation. The shared weight forces the high-level semantics to be decoded in the same way in Generator 1 and Generator 2. However, CoGAN is only target for one constrain on the output, but could not keep another features of the images. In 2018, Cycle Generative Adversarial Networks (Cycle-GANs) is proposed to deal with unpaired image translation [23]. It trains two pairs of generators and discriminators together, and uses cycle consistency loss to preserve the features of images between translation.

3. Approach

3.1. Overview

In order to learn the mapping between domain X and domain Y without paired training example, two pairs of generators and discriminators, i.e. (G, D_Y) and (F, D_X), are required in Cycle-Consistent Adversarial Networks(cycleGAN). First, we divide training images into two datasets. The dataset X is for images of youth, and the dataset Y is for images of the old. For generator G, it generate fake $\hat{y_i}$ using real x_i, where x_i belongs to training dataset X, and discriminator D_Y would try to discriminate the generated fake $\hat{y_i}$ from true y_j from training dataset Y. Here, $\hat{y_i}$ shares the same facial features with x_i, which can be validated by a facial recognition function $H(x_i, \hat{y_i})$. However, x_i and true y_j would not necessarily be the image of the same person. In other words, true x_i and true y_j can be unpaired.

Same goes another component. for generator F, it generates fake $\hat{x_i}$ using real y_i from the training dataset Y, and discriminator D_X would try to discriminate fake $\hat{x_i}$ from true x_j which belongs to the training dataset X. Similarly, $\hat{x_i}$ shares the same facial features with y_i, which can be validated by the same facial recognition function $H(\hat{x_i}, y_i)$. y_i and true x_j would not necessarily be the image of the same person, which can also be formalized with function H.

A key feature of cycle-GAN, which makes it stands out, is the cycle consistency. If the generators, both G and F, are good enough to do the right job, i.e if G can translate a young person's photo to the photo of the same person with aging effects while F can translate an old person's photo to the photo of the same person without aging features, the generated fake images produced by the composite of G and F would be as close to the original input image as possible. For example, if a real x_i from domain X is inputted to generator G, then the output of G will be the fake $\hat{y_i}$ in domain $\hat{Y}$ which has the same distribution as $\hat{Y}$. After that, if we input fake $\hat{y_i}$ input to generator F, then the output of F, i.e. fake $\hat{x_i}$, should as close to the originally input real x_i as possible.

To summarize, the loss for discriminators in cycle-GANs would be same as other GAN model. But for the generators, the loss would be the sum of the feedback of discriminators and the feedback of cycle consistency.

3.2. Architecture and Loss

As mentioned in the overview, the overall architecture is shown below in figure 1(a) [23]. The generator G would take in images from domain X and generates images in domain $\hat{Y}$, which has the same distribution as Y. the generator F would take in images from domain Y and generate images in domain $\hat{X}$, which has the same distribution as X. Discriminator D_X can take in real images from domain X and fake images generated from generator F and distinguish between them. Discriminator D_Y would take in real images from domain Y and fake images generated from generator G and distinguish between them.

When a real image x in domain X is input to generator G and output the generated fake image $\hat{y}$, and then input fake image $\hat{y}$ to generator F, the output generated fake image $\hat{x}$ should be as closer to the original input real image x as possible, as shown in the figure 1(b) and (c) [23]. This is the property of cycle consistency.

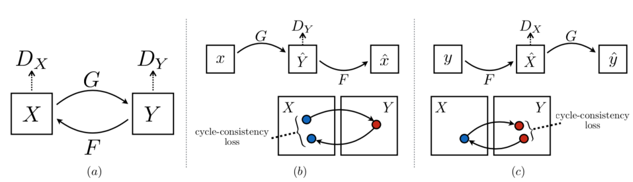

Figure 1. Architecture of cycleGAN [23]

In cycle-GANs, two types of loss function are introduced, namely *Adversarial Loss* and *Cycle Consistency Loss*. Adversarial loss aims at matching the generated fake images' distribution to the target domain's image distribution, while Cycle Consistency Loss aims at preventing generators G and F from contradicting each other [23].

The adversarial loss for generator $G : X \rightarrow Y$'s corresponding discriminator D_Y can be defined as follows:

$$\begin{aligned} L_{D_Y}(G, D_Y, X, Y) = \mathbb{E}_{y\sim p_{data}(y)}[D_Y(y)] \\ + \mathbb{E}_{x\sim p_{data}(x)}[1 - D_Y(G(x))] \end{aligned} \tag{1}$$

Generator G tries to generates image $\hat{y} = G(x)$ where $x \in X$. $\hat{y}$ is similar to images in Y. Discriminator D_Y tries to distinguish the generated image $\hat{y}$ from images from Y. $\mathbb{E}_{y\sim p_{data}(y)}[D_Y(y)]$ means D_Y should give large scores to real images from Y, and $\mathbb{E}_{x\sim p_{data}(x)}[1-D_Y(G(x))]$ means $D_Y(G(X))$ should be as small as possible.

Similarly, we can define $L_{GAN}(F, D_X, Y, X)$ to constrain the generator F and its corresponding discriminator D_X.

The cycle consistency loss is made to reduce the possible mappings of G between real image x and generated image $\hat{y}$. This is because, in a neural network, there are a huge number of possible mappings from X to Y. Only adversarial loss cannot guarantee that the learned mapping can translate the input x to desired $\hat{y}$. The cycle consistency property means for each image x, G can translate x to a fake image $\hat{y} = G(x)$. We require that, after we input $\hat{y}$ to generator F, the output $F(G(x))$ should be as close to x as possible, i.e. $x \rightarrow G(x) \rightarrow F(G(x)) \approx x$. Similarly, we also need $y \rightarrow F(y) \rightarrow G(F(y)) \approx y$ [23]. This cycle consistency loss can be formalized as

$$\begin{aligned} L_{CYC}(G, F) = \mathbb{E}_{x\sim p_{data}(x)}[\|F(G(x)) - x\|_1] \\ + \mathbb{E}_{y\sim p_{data}(y)}[\|G(F(y)) - y\|_1] \end{aligned} \tag{2}$$

Therefore, the full loss can be defined by adding all adversarial loss and cycle consistency loss together:

$$\begin{aligned} L(G, F, D_X, D_Y) = L_{GAN}(G, D_Y, X, Y) \\ + L_{GAN}(F, D_X, Y, X) \\ + L_{CYC}(G, F) \end{aligned} \tag{3}$$

While D_X and D_Y are aimed at maximizing the overall loss function because they want to distinguish between real and fake images, G and F are aimed at minimizing the loss function because they want to generate fake images which are almost real. So the optimization objective is as follows:

$$G^*, F^* = \arg\min_{G,F} \max_{D_X, D_Y} L(G, F, D_X, D_Y) \tag{4}$$

4. Experiment

We implemented the aforementioned model based on Tensorflow and Pytorch for prediction. We evaluate our proposed method on three dataset. The first one is non-human face dataset horse2zebra [5]. By using this exempliary dataset, we can evaluate the validation of the Cycle-GAN-based model. Then, we apply our model to human face datasets. The two datasets we use are Labelled Faces in the Wild (LFW) dataset [10] and Cross-age Celebrity Dataset (CACD2000, or simply CACD) [2, 3].

In subsection 4.1, we will introduce the horse2zebra, LFW, and CACD2000, respectively. In subsection 4.2, the implementation details are described. In subsection 4.3, we will show our model's results and give detailed analysis to them.

4.1. Datasets

In this section, we will describe the datasets we use to train the cycle-GAN-based model. The horse2zebra, LFW, and CACD2000 are described in subsection 4.1.1, 4.1.2, and 4.1.3, respectively.

4.1.1 Horse2zebra Dataset

Horse2zebra dataset includes the extracted horse and zebra images from ImageNet dataset[5]. ImageNet is an image dataset organized according to the WordNet hierarchy. Each meaningful concept in WordNet, possibly described by multiple words or word phrases, is called a *synonym set* or *synset*. There are more than 100,000 synsets in WordNet, majority of them are nouns (80,000+). In ImageNet, Images of each concept are quality-controlled and human-annotated. There are 1067 horse images and 1334 zebra images for training and 120 horse images and 140 zebra images for testing to perform experiment.

4.1.2 LFW Dataset

Labelled Faces in the Wild (LFW) dataset [10] is a large-scale face attributes dataset with of 13233 images of 5749 people. The images in this dataset cover large pose variations and background clutter. There are 1680 people with two or more images in the Labelled Faces in the Wild dataset [10]. LFW has 132 attributes annotations per image for image. We extract 905 images for the aged and 654 images for youth based on the attributes annotations and perform experiment on these two data domain.

4.1.3 CACD2000 Dataset

Cross-Age Celebrity Dataset (CACD) has 163,446 images from 2,000 celebrities automatically downloaded from the Internet. The images are collected by using search engine with the keywords of celebrity name and year (2004-2013). We can therefore infer the ages of the celebrities on the images by simply subtract the birth year from the year of which the photo was taken [2, 3].

In practice, during inspecting the images in the dataset, we find there are a certain number of mislabelled images in CACD. We try to filter out these images by hand. In order to accelerate the speed, we use only a small amount of data during training after data cleaning. We use 4650 images of young celebrities, and use 4148 images of old celebrities during training.

4.2. Implementation Details

As shown in the previous section, we have two pairs of generators and discriminators, i.e. (G, D_Y) and (F, D_X), in the model. Generator G's mission is to generate fake images $\hat{y}$ in domain $\hat{Y}$ based on real images x in domain X. Another generator F to generate fake images $\hat{x}$ in domain $\hat{X}$ based on real images y in domain Y. The discriminator D_X can differentiate fake images $\hat{x}$ from real images in domain X. Similarly, another discriminator D_Y can differentiate fake images $\hat{y}$ from real images in domain Y. In practice, we would use the same network structure for both generators and both discriminators.

We adopt the network from the original paper of cycle-GAN [23]. For generators G and F, we perform down-sampling for the first three layers, nine layers of residual blocks, and three layers of up-sampling. For discriminator, we performed 5 layers of down-sampling.

In the experiment, we add one more loss called *identity loss*. Identity loss hopes that if we throw an image x from X to generator F, the output will be as close to x as possible. Similarly, if we throw an image y from Y to generator G, the output will be as close to y as possible. This means the age progression generator will not change the image of the aged much, and the age regression generator will not change the image of the youth much. This loss is not formally and clearly introduced in the original cycle-GAN paper [23], but appears in the source code. After performing experiments with this new identity loss, we find that the result with identity loss seems slightly better. We think identity loss may have the functionality of constraining the background and other features so that they keep unchanged when transforming the image from one domain to another.

We also performed experiments with the batch size. We experiment with batch size 1 and 50. The output of batch size 1 and batch size 50 have huge difference. For batch size 50, the output images just got a bit more blur and have some strikes among the images. No evident translation shown in the output result. But for batch size 1, there is a great improvement of the result. Therefore, we chose batch size 1 as the parameter.

For generators, we did experiments on the base number of filter. We tested this hyper-parameter with 32 and 64. While 64 works fine, using 32 as the base number of filters in generator have better result on reduce the mistakenly adding stripes on background of horse images.

Instead of using generated fake images for training discriminator each time, we adopt fake image pool. When there are not enough images to fill in the fake image pool, we kept adding the fake images generated by generator each time when we train the generator. When the fake image pool is full, there is fifty percent chance that we apply that fake image to train discriminator directly, and there is fifty percent that we randomly replace the one image in fake pool with the newly generated fake image and apply the old replaced fake one to train the discriminator.

Figure 2. Horse2Zebra Result on 100 epoch

4.3. Result

Since human face is a very delicate structure, we first tested our model on the less delicate images. We first evaluated our proposed method on horse2zebra dataset from ImageNet[5] After training with 100 epochs on 256 x 256 images, we got the following results (see Figure 2). The first column is input images of real zebras and the second column is corresponding horse images generated by F, the third column is input images of real horses and the fourth column is corresponding zebra images generated by G. As we can see from the figure, the transformation from the horse to the zebra is slightly better than the transformation from the zebra to the horse.

We then apply the cycle-GAN-based model on Labelled Faces in the Wild dataset [10]. In Figure 3, the first column is input images of real aged people and the second column is corresponding output images of younger version of them generated by F, the third column is input images of real youth and the fourth column is corresponding output images of them when they were aged generated by G. The figure implies the transformation from youth to the aged is better than the transformation from the aged to youth.

Figure 3. LFW Result on 50 epoch

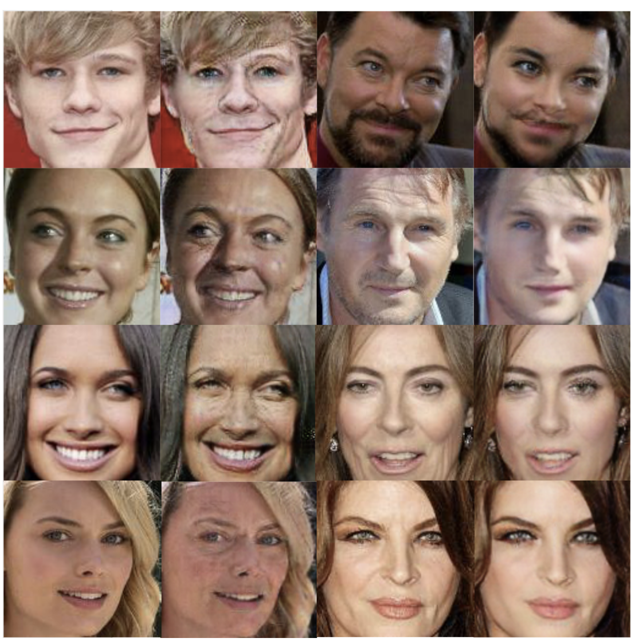

Figure 4. CACD Result on 100 epoch

After that, we apply the model to another facial dataset called Cross-aged Celebrity Dataset [2, 3]. In Figure 4, the first column is input of the real images of young people, and the second column is the images of corresponding generated fake old people. The third column is input of the real images of old people, and the fourth column is the images of the corresponding generated fake young people. We can see from Figure 4 that the model learns how to add or remove wrinkles on faces. It also learns to transform eyes so that people can look younger.

5. Conclusion

In our project, we propose a new cycle-GAN-based method for Age Progression Problem. As a by-product, the new method we propose can also address Age Regression Problem. This is the first time cycle-GAN is introduced to this classic problem.

In the future work, we will try to introduce a face recognition hash function $H(a, b)$ to measure the similarity of facial features after age progression and age regression. We are also thinking about introducing facial recognition based neural network structures to discriminators.

References

[1] G. Antipov, M. Baccouche, and J.-L. Dugelay. Face aging with conditional generative adversarial networks. In *Image Processing (ICIP), 2017 IEEE International Conference on*, pages 2089–2093. IEEE, 2017.

[2] 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.

[3] B.-C. Chen, C.-S. Chen, and W. H. Hsu. Face recognition and retrieval using cross-age reference coding with cross-age celebrity dataset. *IEEE Transactions on Multimedia*, 17(6):804–815, 2015.

[4] L. Chen, X. Hu, and Z. Zhang. Face aging with boundary equilibrium conditional autoencoder. *IEEE Access*, 6:54834–54843, 2018.

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[6] Y. Fu, G. Guo, and T. S. Huang. Age synthesis and estimation via faces: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(11):1955–1976, Nov 2010.

[7] Y. Fu and N. Zheng. M-face: An appearance-based photorealistic model for multiple facial attributes rendering. *IEEE Transactions on Circuits and Systems for Video technology*, 16(7):830–842, 2006.

[8] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks. *Advances in Neural Information Processing Systems*, 3:2672–2680, 2014.

[9] I. Kemelmacher-Shlizerman, S. Suwajanakorn, and S. M. Seitz. Illumination-aware age progression. In *Proceedings*

of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3334–3341, 2014.

[10] E. Learned-Miller, G. B. Huang, A. RoyChowdhury, H. Li, and G. Hua. Labeled faces in the wild: A survey. In *Advances in face detection and facial image analysis*, pages 189–248. Springer, 2016.

[11] M.-Y. Liu and O. Tuzel. Coupled generative adversarial networks. In *Advances in neural information processing systems*, pages 469–477, 2016.

[12] Y. Liu, Q. Li, and Z. Sun. Attribute enhanced face aging with wavelet-based generative adversarial networks. *CoRR*, abs/1809.06647, 2018.

[13] Z. Liu, Z. Zhang, and Y. Shan. Image-based surface detail transfer. *IEEE Computer Graphics and Applications*, 24(3):30–35, 2004.

[14] M. Mirza and S. Osindero. Conditional generative adversarial nets. *Computer Science*, pages 2672–2680, 2014.

[15] G. Perarnau, V. D. W. Joost, B. Raducanu, and J. M. lvarez. Invertible conditional gans for image editing. 2016.

[16] J. Suo, S.-C. Zhu, S. Shan, and X. Chen. A compositional and dynamic model for face aging. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(3):385–401, 2010.

[17] Y. Tazoe, H. Gohara, A. Maejima, and S. Morishima. Facial aging simulator considering geometry and patch-tiled texture. In *ACM SIGGRAPH 2012 Posters*, page 90. ACM, 2012.

[18] W. Wang, Z. Cui, Y. Yan, J. Feng, S. Yan, X. Shu, and N. Sebe. Recurrent face aging. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2378–2386, 2016.

[19] Z. Wang, X. Tang, W. Luo, and S. Gao. Face aging with identity-preserved conditional generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7939–7947, 2018.

[20] Wikipedia contributors. Generative adversarial network — Wikipedia, the free encyclopedia, 2018. [Online; accessed 15-December-2018].

[21] H. Yang, D. Huang, Y. Wang, and A. K. Jain. Learning face age progression: A pyramid architecture of gans. *CoRR*, abs/1711.10352, 2017.

[22] Z. Zhang, Y. Song, and H. Qi. Age progression/regression by conditional adversarial autoencoder. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4352–4360, 2017.

[23] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017.