Face Aging with Cycle-GAN

Maria Rosa Scoleri
Politecnico di Torino
s301841@studenti.polito.it

***Abstract*—Age progression refers to the process of artificially depicting how a person's appearance may change over time. This project aims to present an approach to age progression in facial images through the implementation of a CycleGAN trained using two custom datasets. The collection of images is created through a combination of a scraping script followed by a manual selection and handpicked images from the public UTKFace dataset. Every image is then passed through a slightly modified version of OpenCV's Face Detection Neural Network to create portrait images. The resulting pictures are then subject to various pre-processing operations. Two main models are used: a first attempt is performed employing the U-Net generator and a different version is implemented using a custom ResNet-based generator. Finally, we perform an experiment using transfer learning: we apply fine-tuning on the pre-trained Cycle-GAN model horse2zebra. We qualitatively evaluate our results by visualizing the images and comparing the different results of our models. Ultimately, we perform a quantitative evaluation using an Age Detection Neural Network to estimate the age difference between the original and generated faces.**

I. Introduction

Age progression consists of artificially attempting to predict how a person's appearance will change over time as they grow older. The interest in this process is increasing due to its applications in various fields, including entertainment, forensics, and healthcare. The most immediate strategy to generate aged faces is to train a generic GAN using paired images. The biggest challenge to overcome in order to achieve this goal would be the collection of a suitable dataset. Paired training data, in this context, would require images of the same individual in both young and old age, and finding them would be very arduous if not impossible. For this reason, we need a network that can perform an unpaired image-to-image translation. The most effective starting point to achieve the transformation from the "young domain" to the "old domain" (and vice versa) with unpaired data is the CycleGAN framework [1]. This network is known for its unique ability to map features between different domains without the need for paired training data. Several works based on the Cycle-GAN network that deal with this task have been published and use different approaches: the RAGAN [2] framework, for example, can learn the personalized age features by using high-order interactions between given identity and target age, the Triple-GAN [3] network adopts triple translation loss to model the strong interrelationship of age patterns among different age groups, and the same goal is pursued with the PFA-GAN [4].

Our first approach consists of using a CycleGAN with U-net generators[1]. In particular, we use the Pix2Pix [5] generators and discriminators imported from TensorFlow's *example* module. The generator consists of a modified version of the U-Net network originally introduced in [6]. The second method we implement consists of replicating the original CycleGAN paper where the generator is based on the ResNet [7] architecture. For the purpose of this project, we developed a ResNet generator that only uses 6 residual blocks, due to the limited computing capabilities of the available resources. As a final experiment, we load the pre-trained cycle-GAN model horse2zebra and fine-tune it for our task. To evaluate our results, we visually inspect the output images, examining the signs of aging and checking that the images are realistic and consistent. Finally, we find the age difference between the original and generated faces using an age detection neural network.

II. Dataset

In this section, we will explore the strategies adopted to create the datasets for training and the pre-processing operations applied to each image.

A. Images collection

The training process for this project required the creation of two distinct datasets: *dataset_young* and *dataset_old*, each representing one of the domains needed for the CycleGAN. The images for these datasets were collected in equal part from a web scraping operation and sampling the UTKFace [8] dataset. The web scraping process was executed on Bing Images and it involved the use of several keywords. Some examples are shown in Table I.

Regardless of the source of the images, a manual selection was performed to ensure the quality of the pictures and their suitability to the problem. Full-body pictures, images that were extremely blurred, had excessive clutter, or depicted too many

[1]Our code relative to this task is strongly inspired by the TensorFlow tutorial, which is a TensorFlow adaptation of the CycleGAN code [1]. In this adaptation, the original ResNet generator is replaced with a simpler U-net network.

dataset_old	**dataset_young**
elderly portrait photography	women photography id
old actors	man photography id
old actresses	female portrait photography
elderly photos	male portrait photography
elderly passport photo	close up face photography

TABLE I: Some keywords used in the scraping process to create dataset_old and dataset_young.

(a) Sample images from dataset_young

(b) Sample images from dataset_old

Fig. 1: Examples images from the collected datasets

people were removed. Particular attention was given to the images collected through web scraping, as some keywords captured non-related images. Furthermore, the selection took into consideration the diversity of the people depicted: we focused on including people of different genders, ethnicities, and facial features. The resulting *dataset_young* and *dataset_old* contain 2000 pictures each. Despite the total number of images may seem modest, it is important to notice that this size is a deliberate choice, considering the computational limitations of Google Colab. Some examples of the collected images are shown in Figure 1.

B. Face extraction

Following the initial manual curation, the datasets still required some additional modifications. Each image in the dataset maintained its shape and specific characteristics concerning facial dimensions, poses, and image compositions. The faces of the people depicted could be in the upper or lower part of the image, corners, or centers. The scale of the faces could vary from close-ups to moderately distant shots. Finally, the pictures could be vertical, square, or horizontal. Given these differences, a simple resizing of the images would distort the depicted faces and would not make the dataset more suitable for training. To address this issue, we applied a specialized face extraction procedure using OpenCV's [9] Face Detection Neural Network [2]. This work allowed us to obtain images that are all similar to each other: square, with the face at the center, and without many distracting elements that were present in the original images. The original model is very strict in extracting the face: it focuses strictly above the eyes and below the mouth, often excluding hair, chin, and in some cases ears. For this reason, we implement a slightly modified version of the original code. In our work, the resulting images contain the face and also incorporate a larger context around the facial area. The choice to expand the selected area is driven by the will to find a compromise between avoiding unnecessary and distracting elements from the images while also maintaining all the essential facial features. Some examples that display the application of the face extraction tool are shown in Figure 2.

[2]We used the code provided in this GitHub folder

C. Preprocessing

After the extraction of faces from the original datasets, we split them into train and test, consisting respectively of 90% and 10% of the images. In order to make the pictures more suitable for training, several pre-processing operations are performed:

- **Resizing**: each image is resized to dimensions 256x256, ensuring uniformity in the input size for the model.
- **Random Mirroring**: it is applied to perform some data augmentation and improve model generalization, introducing some variation in facial orientation.
- **Standardization**: all the images are normalized so that each pixel value is in the $[-1, 1]$ interval.

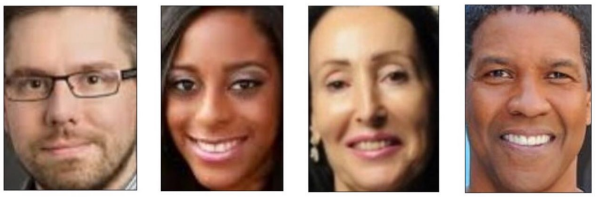
After the initial collection, face extraction, and preprocessing techniques, we obtain our final *dataset_young* and *dataset_old*.

III. Method

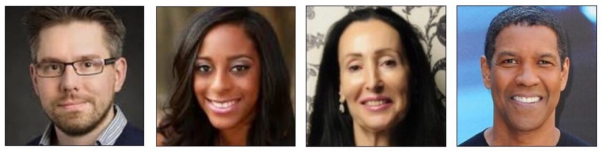
A. Cycle-GAN structure and losses

The goal of the project is to create a model able to predict how the appearance of a young person will change over time. To accomplish this task we implemented a solution based on the CycleGAN architecture. This network has a very peculiar

(a) Input images



(b) Output images after original face detection



(c) Output images after our modified face detection

Fig. 2: The Figure shows some sampled images from dataset_old and dataset_young (a), the corresponding output using the original OpenCV's face detection algorithm (b) compared with our slightly modified version (c).

structure that consists of two generators (G and F) and two discriminators (D_X and D_Y). Generator G transforms images from domain X to domain Y, while generator F does the opposite. We denote as y an image belonging to domain Y and its data distribution as $y \sim p_{\text{data}}(y)$. Similarly, x is an image from the X domain, with data distribution denoted as $.x \sim p_{\text{data}}(x)$. The two discriminators evaluate the realism of the images in their respective domains. The structure of the network is shown and further explained in Figure 3.

To establish a correct learning process for the model, two different losses are used:

- **Adversarial Loss**: the purpose of this loss is to ensure that the generators learn to produce images in the target domain that are realistic enough to deceive the discriminators. The resulting images should be indistinguishable from images that actually belong to the target domain. The objective is defined as:

$$\begin{aligned}\mathcal{L}_{\text{GAN}}(G, D_Y, X, Y) =& \mathbb{E}_{y \sim p_{\text{data}}(y)}[\log D_Y(y)] \\ &+ \mathbb{E}_{x \sim p_{\text{data}}(x)}[\log(1 - D_Y(G(x)))]\end{aligned}$$

  where G tries to generate images $G(x)$ that look similar to images from domain Y, while the discriminator D_Y learns to distinguish between real samples y and generated samples $G(x)$. G tries to minimize this objective against D_Y that aims to maximize it: $\min_G \max_{D_Y} \mathcal{L}_{\text{GAN}}(G, D_Y, X, Y)$. A loss symmetrical to this one is defined to train the generator F and the discriminator D_X.
- **Cycle Consistency Loss**: it is used to ensure that the reconstructed images after a "round-trip" through both generators remain close to the original images. Adversarial training can learn mappings that produce outputs with the same distribution as the images in the target domain. However, with adversarial loss only, the network could learn to create an image in the target domain that loses its similarity with the input image. For example, considering the young-old domains, the network could learn to generate an image of an older person that has lost the features of the young face given as input. The cycle consistency loss helps avoid this issue by forcing the network to produce output images in the target domain that are still coherent with the input. The objective function is defined as follows:

$$\begin{aligned}\mathcal{L}_{\text{cyc}}(G, F) =& \mathbb{E}_{x \sim p_{\text{data}}(x)}\left[\|F(G(x)) - x\|_1\right] \\ &+ \mathbb{E}_{y \sim p_{\text{data}}(y)}\left[\|G(F(y)) - y\|_1\right]\end{aligned}$$

The full objective function is:

$$\begin{aligned}\mathcal{L}_{\text{GAN}}(G, F, D_X, D_Y) =& \mathcal{L}_{\text{GAN}}(G, D_Y, X, Y) \\ &+ \mathcal{L}_{\text{GAN}}(F, D_X, Y, X) \\ &+ \lambda L_{\text{cyc}}(G, F)\end{aligned}$$

B. Transfer Learning

Transfer learning is a technique in machine learning that involves leveraging knowledge gained from training a model on one task and applying it to a different but related task. We use the pre-trained Cycle-GAN model horse2zebra, created to translate images from the *horse* domain to the *zebra* domain and vice-versa, and fine-tune it on our datasets. The original work on the CycleGAN takes into consideration several origin and target domains: photo $\leftrightarrow$ Cezanne, winter $\leftrightarrow$ summer, aerial photos $\leftrightarrow$ Google maps, Monet $\leftrightarrow$ photos, and many others. All these translations deal with domain switches where all the features of the image need to be translated into the target domain. For this reason, we chose to fine-tune horse2zebra: this model deals with creating or removing a specific pattern. Even though our task is more difficult than replicating a simple pattern, we believe that horse2zebra constitutes the best possibility among the aforementioned models.

To implement our transfer learning approach, we maintain the pix2pix discriminators we trained for the Cycle-GAN with the ResNet implementation. The horse2zebra pre-trained generators have all the layers frozen until the eight residual block, leaving for training only the ResNet part going from the last residual block forward.

C. Evaluation strategies: Visual and Quantitative Analysis

The primary method of evaluating the effectiveness of the proposed aging CycleGAN model is through visual inspection. This qualitative approach aims to assess the realism and coherence of the generated images depicting aged faces. The resulting faces can be examined to check if the aging

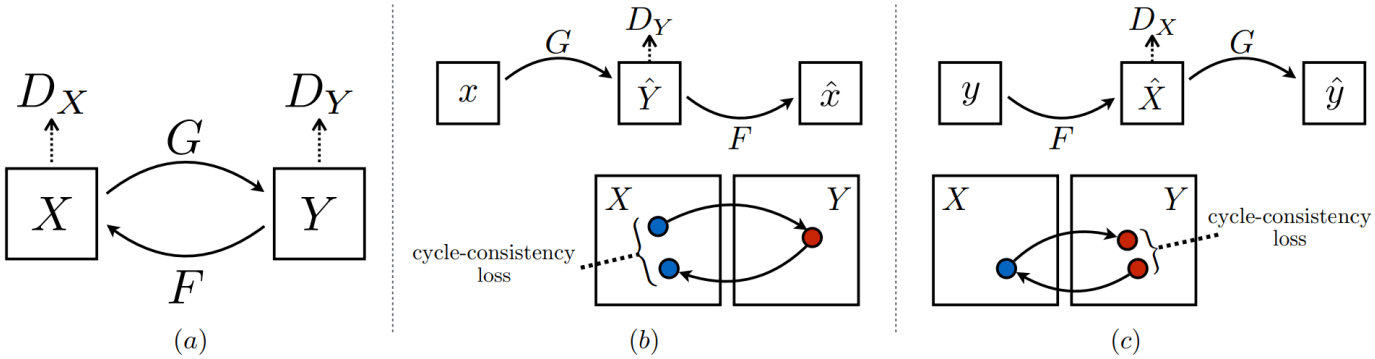


Fig. 3: (a) The model contains two mapping functions G : X → Y and F : Y → X, and associated adversarial discriminators D_Y and D_X. D_Y encourages G to translate X into outputs indistinguishable from domain Y, and vice versa for D_X and F. To ensure the coherence of the produced images, two consistency losses are applied: (b) forward cycle-consistency loss: x → G(x) → F(G(x)) ≈ x, and (c) backward cycle-consistency loss: y → F(y) → G(F(y)) ≈ y. The logic behind this use of the consistency loss is that if switch from domain X to domain Y with the generator G, and then we switch back with F, we should obtain an image very similar to the one we started with.

effects are realistic, considering for example textured skin, differences in hair color, and the addition of wrinkles. Another important assessment can be made by considering whether the resulting faces maintained the original features. We will see that the considered models perform slightly differently in these regards.

While a visual analysis to assess the realism of the images is probably the most straightforward evaluation measure for this type of network, we also implement a quantitative approach. Throughout the years, several age detection approaches have been developed, like those examined in [10] or those compared in [11]. For the purpose of our project, we choose the Age-Gender-Detection Network that can be found in the Deepface library[3] developed by OpenAI. Given an image x and its transformation $G(x)$, we estimate the respective ages a_x and $a_{G(x)}$ and then compute the age difference $d_x = a_{G(x)} - a_x$. Finally, we evaluate the percentage of images p_{d_x} in which $d_x > 0$, $d_x > 5$, and $d_x > 10$, i.e. the percentage of images on which our method was effective with different degrees of success.

IV. Experiments

A. Implementation details

Three models were tested: Cycle-GAN with U-Net generator, Cycle-GAN with ResNet generator and the horse2zebra pre-trained and fine-tuned model.

Concerning the first approach, the generators and discriminators are those found in pix2pix, which is a conditional generative adversarial network (cGAN) [5]. The generator is a U-Net network and the discriminator is a simple classifier in which each block is made of a convolution layer, a normalization layer, and a LeakyReLU. The complete cycle-GAN model with the U-Net generator was trained for 50 epochs.

The second model revolves around the use of a ResNet generator. The original Cycle-GAN paper employs ResNet networks with 9 residual blocks, which have around 11.4 million parameters each. Considering that the final model is comprised of two generators and two discriminators, we were required to apply some changes to the original implementation to adapt to our resources. To speed up the training process and work with the available computational power, we switched to a ResNet network with 6 residual blocks, significantly reducing the number of parameters. The discriminators for this second model have the same structure as those in the first one. Similarly to the previous experiment, a 50 epochs training was performed.

Concerning the transfer learning task, we utilize the pre-trained model horse2zebra[4] and fine-tune it on our datasets. From the horse2zebra pre-trained model we were able to extract the two cycle-GAN generators, based on ResNet's architecture, which contain 9 residual blocks each. Our fine-tuning was performed by freezing every layer before the last residual block. The discriminators used for this approach are the same pix2pix discriminators that we already trained for the previous task mentioned. This final model was fine-tuned for 30 epochs.

The chosen optimizer is always *Adam* with a learning rate set to $2 \cdot 10^{-4}$, as suggested in the Cycle-GAN paper, and a beta_1 parameter equal to 0.5.

B. Visual evaluation

In this section, we discuss the results obtained with the different methods employed. Figure 4 shows the results obtained after training the CycleGAN model with the two given configurations and after transfer learning. We can see that all three models have transformed the images, and in most cases, the people depicted look older. However, the configurations still have some issues and the generated images do not always look realistic.

[3]Here is the official website of the Deepface library and this is the library's GitHub repository.

[4]the script to download the pre-trained model can be found in this GItHub folder, containing the PyTorch code for the original cycle-GAN implementation.

(a) Input images

(b) Output with CycleGAN using U-Net generator.

(c) Output with CycleGAN using ResNet generator.

(d) Output with CycleGAN after fine-tuning horse2zebra.

Fig. 4: This comparative picture, starting from a given input (a) shows some results obtained with the CycleGAN using the U-Net generator (b), those with the ResNet generator (c), and those obtained after fine-tuning horse2zebra (d). We can see that the first model tends to add unwanted texture to the skin and whiten the images. The second model provides more realistic pictures but the aging effect is very light. The third model provides results that are quite realistic and the people look aged, even if not always in a homogeneous way.

Let us consider the CycleGAN with the U-Net model first: the images appear to be brightened and whitened with respect to the input, the faces are more textured, the hair is in most cases lighter or gray, and the faces on average do look older. Nonetheless, some critical concerns arise, as the model does not always produce images that appear realistic. This model introduces unintended elements such as discolored spots, light stripes, and white and black dots. Moreover, certain images manifest peculiar repeating patterns that were not present in the input images.

When evaluating the second model with the ResNet generator, instead, we can appreciate a visible improvement in realism: it avoids introducing extraneous patterns, maintaining a higher level of fidelity to the original pictures. The images are still brightened and the faces are lighter in complexion, but the appearance is more natural when compared with the output of the first model. The skin appears with more fine lines and texture with respect to the input, and the hair is in most cases gray or white, even though it is not homogeneous. While the overall visual quality of the images is notably enhanced, some limitations persist. Certain individuals appear to age in a coherent and well-defined way, while others undergo a less perceptible aging process, and the skin is on average less textured than expected.

Model	$p_{d_x>0}$	$p_{d_x>5}$	$p_{d_x>10}$
U-Net	60.5	20.5	7.5
ResNet	56.5	6.0	1.5
horse2zebra	70.0	17.0	2.5

TABLE II: This table shows the percentage of people that the Deepface model considers aged with respect to the input. The $p_{d_x>0}$ column contains the percentage of images that show any kind of aging, the next column shows the percentage of images that appear to be aged more than 5 years, and the last column shows the percentage of those aged more than 10 years.

The results after the transfer learning approach are somewhere in the middle between the two presented before. The faces look particularly aged and textured, while still maintaining a natural appearance and the original face features. This model still presents some challenges: the light can reflect in unexpected ways, the skin tones appear more gray, some added texture is not as realistic as anticipated, and not every image is aged to the same degree.

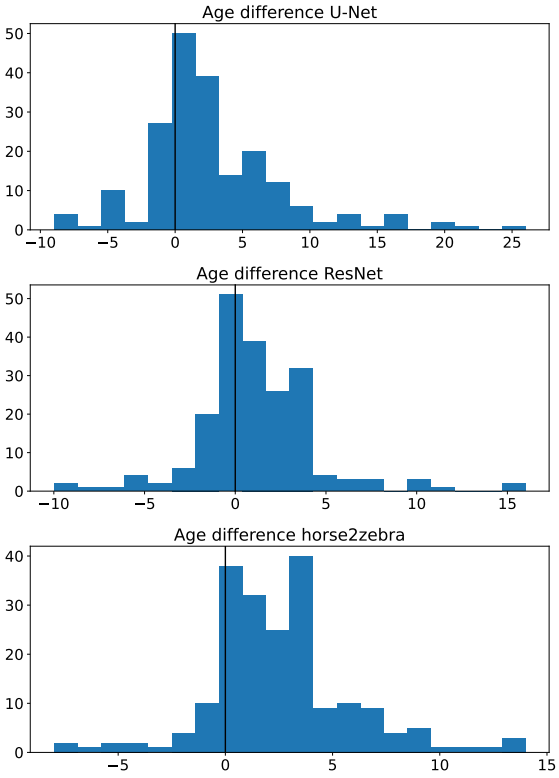


Fig. 5: The figure shows the distribution of the age difference d_x of the faces before and after the images go through the models.

C. Quantitative evaluation

We ran the Deepface Age-Gender-Detection Network on our images before and after passing them through the network in order to evaluate our aging algorithms. The main results are shown in Tab II, where we can see the percentage of images p_{d_x} that are considered aged. Relying on the results given by the Age-Gender-Detection Network, the model pre-trained on horse2zebra is the one that ages the larger number of images, as its $p_{d_x>0}$ is the most elevated. The cycle-GAN model based on the U-Net generator appears to be the one to apply the most drastic aging effect, as $p_{d_x>5}$ and $p_{d_x>10}$ are the highest among all three models. The cycle-GAN based on ResNet produces the most subtle transformation, even if the resulting images appear to be the most realistic. Figure 5 shows the distribution of the age difference of the faces before and after going through our models, giving us an estimate of how much the faces are aged. These plots show that most people result aged between 1 and 5 years and that some faces are even identified as younger by the Age-Gender-Detection Network.

V. Conclusion

Our experimentation with three distinct models for face age progression using Cycle-GAN has provided valuable insights into the capabilities of each approach. The first model, employing the U-Net generator, introduces the most pronounced aging effects. However, the generated images exhibit disturbances and peculiar patterns, compromising their overall realism. The second model, incorporating a custom ResNet-based generator, produces the most realistic facial transformations. Still, the aging effects are more subtle than anticipated. The third model, utilizing generators pre-trained on horse2zebra, manages to reach a trade-off between realism and aging effects. Despite the training for this model being less prolonged, the images look more realistic than the model based on U-Net and more transformed than those produced by the ResNet-based model. Notably, the results are still not realistic enough and there is potential for improvement with further training.

For future works, with the availability of additional computational resources, there is significant potential to enhance the performance of our models. Experimenting with different learning rates, optimizers, and normalization layers, and extending the training duration would definitely elevate the results. Similarly, a more extensive dataset would not only diversify the aging patterns captured by the model but also contribute to its generalization capabilities. Moreover, it would be fascinating to further explore the possibilities provided by transfer learning. Testing the other domain transfers associated with the Cycle-GAN work, beyond horse2zebra, may provide an interesting starting point for fine-tuning our models.

Incorporating all these enhancements could achieve better realism and improved adaptability to diverse facial features and aging patterns.

References

[1] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks, 2020.

[2] Farkhod Makhmudkhujaev, Sungeun Hong, and In Kyu Park. Re-aging gan: Toward personalized face age transformation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3908–3917, October 2021.

[3] Han Fang, Weihong Deng, Yaoyao Zhong, and Jiani Hu. Triple-gan: Progressive face aging with triple translation loss. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020.

[4] Zhizhong Huang, Shouzhen Chen, Junping Zhang, and Hongming Shan. Pfa-gan: Progressive face aging with generative adversarial network. *IEEE Transactions on Information Forensics and Security*, 16:2031–2045, 2021.

[5] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks, 2018.

[6] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[8] Zhifei Zhang, Yang Song, and Hairong Qi. Age progression/regression by conditional adversarial autoencoder. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.

[9] Open source computer vision library. https://opencv.org/.

[10] Marwa Badr, Amany Sarhan, and Reda Elbasiony. Facial age estimation using deep neural networks: A survey. pages 183–191, 12 2019.

[11] Alice Othmani, Abdul Rahman Taleb, Hazem Abdelkawy, and Abdenour Hadid. Age estimation from faces using deep learning: A comparative analysis. *Computer Vision and Image Understanding*, 196:102961, 2020.