

2D-FACE ALIGNMENT WITH CYCLEGAN FACE AGING IMAGE-TO-IMAGE TRANSLATION

Nabilah Hamzah^a, Fadhlan Hafizhelmi Kamaru Zaman^{a*}, Nooritawati Md Tahir^{ab}

^aCollege of Engineering, Universiti Teknologi Mara (UiTM), Shah Alam, Malaysia

^bInstitute for Big Data Analytics and Artificial Intelligence (IBDAAI), Universiti Teknologi Mara (UiTM), Malaysia.

Article history

Received

17 August 2021

Received in revised form

31 March 2022

Accepted

02 April 2022

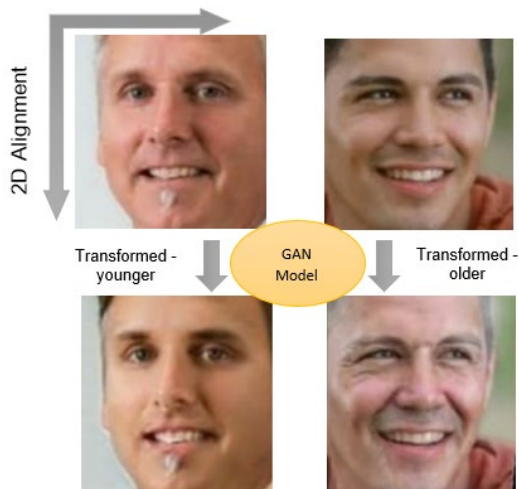
Published online

30 November 2022

*Corresponding author

fadhlan@uitm.edu.my

Graphical abstract



Abstract

Face alignment is one of the pre-processing processes where the face plays a crucial part in image tasks and computer vision. As part of the pre-processing step, it is the first step taken before implementing an image processing task. By aligning face, it is expected to improve the network model performance, because good input data is now represented in the network model. This research aims to see whether pre-processing the input data can improve the network model performance. A 2D-face alignment technique is used to align all the input images. All the input image that is already being aligned is used as the input image for the CycleGAN face aging image-to-image translation model. In this work, the CycleGAN network model is used to translate an image of a young face to their older version and vice versa. The result obtained shows that if the network model is presented with a properly aligned face, it can translate the image into a younger or older version better than when presented with a non-aligned face.

Keywords: Face Alignment, image-to-image translation, CycleGAN, Deep Learning, AI

© 2022 Penerbit UTM Press. All rights reserved

1.0 INTRODUCTION

Many image tasks and computer vision such as image stylization, segmentation and abstraction can be classified as an image-to-image translation problem [1], [2]. The image-to-image translation is a process of conversion of one image to another style, version or new image depending on the features being extracted from the sample data or training dataset. The image-to-image translation is a difficult task until deep learning network is introduced. The most noticeable deep learning network used in image-to-image translation is Generative Adversarial Network (GAN) where GAN used a generator and discriminator model to create a new style image by learning the image feature from the sample image [3]. The basic idea of training GAN is to simultaneously train the generator and discriminator model [4], where the discriminator model goal is to distinguish the real and generated samples [3], [4]; while the generator model will generate an output sample from the

training dataset to dupe the discriminator model. There are many extended GAN research projects such as CycleGAN [3], [5], DualGAN [2], [6], StarGAN [7] and many more [8]–[10]. All extended GAN research focuses on mitigating GAN flaws by changing the framework or pipeline to improve GAN performance. Each extended GAN technique focuses on mitigating different flaws. State-of-the-art extended GAN model shows a magnificent result in computer vision tasks especially image-to-image translation so instead of focusing to improve the GAN model by changing a new pipeline or framework, this research is focusing on the image pre-processing process where this also one of the crucial steps to enhance GAN model. In this research paper, an extended GAN calls CycleGAN is chosen because CycleGAN is one of the extended GAN models that are popular and shows an impressive result in translating images. Before the CycleGAN is trained, all the sample data or training dataset will undergo an image pre-processing process.

Image pre-processing is a process of preparing the dataset before the image is used as an input for the network model. Alignment, crop, and rotation are examples of the image pre-processing process. As this research involves face image, the face alignment technique is used to detect facial landmarks i.e., eyes, mouth, and nose after that align all the sample data to get a standardise face image [11]

2.0 RELATED WORKS

This section elaborates and discusses any related works before. This section is divided into two parts: Face Alignment and Generative Adversarial Network (GAN). The face alignment part discussed the literature review from the classic to the current face alignment technique. The Generative Adversarial Network (GAN) part discussed all the flaws and disadvantages of the GAN network and how to mitigate each of the flaws by introducing a new pipeline or framework based on the GAN approach.

Face Alignment

The face alignment technique can be further divided into two categories: regression-based face alignment and analysis-by-synthesis approach [12]. The analysis-by-synthesis face alignment approach for example ASM [13], [14] and AAM [15], [16] approximate facial landmarks using a global shape model with a generative texture model [17]. The CLM is the extended method from AAM, where this model performs much faster than AAM [18], [19]. CLM assumes that all face lies in linear subspace and determine the dimension using Principal Component Analysis (PCA).

A classic regression-based approach was proposed to directly estimate the position of facial landmarks i.e. eyes, mouth and nose using a cascade regression [20]. This regression-based approach cannot approximate the location of the facial landmarks if the image is under a different view angle or have an occlusion. Due to that reason, the performance of the regression method is decreasing if the image is not frontal [21] and also this approach suffers from a corrupted image [20]. As for now, there is plenty of research that improves the regression-based model approach to increase model performance. Not like other regression-based models, Cou *et al.* (2014) suggest an explicit shape regression where this approach uses a whole face as a set of facial landmarks and by doing that it can minimise the alignment errors [22]. Another regression-based model approach is by using coarse-to-fine shape searching, where this approach performs better on large pose variation and initialisation independent [23]. Yang *et al.* (2015) suggest a regression-based model using power estimation where this model is applied to the facial image that has an occlusion in it [24]. This model shows an impressive result on the challenging dataset i.e. COFW and 300-W where the estimate accuracy is at 72,4% in estimated face and non-face objects. Adaptive cascade regression is another example of a regression-based model approach. Liu *et al.* (2016) suggest an adaptive cascade regression where the shape index is introduced to approximate the occlusion on each landmark (eye, nose, and mouth) and weight the landmark according to the occlusion level [20]. By doing this it can reduce the noise on

the shape indexed feature to robust the face alignment process.

Nowadays, deep learning has become one of the hottest items and key to success in the computer vision field. A large and growing body of literature has investigated deep learning frameworks to solve many complex tasks in the computer vision field including face alignment. Several attempts have been made to build a more robust face alignment model based on a deep learning approach. Liu *et al.* (2016) suggest a dual sparse constrained cascade regression model where this model uses a deep learning approach framework called Convolutional Neural Network (CNN) to detect the facial landmarks [25]. Deep Alignment Network or DAN is another approach used deep learning framework where DAN consists of many stages and each of the stages improves the location of the facial landmark estimated by the previous stage [26]. Wu and Yang (2017) suggest using an inter and intra dataset variation with a Deep Variation Leveraging Network (DVLN) that consists of two sub-network [27]. Recently, another study with deep learning is published using Reasoning Decision Network (RDN) where the researcher suggests improving the conventional coarse-to-fine shape approach by using RDN. The RDN is used to decide to remove an outlier for more robust initialisation [28].

Face alignment remains a challenging topic in computer vision tasks. As there are many models and technique in approaching too robust the face alignment problem and for now the state-of-the-art methods for face alignment is at a very good state. But there is still a lot of room for improvement to get a better face alignment technique to solve many variations pose and face pose.

Generative Adversarial Network (GAN)

GAN's initial work was started by Goodfellow *et al.* (2014) where they proposed a new generative model via adversarial and train two models simultaneously [4]. GAN has changed the perspective of image-to-image translation where GAN makes it a lot easier to translate one image style to another style. GAN has two different models: (1) generator model and (2) discriminator model where these two models are the CNN model that competes to complete their task [29]. The generator model task is to generate images by learning the features from the sample data. The discriminator model task is to inspect whether the generated images are real or synthetic [3]. Not only image-to-image translation but GAN has been used in numerous applications such as generating example form image datasets, generating realistic photography, image-to-image translation [1], [2], [5], [29], [30], text-to-image translation [31], [32], face aging [6], [33]–[35], medical imaging [36], [37] and super-resolution [9], [38]. GAN has proved that image tasks and computer vision can be a lot easier and faster.

GAN's biggest advantages are it allows training a large number of samples, fast to simulate and produce visually compelling sample images [39]. However, all these advantages came with vulnerability. Even though GAN allows to train a large number of samples, it became unstable to train which can lead to mode-collapse [39], [40]. GAN also have no inference capabilities for most application [8]. GAN itself has many flaws and drawbacks that make other researchers venture GAN with another technique or change the pipeline to enhance GAN performance.

Radford *et al.* (2015) try to improvise GAN by using the generator and discriminator model as a feature extractor for supervised tasks [40]. Doing this makes the GAN model more stable. Another technique is called Auto-encoding GAN (AE-GAN) where this technique is a combination technique between the auto-encoder and GAN model [39]. This model discriminates the real and the generated image by using a loss given by the auto-encoder. GAN is also popular in face aging image translation, where Song *et al.* (2017) use Dual Conditional GAN to predict the face of a person at a different age level [6]. In the medical imaging field, the researcher implements GAN for image-to-image translation on the multi-contras MR images [29], [36], [37]. It shows that GAN is multiplayer in the computer vision field where it can do almost everything in translating images including a medical image. Another interesting project that caught our attention is the researcher using GAN to predict the geometric surface in injection molding [41]. Zhang *et al.* (2017) put GAN into another level up, whereby only using a text or word to describe a thing and GAN will produce an image output from the text input [31]. GAN also ventures with natural language processing (NLP) where GAN is used to process text to extract crucial information [42], [43].

Even though GAN itself has many flaws and disadvantages but when GAN is improved it can be a very versatile model network as there are many GAN that can do such as medical imaging, face aging, surface prediction, text-to-image translation, text classification and many more. It shows that GAN based approach can archive much more in computer vision tasks in the future.

3.0 PROPOSED FRAMEWORK

The proposed framework for this research is by using an extended GAN model called CycleGAN. This section is divided into three; face detection, face alignment and CycleGAN. Figure 1 shows the general methodology for this research project, where the first step is face detection and then face alignment and the last step in image-to-image translation using the CycleGAN model.

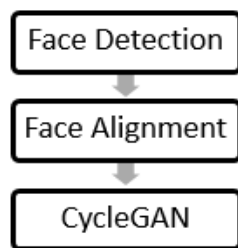


Figure 1 General Methodology

Face detection and face alignment are combined processes where these two processes are processed including in the image pre-processing process. The flow diagram for face detection, face alignment and CycleGAN process is show in Figure 2.

Face Detection

Face detection is the process of detecting a human facial form in the input image. In this research study, a Haar Cascade is used as a detector to detect a human facial form. To implement Haar Cascade a Haar feature is used to extract crucial features from the input image where the Haar feature is in the shape of a rectangle designed for rapid face detection [44], [45]. Equation 1 shows how to calculate the Haar feature whereby subtracting the sum of the pixel under a white rectangle from the sum of the pixel under the black rectangle (refer to Figure 3).

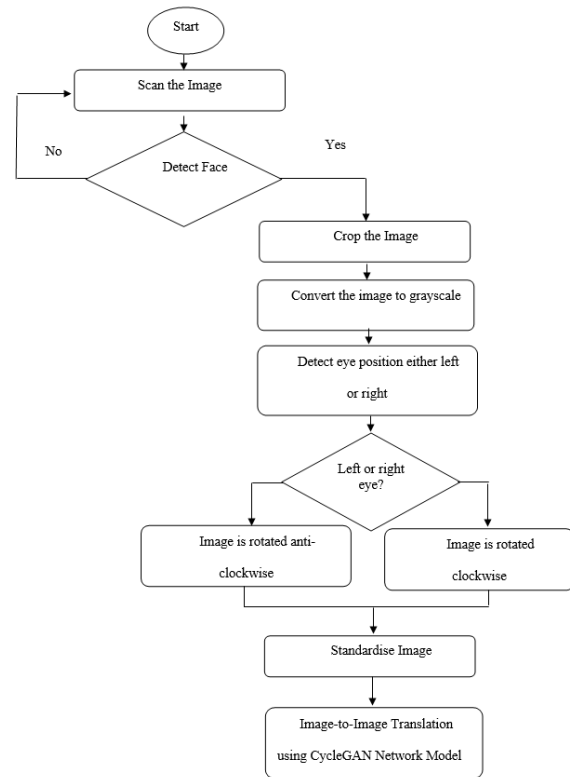


Figure 2 Flow Diagram Process

$$p(x) = Sum_{Black\ Rectangle} - Sum_{White\ Rectangle} \tag{1}$$

Where $p(x)$ is the Haar feature. The Haar feature will start looking over the image for human facial form from the top left corner to the bottom right corner [44]. The input image will be scanned many times to detect a human facial form on the image. Luckily, now OpenCV offers Haar Cascade, and it can be easily used and obtained it also offers nose and eye detection. Figure 3 shows some of the Haar feature examples.

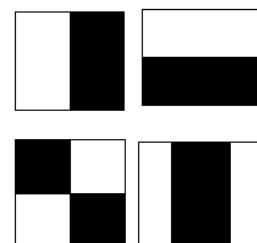


Figure 3 Haar Cascade Feature Examples

Face Alignment

Face alignment in this work aims to increase the performance of the CycleGAN model by providing a normalized input. This face alignment is a 2D face alignment approach using OpenCV and Python. It is much easier, lightweight and less complex compared to others face alignment techniques. As a detector, Haar Cascade will detect a face and eyes on the input image and give the coordinate of the face and eyes position. Firstly, is to detect the facial form and then crop the image. The Haar Cascade will detect a face on the image by using Haar features and eliminating all the background images. After that, the Haar Cascade will return the position of the face in form of point values (x , y , width and height). After the point values are obtained, the image is cropped into rectangle form. The outcome of this process is an image with the only face area on is obtained. To detect the eye's position, the input image should be in a grayscale version. For the eye's detection method, the OpenCV will detect both eyes' positions and store the coordinate values in an array form. In the array, it contains four values; x , y , width and height. In OpenCV, the top-left point is the (0,0) point so to decide which eye is left or right is by comparing the x coordinate value that is stored in the two arrays. Figure 4 shows the pseudocode:

-
- 1: **Input: Array**, two arrays (x_1, y_1, w_1, h_1) and (x_2, y_2, w_2, h_2)
 - 2: **if** $(x_1) < (x_2)$
 - 3: Left_eye coordinate = (x_l, y_l)
 - 4: Right_eye coordinate = (x_r, y_r)
 - 5: **else**
 - 6: Left_eye coordinate = (x_l, y_l)
 - 7: Right_eye coordinate = (x_r, y_r)
-

Figure 4 Determine the Left and Right Eye

Next, find the centre of the left and right eye by dividing the coordinate values obtained from the left and right eye. By doing this, the coordinate of the centre of the left and right eye are found (Refer to Figure 5).

-
- 1: **Input: Array**, two arrays, right_eye (x_r, y_r, w_r, h_r) and left_eye (x_l, y_l, w_l, h_l) , where r is right and l is left
 - 2: Left_eye_center,

$$(x_{CL}, y_{CL}) = \left(\left(x_l + \left(\frac{w_l}{2} \right) \right), \left(y_l + \frac{h_l}{2} \right) \right)$$
Right_eye_center,

$$(x_{CR}, y_{CR}) = \left(\left(x_r + \left(\frac{w_r}{2} \right) \right), \left(y_r + \frac{h_r}{2} \right) \right)$$
-

Figure 5 Find Coordinate of the Centre of the Eyes

After obtaining the centre of the two eyes (x_{CR}, y_{CR}) and (x_{CL}, y_{CL}) , the image needs to be rotated because some images are most likely incline or tilt. After all, face alignment aims to provide a normalised input image. If the left eye is above the right eye, the image needs to be rotated anti-clockwise and if the right eye is above the left eye the image needs to be rotated clockwise. To determine the rotation either clockwise or anti-clockwise, the position of the eye is important. Figure 6 shows how to determine whether the image needs to be rotated clockwise or anti-clockwise.

-
- 1: **Input: center eyes coordinate**, two coordinates, (x_{CR}, y_{CR}) and (x_{CL}, y_{CL})
 - 2: **if** $y_{CL} < y_{CR}$
 - 3: Third_point = (x_{CR}, y_{CL})
 - 4: Direction = -1, **Rotate clockwise**
 - 5: **Else**
 - 6: Third_point = (x_{CL}, y_{CR})
 Direction = 1, **Rotate anticlockwise**
-

Figure 6 Determine the Image Rotation

Figure 6 explained how the direction of rotation is determined and from that, a third point is also obtained. Summarize all the points obtained, (x_{CR}, y_{CR}) , (x_{CL}, y_{CL}) and $(x_{CR}, y_{CL}) / (x_{CL}, y_{CR})$ will form a right-angle triangle as shown in Figure 7. Figure 7 shows the right-angle triangle. To understand how the formula is found, Figure 7 will be as a reference:

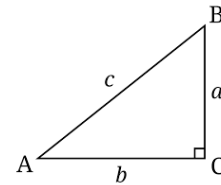


Figure 7 Right Angle Triangle

After knowing where to rotate the image either clockwise or anti-clockwise, the next step is to find how much the image needs to be rotated, which means the angle of rotation, A° . Because of that, a little trigonometry is applied to find the angle of rotation. Figure 7, assume A as a point of the center of the left eye and B as a point of the center of the right eye. Figure 6 shows that the right eye is above the left eye, so the image needs to be rotated clockwise.

By applying a cosine rule, the angle of rotation, A° can be found. Equation 2 below shows the formula to find the angle of rotation, A° . Where b , c and a in the distance or length of the triangle.

$$\cos(A) = \frac{(b^2 + c^2 - a^2)}{2bc} \quad (2)$$

By using a Euclidean distance to find the distance between the two points i.e., c and b . Equation 3 below is the Euclidean distance formula:

$$d(b, c) = \sqrt{\sum_{i=1}^n (b_i - c_i)^2} \quad (3)$$

After obtaining the distance between the two eyes, by using a cosine rule to find the angle that needs to be rotated. From that, a right-angle triangle is built. As we know, for the right-angle triangle one angle is 90 degrees and the sum of the two other angles will be 90 degrees as well. If the image needs to rotate clockwise, then 90 degrees' minus with the found angle. Then rotate the image using the angle found by minus it by 90 degrees. Finally, the output is a facial image that is aligned, and it is implemented to all image in the dataset so that all the image is standardised.

Cycle Generative Adversarial Network (CycleGAN)

CycleGAN consists of two different models Generative (G) and Discriminative (D) model. The G model task is to generate an image that follows the desired dataset, and the D model is to discriminate image whether it is the generated or actual dataset. The main objective of the CycleGAN model is to learn a mapping function between two different datasets X and Y , where X is the input dataset and Y is the desired dataset. The CycleGAN architecture contains two different mapping functions that act as the G model; $G : X \rightarrow Y$ and $F : Y \rightarrow X$ and for the corresponding D model; D_x – distinguish the generated output $G(x)$ from Y dataset and D_y – distinguish the inverse output from the X dataset. To regularize the network, two more losses is introduced to the network with the adversarial loss. The loss is called forward cycle consistency loss and backward cycle consistency loss.

Adversarial Loss:

The adversarial loss is applied to the mapping function $G : X \rightarrow Y$ and $F : Y \rightarrow X$ and its corresponding D_x and D_y . Equation 4 below shows how adversarial loss is applied to the $G : X \rightarrow Y$ mapping function:

$$\mathcal{L}_{GAN}(G, D_y, X, Y) = \mathbb{E}_{y \sim P_{data}(y)} [\log D_y(y)] + \mathbb{E}_{x \sim P_{data}(x)} [\log(1 - D_y(G(x)))] \quad (4)$$

Where G is the generative model and $G(x)$ is the generated images. The same adversarial loss is also introduced to the $F : Y \rightarrow X$ mapping function and its discriminator D_x .

Cycle Consistency Loss:

To reverse back the generated image into its original image is by using cycle consistency loss concept on each of the generated image $\{G(x)\}$ and $\{F(y)\}$ where the equation is described in Equation 5.

$$\mathcal{L}_{cycle}(G, F) = \mathbb{E}_{x \sim P_{data}(x)} [\|F(G(x)) - x\|_1] + \mathbb{E}_{y \sim P_{data}(y)} [\|G(F(y)) - y\|_1] \quad (5)$$

CycleGAN Objective:

The objective is to find the sum of all the loss in Equation 6:

$$\mathcal{L}(G, F, D_x, D_y) = \mathcal{L}_{GAN}(G, D_y, X, Y) + \mathcal{L}_{GAN}(F, D_x, Y, X) + \lambda \mathcal{L}_{cycle}(G, F) \quad (6)$$

Where λ is important to control the relative of two objectives. To solve Equation 7:

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim P_{data}(x)} [\log D(x)] + \mathbb{E}_{y \sim P_y(y)} [\log(1 - D(G(x)))] \quad (7)$$

Network Architecture:

For the network architecture, the concept came from Johnson et al. (2016) where it shows a magnificent result for super-resolution and neural style transfer [38]. This network consists of layers of CNN and several residual blocks. For the D model, the network used PatchGAN maps from 256 x 256 to 70 x 70 to discriminate whether the generated image $G(x)$ is real or fake.

Training Procedure:

By using TensorFlow 2.0 as a deep learning platform and python as a coding language. Even though there are many other deep learning platform TensorFlow is easier to implement and have a large community that can help with the research. For a hardware platform, a GPU GeForce RTX 2070 Super is used to enhance the deep learning performance.

Training Dataset:

UTKFace dataset is used in this research (<https://www.kaggle.com/jangedoo/utkface-new>) where this dataset provides more than 20,000 with a size of 200 x 200 face images in the wild with a single face in each of the images. UTKFace dataset also offers a long age span face image range from 0 to 116 years old. As this research is about facial aging, the UTKFace dataset is a suitable dataset to use.

The training dataset is divided into two; X and Y , where X is images from the age range of 18 to 25 years old male and Y consist of images from the age range of 58 to 70 years old male. The range for X is chosen between 18 to 25 years old is because at that age range mostly the face has a similar facial feature and the same goes for the Y .

Another set of data used in the CycleGAN testing process is a celebrity image from the google search engine. All the image is used for the CycleGAN testing process only. The testing process does not need a large number of images as the training process so only a few celebrities are chosen based on their age.

4.0 RESULT AND DISCUSSION

This section shows the result and discusses the result obtained from the face detection, face alignment, and CycleGAN process. This section is divided into two: Face Detection and Alignment, Image Translated by CycleGAN. For the first section, the face detection and alignment process show the original image and

the aligned image. The second section shows the original image and the translated image result.

Face Detection and Alignment

The result from 2D-face alignment is presented in Figure 8 where the original image is the image from the UTKFace dataset where the size of each image is 200 x 200, and the aligned image is the image that undergoes the 2D-face alignment process, and the size is the same as before.

From Figure 8, it can be observed that there is not much change after the images undergo a 2D-face alignment process. It is because all the images in the UTKFace dataset have already been cropped into a frontal image. So, the result shows that an image that already crops into a frontal image does not have to undergo a 2D-face alignment process because it will not give any change to the image. Then, images from the internet that did not undergo an alignment are used to see the 2D-face alignment performance.

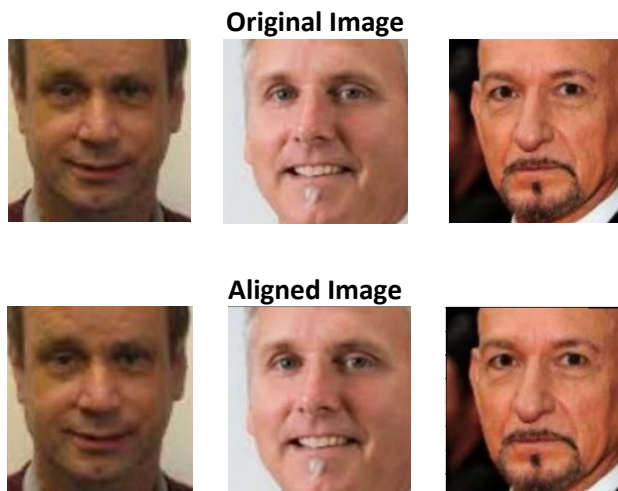


Figure 8 Face Detection and Alignment (UTKFace Dataset)

From Figure 9, the original image is the image from the internet and the aligned image is the image that already undergoes 2D-face alignment. From Figure 9 below it can be observed that all the original images are being cropped and aligned. From the result obtained, it shows that all the images experience image degradation. Where all the images that undergo 2D-face alignment become blurry.

Images Translated by CycleGAN

Figure 10 and Figure 11 below show the result of the CycleGAN network image-to-image translation process where the original is the image from the UTKFace dataset, and the translation image is the image being translated by the CycleGAN network model. From the result, it can be observed that an image of old people can be successfully translated to much younger people and vice versa. The result shows that the CycleGAN model network successfully translated all the images either from old to young and young to old.

As mentioned before this CycleGAN network model is using a CNN and a residual network model in the generative (G) and

discriminative (D) models. For the training process, it has been set to train until 200 epochs. All the images are resized to 256 x 256 and then crop to make sure that all the images that undergo the training process are at the same size, even though all the images already undergo a 2D-face alignment.

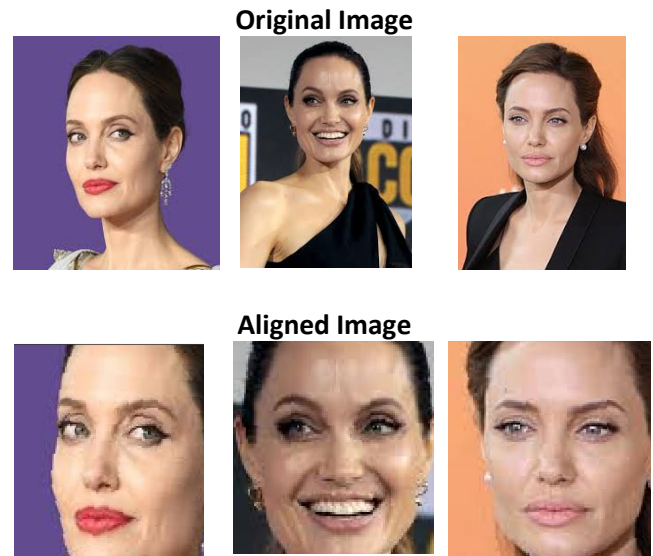


Figure 9 Face Detection and Alignment (Internet Image)

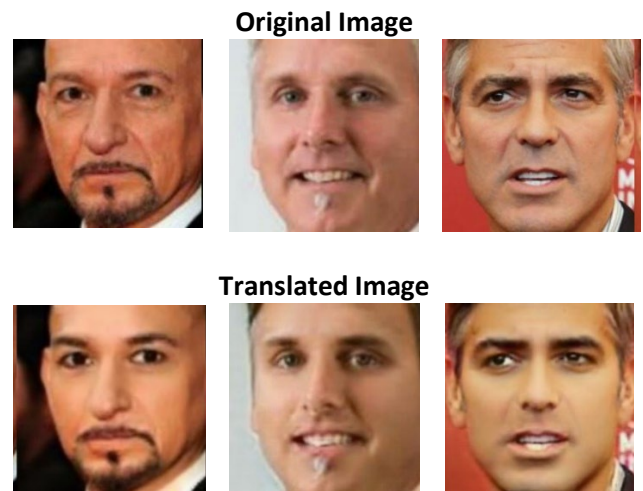


Figure 10 CycleGAN image-to-image Translation (UTKFace Dataset) – Old to Younger Translation

Figure 12 shows the CycleGAN image-to-image translation by using an image from the internet. All the images already undergo 2D-face alignment. From the result, it can be observed that the translated image becomes blurry and not as sharp as the image from the UTKFace dataset. Even though CycleGAN can translate the internet image from the young to the old version, but the generated image (by the CycleGAN model) is blurry and not as sharp as the generated image from the UTKFace dataset. It is because the image from the internet that undergoes 2D-face alignment experiences degradation. The internet image experiences degradation because the image is too far from the camera and not a frontal image. After the alignment process, some feature from the image is lost and the

image experience some degradation. Because of that, the CycleGAN performance decreases as the input image is not sharp and experiences degradation.

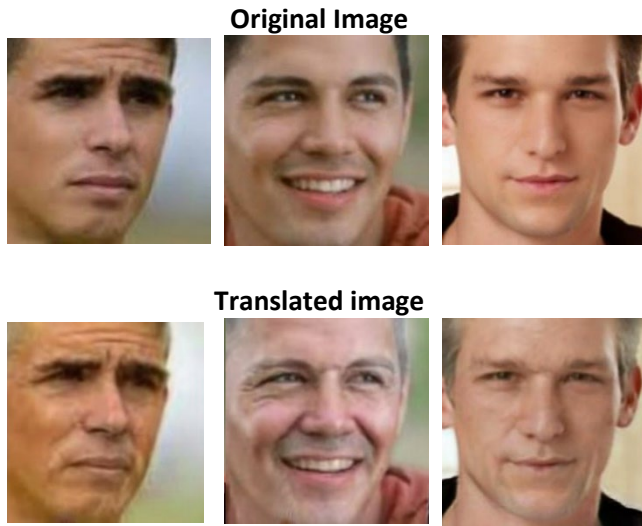


Figure 11 CycleGAN image-to-image Translation (UTKFace Dataset) – Young to Older Translation

Figure 13 shows the image-to-image translation by using an image from the internet that did not undergo 2D-face alignment. The result can be observed that the translated image becomes blurry and does not show a translation of the image from young to old. From the result shown in Figure 12 and Figure 13, it can be observed that an image from the internet that undergoes 2D-face alignment shows a good image translation from young to older rather than the internet image that did not undergo a 2D-face alignment. By undergoing the

2D-face alignment a CycleGAN model network successfully translates the image even though the image is blurry and experiences degradation. It shows that if the CycleGAN model network is provided with a frontal image that is not too far from the camera the network model can successfully translate the image.

5.0 CONCLUSION

In a conclusion, the CycleGAN model network successfully translates the image from the UTKFace dataset from old face to young face and vice versa. The face alignment works well with the UTKFace dataset and gives a good result in CycleGAN image translation. The result shows that using a face alignment can improve the CycleGAN image translation process. As the UTKFace dataset also uses a frontal image makes the result even better. The UTKFace dataset that already provides a very good frontal face image makes it easier for the CycleGAN network model to translate the image beside the 2D-face alignment process making it shows a very impressive result. An image from the internet that undergoes 2D-face alignment experiences degradation. Because of that, the result from the CycleGAN process is not as good as the result when using the UTKFace dataset. But compare with the internet image that did not undergo a 2D-face alignment it shows that the aligned image is translated much better. An aligned image is a good data representation where it can increase and boost CycleGAN model network performance. In future work, a good data alignment is needed to increase the CycleGAN model network performance. So, it can be concluded that if the image provided is a frontal image it makes the CycleGAN perform better and give a very impressive result.

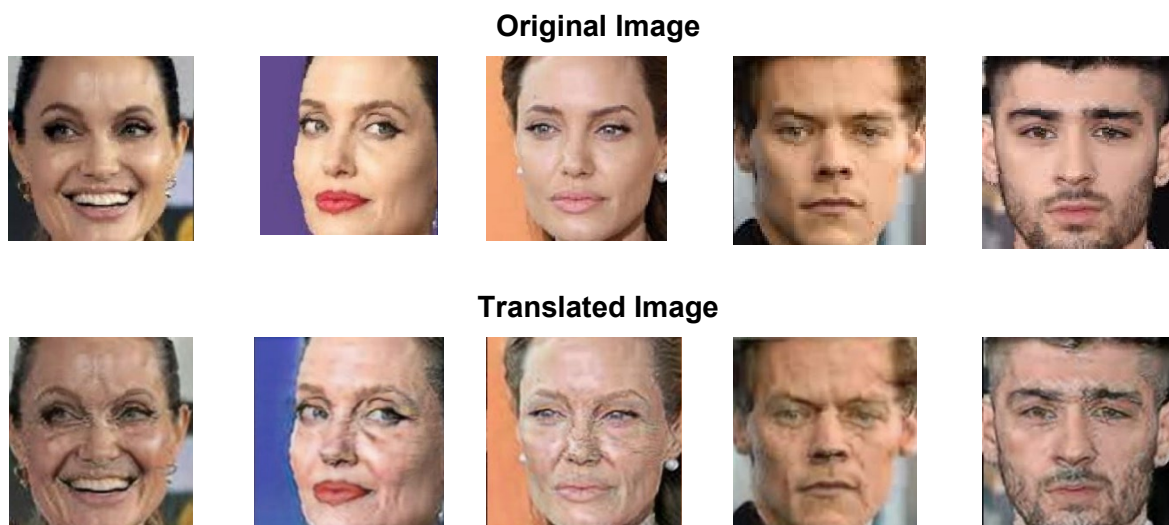


Figure 12 CycleGAN image-to-image Translation with 2D-Alignment (Internet Image)



Figure 13 CycleGAN image-to-image Translation without 2D-Alignment (Internet Image)

Acknowledgement

We would like to extend our gratitude to Universiti Teknologi Mara (UiTM) and to those who have directly or indirectly contributed to our research. This work was supported by the Trans-Disciplinary Research Grant Scheme (600-IRMI/TRGS 5/3 (001/2019)-1), Ministry of Higher Education Malaysia.

References

- [1] P. Isola, Y. Zhu, T. Zhou, A. A. Efros. 2016. Image-to-Image Translation with Conditional Adversarial Networks. *Proceeding of the IEEE International Conference on Computer Vision*. 1125-1134. DOI: <https://doi.org/10.1109/ISAPE.2012.6408893>
- [2] Z. Yi, H. Zhang, P. Tan, and M. Gong. 2017. DualGAN: Unsupervised Dual Learning for Image-to-Image Translation. *Proceeding of the IEEE International Conference on Computer Vision*. 2849-2857. DOI: <https://doi.org/10.1109/ICCV.2017.310>
- [3] N. Hamzah and F. H. Kamaru Zaman. 2020. Face Aging on Implementation Realistic Photo in Cross-Dataset. *IOP Conference Series: Material Science and Engineering*. 917(012080): 7. DOI: <https://doi.org/10.1088/1757-899X/917/1/012080>
- [4] I. J. Goodfellow. 2014. Generative Adversarial Networks. *STAT*. 1050:10. DOI: <http://www.cs.utoronto.ca/~bonner/courses/2020s/csc2547/week5/GANs,-goodfellow,-nips2014.pdf>
- [5] A. E. A. Zhu Jun-Yan, Taesung Park, Isola Phillip. 2010. Unpaired Image-to-Image Translation using Cycle-Consistent Network. *Proceeding of the IEEE International Conference on Computer Vision*. 183-202. DOI: https://doi.org/10.1007/978-1-60327-005-2_13
- [6] J. Song, J. Zhang, L. Gao, X. Liu, and H. T. Shen. 2017. Dual Conditional GANs for Face Aging and Rejuvenation. *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*. 899-905. DOI: <https://doi.org/10.24963/ijcai.2018/125>
- [7] Y. Choi, M. Choi, M. Kim, J. W. Ha, S. Kim, and J. Choo. 2018. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 8789-8797. DOI: <https://doi.org/10.1109/CVPR.2018.00916>
- [8] A. Heljakka, A. Solin, and J. Kannala. 2019. Pioneer Networks: Progressively Growing Generative Autoencoder. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 22-38. DOI: https://doi.org/10.1007/978-3-030-20887-5_2
- [9] C. Ledig et al. 2017. Photo-realistic single image super resolution using Generative Adversarial Network. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition*. 105-114. DOI: <https://doi.org/10.1109/CVPR.2017.19>
- [10] T. O. Liu Ming-Yu. 2016. Coupled Generative Adversarial Networks. *30th Conference on Neural Information Processing System*. 469-477. DOI: <https://doi.org/10.1177/016173467900100106>
- [11] E. Zhou, Z. Cao, and Q. Yin. 2015. Naive-Deep Face Recognition: Touching the Limit of LFW Benchmark or Not?. *arXiv preprint*. arXiv:1501.04690. DOI: https://ui.adsabs.harvard.edu/link_gateway/2015arXiv150104690/Z/arxiv:1501.04690
- [12] X. Zhu, X. Liu, Z. Lei, S. Z. Li, and H. Shi. 2016. Face Alignment Across Large Poses: A 3D Solution. *Proceeding of the IEEE International Conference on Computer Vision and Pattern Recognition*. 146-155. DOI: <https://doi.org/10.1109/CVPR.2016.2>
- [13] T. F. Cootes, C. J. Taylor, and A. Lanitis. 2013. Active Shape Models: Evaluation of a Multi-Resolution Method for Improving Image Search. In *Proceeding of The British Machine Vision (BMVC)*. 1:32.1-32.10. DOI: <https://doi.org/10.5244/C.8.32>
- [14] D. Cristinacce and T. Cootes. 2007. Boosted Regression Active Shape Models. In *Proceeding of The British Machine Vision (BMVC)*. 2:880-889. DOI: <https://doi.org/10.5244/C.21.79>
- [15] T. F. Cootes, G. J. Edwards, and C. J. Taylor. 2001. Active Appearance Models. *Proceeding of the IEEE transactions on pattern analysis and machine intelligence*. 23(6):681-685. DOI: https://doi.org/10.1007/978-981-10-7593-3_8
- [16] J. Matthews and S. Baker. 2004. Active appearance models revisited. *International Journal of Computer Vision*. 60(2):135-164. DOI: <https://doi.org/10.1023/B:VISI.0000029666.37597.d3>
- [17] N. Hamzah, F. H. Kamaru Zaman, and N. Md Tahir. 2021. *Journal of Electrical & Electronic Systems Research*. 19(OCT2021):7-16. DOI: <https://doi.org/10.24191/jeesr.v19i1.002>
- [18] S. Lucey, Y. Wang, M. Cox, S. Sridharan, and J. F. Cohn. 2009. Efficient constrained local model fitting for non-rigid face alignment. In *the Image and Vision Computing*. 27(12):1804-1813. DOI: <https://doi.org/10.1016/j.imavis.2009.03.002>
- [19] S. W. Chew, P. Lucey, S. Lucey, J. Saragih, J. F. Cohn, and S. Sridharan. 2011. Person-independent facial expression detection using Constrained Local Models. 2011 IEEE International Conference on Automatic Face and Gesture Recognition and Workshops, FG 2011. 915-920. DOI:

- <https://doi.org/10.1109/FG.2011.5771373>
- [20] Q. Liu, J. Deng, J. Yang, and G. Liu. 2016. Adaptive Cascade Regression Model for Robust Face Alignment. *IEEE Transactions on Image Processing*. 26(2):797-807. DOI: <https://doi.org/10.1109/TIP.2016.2633939>
- [21] F. Liu, D. Zeng, Q. Zhao, and X. Liu. 2016. Joint face alignment and 3D face reconstruction. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). 9909 LNCS:545-560. DOI: https://doi.org/10.1007/978-3-319-46454-1_33
- [22] X. Cao, Y. Wei, F. Wen, and J. Sun. 2014. Face alignment by explicit shape regression. *International journal of computer vision*. 107(2):177-190. DOI: <https://doi.org/10.1007/s11263-013-0667-3>
- [23] S. Zhu, C. Li, C. C. Loy, and X. Tang. 2015. Face Alignment by Coarse-to-Fine Shape Searching. Proceedings of the *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 07-12-June:4998-5006. DOI: <https://doi.org/10.1109/CVPR.2015.7299134>
- [24] H. Yang, S. Member, X. He, X. Jia, and S. Member. 2015. Robust Face Alignment Under Occlusion via Regional Predictive Power Estimation. *Proceeding of IEEE Transaction on Image Processing*. 24(8):2393-2403. DOI: <https://doi.org/10.1109/TIP.2015.2421438>
- [25] Q. Liu, J. Deng, and D. Tao. 2016. Dual Sparse Constrained Cascade Regression for Robust Face Alignment. *Proceeding of IEEE Transaction on Image Processing*. 25(2):700-712 DOI: <https://doi.org/10.1109/TIP.2015.2502485>
- [26] M. Kowalski, J. Naruniec, and T. Trzcinski. 2017. Deep Alignment Network : A Convolutional Neural Network for Robust Face Alignment. *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 88-97. DOI: <https://doi.ieeecomputersociety.org/10.1109/CVPRW.2017.254>
- [27] W. Wu and S. Yang. 2017. Leveraging Intra and Inter-Dataset Variations for Robust Face Alignment. *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 150-159. DOI: <https://doi.org/10.1109/CVPRW.2017.261>
- [28] H. Liu, J. Lu, S. Member, M. Guo, and S. Wu. 2020. Learning Reasoning-Decision Networks for Robust Face Alignment. *Proceeding of IEEE Transactions on Pattern Analysis and Machine Intelligence*. 42(3):679-693. DOI: <https://doi.org/10.1109/TPAMI.2018.2885298>
- [29] P. Welander, S. Karlsson, and A. Eklund. 2018. Generative Adversarial Networks for Image-to-Image Translation on Multi-Contrast MR Images - A Comparison of CycleGAN and UNIT. Computing Research Repository. abs/1806.07777. DOI: <https://doi.org/10.48550/arXiv.1806.07777>
- [30] J. Y. Zhu et al. 2017. Toward multimodal image-to-image translation. *Advances in Neural Information Processing Systems*. 2017-Decem(1):466-477. DOI: <https://doi.org/abs/10.5555/3294771.3294816>
- [31] H. Zhang et al. 2017. StackGAN : Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks. *Proceeding of IEEE International Conference on Computer Vision*. 5907-5915. DOI: <https://doi.org/10.1109/ICCV.2017.629>
- [32] S. Reed, Z. Akata, X. Yan, and L. Logeswaran. 2016. Generative Adversarial Text to Image Synthesis. Proceedings of The 33rd *International Conference on Machine Learning*. PMLR 48:1060-1069. DOI: <https://doi.org/abs/10.5555/3045390.3045503>
- [33] E. Pantraki and C. Kotropoulos. 2018. Face Aging as Image-to-Image Translation using Shared-Latent Space Generative Adversarial Networks. 2018 *IEEE Global Conference on Signal and Information Processing*. 306-310. DOI: <https://doi.org/10.1109/GlobalSIP.2018.8646447>
- [34] Z. Wang, X. Tang, W. Luo, and S. Gao. 2018. Face Aging with Identity-Preserved Conditional Generative Adversarial Networks. *Proceeding of IEEE Conference on Computer Vision and Pattern Recognition*. 7939-7947. DOI: <https://doi.org/10.1109/CVPR.2018.00828>
- [35] S. Liu et al. 2017. Face Aging with Contextual Generative Adversarial Nets. Proceedings of the 25th ACM international conference on Multimedia. 82-90. DOI: <https://doi.org/10.1145/3123266.3123431>
- [36] C. Han, H. Hayashi, L. Rundo, R. Araki, and W. Shimoda. 2018. GAN-Based Synthetic Brain MR Image Generation. 2018 *IEEE 15th International Symposium on Biomedical Imaging*. 734-738. DOI: <https://doi.org/10.1109/ISBI.2018.8363678>
- [37] C. Han, L. Rundo, R. Araki, and Y. Furukawa. 2018. Infinite Brain Tumor Images : Can GAN-based Data Augmentation Improve Tumor Detection on MR Images ?. In *Proceeding Meeting on Image Recognition and Understanding*. 1-4. DOI: http://133.11.169.134/pdf/MIRU_GAN.pdf
- [38] J. Johnson, A. Alahi, and L. Fei-fei. 2016. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. European Conference on Computer Vision. 9906:694-711. DOI: https://doi.org/10.1007/978-3-319-46475-6_43
- [39] M. Rosca, B. Lakshminarayanan, D. Warde-Farley, and S. Mohamed. 2017. Variational Approaches for Auto-Encoding Generative Adversarial Networks. *arXiv preprint*. arXiv:1706.04987. DOI: <https://doi.org/10.48550/arXiv.1706.04987>
- [40] A. Radford, L. Metz, and S. Chintala. 2015. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. Lecture Notes in Computer Science. 10667:1-16. DOI: https://doi.org/10.1007/978-3-319-71589-6_9
- [41] P. Nagorny et al. 2019. Generative Adversarial Networks for geometric surfaces prediction in injection molding. 2018 *IEEE International Conference on Industrial Technology*. 1514-1519. DOI: <https://doi.org/10.1109/ICIT.2018.8352405>
- [42] A. Conneau, H. Schwenk, Y. Le Cun, and B. Loic. 2017. Very Deep Convolutional Networks for Text Classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*. 1:1107-1116. DOI: <https://doi.org/10.18653/v1/e17-1104>
- [43] K. Kowsari, D. E. Brown, and M. Heidarysafa. 2017. HDLTex : Hierarchical Deep Learning for Text Classification. 2017 16th *IEEE International Conference on Machine Learning and Applications*. 364-371. DOI: <https://doi.org/10.1109/ICMLA.2017.0-134>
- [44] V. Singh, V. Shokeen, and B. Singh. 2013. Face Detection by Haar Cascade Classifier with Simple and Complex Background Images using OpenCV Implementation. *International Journal of Advanced Technology in Engineering and Science*. 1(12):33-38. DOI: http://www.ijates.com/images/short_pdf/1530959991_P33-38.pdf
- [45] P. I. Wilson and J. Fernandez. 2006. Facial feature detection using Haar Classifiers. *Journal of computing sciences in colleges*. 21(4):127-133. DOI: <https://doi.org/abs/10.5555/1127389.1127416>