

PREDIKSI PENUAAN WAJAH MANUSIA BERBASIS *GENERATIVE ADVERSARIAL NETWORK*

Beladina Elfriti¹, Ema Rachmawati^{*2}, Tjokorda Agung Budi Wirayuda³

^{1,2,3}Universitas Telkom

Email: ¹beladinaelfriti@student.telkomuniversity.ac.id, ²emarachmawati@telkomuniversity.ac.id,

³tokagung@telkomuniversity.ac.id

^{*}Penulis Korespondensi

(Naskah masuk: 29 Desember 2022, diterima untuk diterbitkan: 4 Desember 2023)

Abstrak

Karena struktur wajah manusia yang berbeda-beda, wajah merupakan salah satu ciri yang digunakan untuk mengidentifikasi seseorang. Wajah sering digunakan sebagai pengenalan biometrik. Namun, seiring bertambahnya usia manusia, wajah mereka bisa berubah karena faktor lingkungan dan gaya hidup. Karena efek penuaan pada wajah, komputer tidak dapat mengenali kemiripan antara citra wajah dari orang yang sama pada usia yang berbeda. Penelitian pengenalan wajah biasanya menggunakan data berpasangan (*paired data*), yang sangat sulit didapat. Di sisi lain, volume data yang tidak berpasangan (*unpaired data*) sangat besar dan mudah diakses. Sebaliknya, keterbatasan data berpasangan memotivasi para peneliti untuk mengembangkan teknik sintesis citra yang tidak bergantung pada data berpasangan. Tanpa perlu data berpasangan, metode CycleGAN mampu menghasilkan citra sintetik yang lebih realistis dengan resolusi lebih tinggi. Hal itulah yang memotivasi penelitian ini dalam penggunaan data tidak berpasangan untuk memprediksi penuaan wajah manusia menggunakan CycleGAN. Pada penelitian ini, digunakan citra dari dataset UTKFace yang terdiri atas citra wajah berbagai usia. Untuk keperluan eksperimen, citra dari UTKFace dibagi ke dalam dua ranah, yaitu citra wajah usia muda dan citra wajah usia tua, untuk keperluan sistem penuaan wajah yang dibangun. Dengan demikian, citra wajah berusia muda tidak memiliki pasangan pada citra wajah usia tua (*unpaired data*). Dengan nilai *Frechet Inception Distance* (FID) = 2,24, hasil percobaan menunjukkan bahwa metode yang digunakan mampu mencapai kinerja yang sangat baik pada sistem penuaan wajah yang dibangun.

Kata kunci: *CycleGAN, penuaan wajah, Frechet Inception Distance, data tidak berpasangan*

HUMAN FACIAL AGING PREDICTION BASED ON GENERATIVE ADVERSARIAL NETWORK

Abstract

One feature that is utilized to identify a person is their face because of the differences in its structure. Biometric IDs that employ faces are common. Yet, aging-related environmental and lifestyle changes can cause a person's face to alter. Compiling facial photos of the same individual at different ages is impossible for computers to discern similarity because of the effects of aging on the face. Research on facial recognition, on the other hand, typically utilizes paired data, which is exceedingly hard to come by. However, the amount of unpaired data is readily available and quite vast. The constraints of paired data, on the other hand, drive researchers to create methods for image synthesis that do not depend on paired data. Higher quality, more realistic synthetic images can be produced with the CycleGAN approach without requiring paired data. That is what drives this research, which uses CycleGAN to forecast human facial aging from unpaired data. The UTKFace dataset, which includes face images of people of different ages, provided the images utilized in this investigation. For experimental purposes, the facial aging system being developed divides images from UTKFace into two domains: young facial images and senior facial images. Consequently, there is no corresponding image of an aged face to an image of a youthful face (*unpaired data*). The testing results demonstrate that the built-in face aging system may achieve good performance using the method employed, with a *Frechet Inception Distance* (FID) value of 2.24.

Keywords: *CycleGAN, face aging, Frechet Inception Distance, unpaired data*

1. PENDAHULUAN

Perkembangan teknologi dan data yang mulai didigitalisasi membuat penelitian dalam bidang computer vision ikut berkembang pesat. Penelitian menggunakan citra wajah juga banyak digunakan dalam visi komputer, salah satunya yaitu prediksi penampilan wajah seseorang di masa depan. Berkaitan dengan hal tsb, penelitian tentang penuaan wajah (face aging) telah menarik banyak minat penelitian dalam beberapa tahun terakhir. Sistem prediksi penuaan wajah ini pun telah digunakan dan masih akan terus berkembang di sejumlah bidang, misalnya digunakan untuk sistem pengawasan (surveillance) (Sawant & Bhurchandi, 2019), dunia hiburan (Fu, Guo & Huang, 2010, Anon., 2023), industri kosmetik industry (Despois, Flament & Perrot, 2020), dan seniman forensik telah menggunakannya untuk mensimulasikan penampilan orang yang dicari atau hilang (Kemelmacher-Shlizerman, Suwajanakorn & Seitz, 2014).

Face Aging atau biasa dikenal dengan Age Progression (Ren dkk., 2017) merupakan sistem untuk memprediksi bagaimana wajah seseorang akan terlihat di umur yang berbeda. Face aging atau penuaan wajah masih menjadi tantangan (Fu, Guo & Huang, 2010) karena kurangnya citra wajah untuk orang yang sama pada usia yang berbeda, kondisi pencahayaan yang berbeda, ekspresi dan atribut wajah yang berbeda seperti adanya kumis atau kacamata pada data citra wajah.

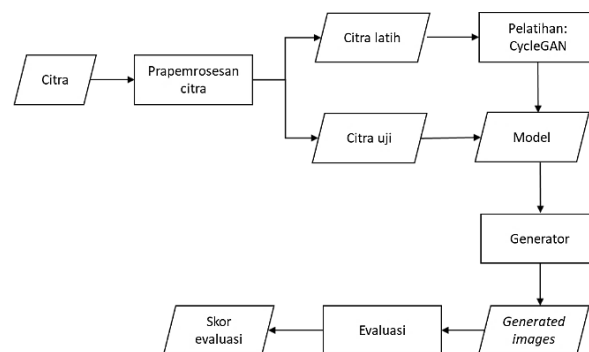
Pendekatan yang telah dilakukan para penelitian untuk menyelesaikan permasalahan prediksi penuaan wajah umumnya menggunakan metode tradisional berupa metode model fisik dan prototype. Pendekatan model fisik akan mempelajari parameter wajah seperti bentuk dan tekstur di setiap kelompok umur lalu digunakan untuk memodelkan kembali citra wajah masukan (Shu dkk., 2015). Pada pendekatan prototype (Kemelmacher-Shlizerman, Suwajanakorn & Seitz, 2014) wajah dibagi berdasarkan umur dan menggunakan perbedaan tekstur antara domain wajah asal dengan target wajah. Implementasi dari pendekatan tradisional lebih sederhana namun citra yang dihasilkan kurang realistis dan membutuhkan data yang sulit untuk dikumpulkan.

Sementara itu, Generative Adversarial Networks (GAN) yang diusulkan (Goodfellow dkk., 2014) mampu menghasilkan membuat citra sintesis yang realistis dan memiliki resolusi tinggi. Beberapa pengaplikasian GAN yaitu dalam image generation (Radford, Metz & Chintala, 2015), text2image (Reed dkk., 2016) dan image editing (Zhu dkk., 2016). GAN terdiri dari sebuah Generator G dan sebuah diskriminator D. Generator bertugas untuk menghasilkan citra sesuai target citra, sedangkan diskriminator bertugas untuk menilai apakah citra yang dihasilkan generator merupakan citra asli atau palsu. Diskriminator dilatih sampai generator dapat

menghasilkan citra yang menyerupai target citra asli dan diskriminator tidak dapat membedakan keasliannya dengan citra rekonstruksi dengan cara memaksimalkan fungsi *adversarial loss*. Sebaliknya, generator berusaha meminimalkan fungsi *loss* agar dapat menghasilkan citra yang mirip dengan citra asli. Penelitian sebelumnya berhasil mengaplikasikan *contextual GAN* (Liu dkk., 2017) dan *conditional GAN* (Antipov, Baccouche & Dugelay, 2017; Sheng dkk., 2020) pada prediksi penuaan wajah. Namun beberapa metode berbasis GAN tersebut masih memerlukan *paired data*. Beberapa tahun kemudian terdapat metode *cycle-consistent generative adversarial network* (CycleGAN) yang berhasil mentranslasi citra menggunakan data yang tidak memiliki pasangan (*unpaired data*). CycleGAN (Zhu dkk., 2017) berhasil mentranslasi citra zebra ke kuda, citra apel ke jeruk serta citra pemandangan alam ke lukisan bergaya monnet dan sebaliknya.

Berdasarkan hal tersebut, keberhasilan penelitian tsb menjadi motivasi agar dapat mengatasi keterbatasan tersedianya *paired data*. Oleh karena itu pada penelitian ini diusulkan penggunaan metode CycleGAN untuk membuat sistem yang dapat memprediksi penuaan wajah dengan menggunakan *unpaired data*.

Struktur dari penulisan penelitian ini adalah sebagai berikut. Bagian 2 membahas implementasi dari model yang dibangun menggunakan metode CycleGAN. Bagian 3 membahas hasil dari pengujian model dan evaluasi model. Bagian 4 membahas kesimpulan dari penelitian yang telah dilakukan.



Gambar 1. Sistem prediksi penuaan wajah menggunakan CycleGAN

2. PREDIKSI PENUAAN WAJAH MENGGUNAKAN CYCLEGAN

Secara umum pembangunan sistem dapat dilihat pada Gambar 1. Sistem dimulai dari prapemrosesan yang menghasilkan data uji dan data latih. Data latih menjadi masukan pada proses pelatihan menggunakan CycleGAN. Dari CycleGAN akan diambil generator yang sudah berhasil dilatih untuk menghasilkan citra sesuai target domain. Proses pelatihan selesai pada tahap ini. Setelah itu data akan diuji dengan memasukkan data uji ke dalam

generator. Setelah itu akan dievaluasi hasilnya. Kemudian skor hasil metrik akan dihasilkan dan sistem selesai dijalankan.

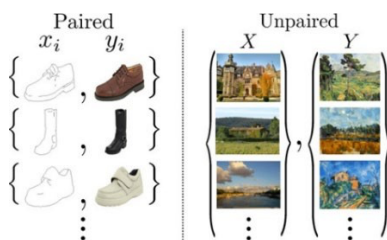
2.2.1. Prapemrosesan Data

Prapemrosesan pertama yang dilakukan yaitu mengubah ukuran citra menjadi berukuran 256x256x3 agar citra yang dihasilkan dapat memiliki resolusi yang baik. Citra memiliki nilai piksel 0-255, angka yang terlalu besar ini dapat memperlambat proses pelatihan. Oleh karena itu citra akan dinormalisasi dengan mengubah nilai piksel pada citra dari rentang nilai 0-255 menjadi -1 hingga 1. Normalisasi dilakukan untuk mempercepat proses pelatihan jaringan. Kumpulan data akan dibagi ke dalam data latih dan data uji secara random dengan perbandingan 9:1, sehingga didapatkan data latih berjumlah 2.827 citra dan data uji sebanyak 315 citra.

2.2. CycleGAN

CycleGAN (Zhu dkk., 2017) merupakan pengembangan dari GAN yang mampu mentranslasi citra dari domain asal ke target dengan mempelajari pemetaan karakteristik data dari domain asal ke domain target dan sebaliknya. CycleGAN memiliki 2 (dua) generator dan 2 (dua) diskriminator. Dua diskriminator merepresentasikan fungsi $D_A: A \rightarrow \mathbb{R}$ dan $D_B: B \rightarrow \mathbb{R}$. Sementara itu, dua generator merepresentasikan fungsi $G_A: A \rightarrow B$ dan $G_B: B \rightarrow A$.

G_A memetakan citra dari domain A ke B. D_B menerima citra asli dari B dan citra palsu yang dibangkitkan G_A dan memprediksi apakah citra tersebut asli atau palsu. Citra yang dibangkitkan oleh G_A dipetakan kembali ke domain asli oleh G_B . Citra yang dibangkitkan G_B seharusnya merepresentasikan citra asli yang menjadi input G_A . Dalam hal ini, *cycle consistency loss* meminimalkan jarak antara citra actual dengan citra hasil rekonstruksi. G_A memetakan citra ke B. Jika input merupakan hasil sampel dari B, generator harus memetakan input ke dirinya sendiri. Dalam hal ini, *identity loss* meminimalkan jarak antara input dengan citra hasil pembangkitan dengan memetakannya ke domain yang sama.



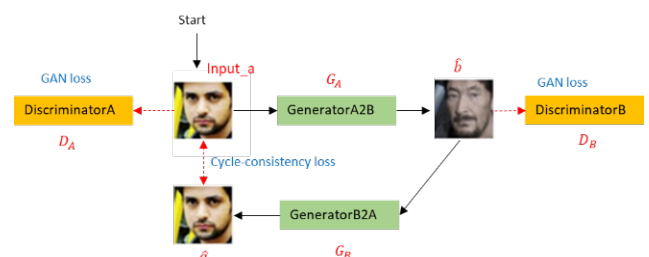
Gambar 2. Perbedaan *paired* dan *unpaired* data (Zhu et al., 2017)

Terdapatnya dua generator dan dua discriminator itulah yang membuat CycleGAN mampu menghasilkan citra sintesis dengan

menggunakan *unpaired data*. *Unpaired data* merupakan data yang domain dan target tidak memiliki pasangan serta berbeda karakteristiknya. Perbedaan *paired* dan *unpaired data* dapat dilihat pada Gambar 2. Pada umumnya, *paired data* lebih sulit untuk dikumpulkan, seperti pada contoh Gambar 2 di mana kita perlu sketsa yang mirip dengan citra aslinya.

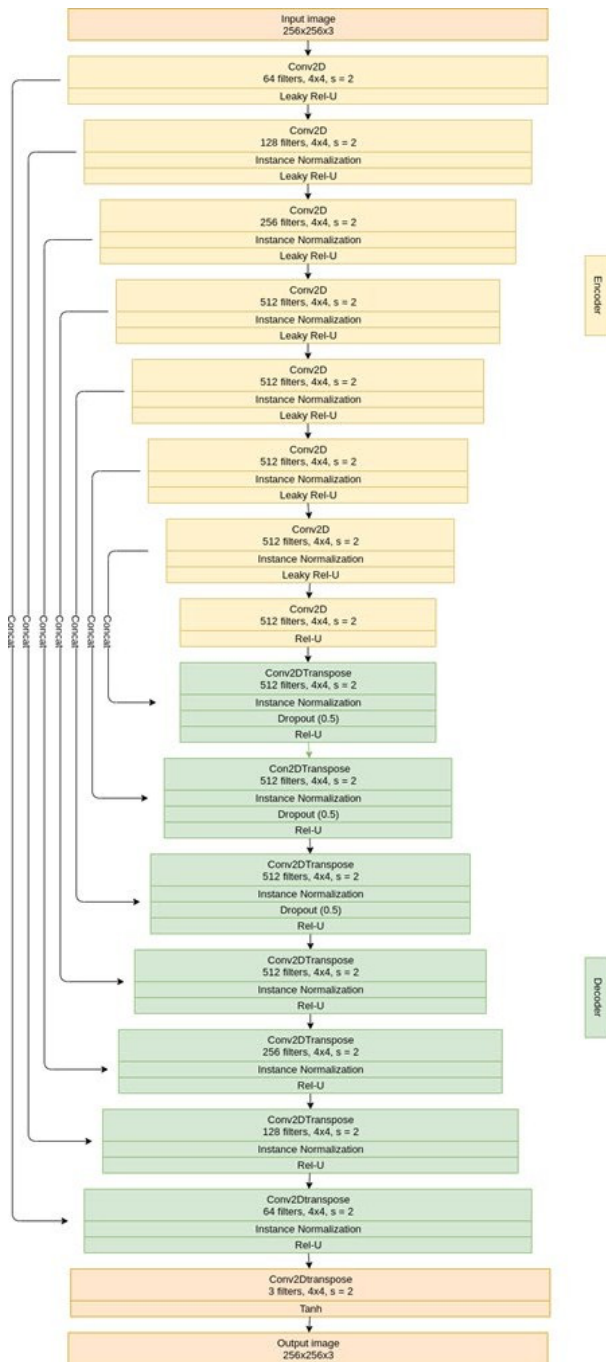
Dalam kumpulan data berpasangan, setiap citra, misal $imgA$, dipetakan secara manual ke beberapa citra di domain target, misal $imgB$, sehingga mereka berbagi berbagai fitur. Pada dasarnya, *pairing* dilakukan untuk membuat input dan *output* berbagi beberapa fitur umum. Pemetaan ini mendefinisikan transformasi citra yang bermakna dari satu domain ke domain lain. Jadi, ketika kita telah memasang dataset, generator harus mengambil input, misal $inputA$, dari domain DA dan memetakan citra ini ke citra keluaran, yaitu $genB$, yang harus dekat dengan pasangannya yang dipetakan. Namun hal ini tidak dijumpai dalam kumpulan data yang tidak berpasangan. Kita perlu memastikan bahwa ada beberapa hubungan yang bermakna antara citra input dan citra yang dihasilkan. Untuk memastikan bahwa ada hubungan yang bermakna antara citra-citra ini, mereka harus berbagi beberapa fitur, yaitu fitur yang dapat digunakan untuk memetakan kembali citra keluaran ini ke citra asli, sehingga harus ada generator lain yang harus dapat memetakan kembali citra keluaran ini kembali ke citra asli.

Ilustrasi sederhana atas proses yang terjadi pada CycleGAN pada sistem prediksi penuaan wajah yang diusulkan pada penelitian ini dapat dilihat pada Gambar 3. Metode ini bekerja dengan mengambil citra masukan dari domain A ($input_a$) yang diumpungkan ke generator pertama, G_A , yang tugasnya mengubah citra yang diberikan dari domain A ke citra di domain target B, yaitu \hat{b} . Citra baru yang dihasilkan ini kemudian diumpungkan ke generator lain, G_B , yang mengubahnya kembali menjadi citra \hat{a} , dari domain asli A. Citra keluaran ini harus dekat dengan citra masukan asli untuk mendapatkan pemetaan bermakna, yang memang tidak dijumpai dalam kumpulan data yang tidak berpasangan (*unpaired data*).



Gambar 3. Blok diagram penggunaan CycleGAN pada sistem prediksi penuaan wajah. Terdapat dua generator dan 2 diskriminator yang menjadi ciri khas dari CycleGAN.

DiskriminatorB D_B mendorong agar generator G_A mentranslasikan citra input_a ke dalam citra output \hat{b} yang tidak dapat dibedakan dari citra pada domain B. Selanjutnya generator G_B akan mentranslasikan citra \hat{b} ke dalam citra pada domain A. DiskriminatorA mendorong agar generator G_B mentranslasikan \hat{b} ke dalam output \hat{a} yang tidak dapat dibedakan dari citra pada domain A. Pelatihan dilakukan terhadap arsitektur tersebut, sampai menghasilkan model generator terbaik, yang akan digunakan sebagai generator untuk membangkitkan citra penuaan wajah.



Gambar 4. Arsitektur U-Net sebagai generator

2.2.1. Generator: UNet

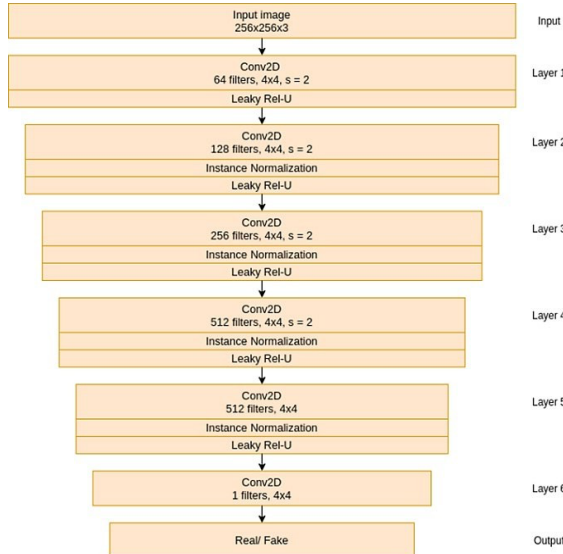
Arsitektur generator yang digunakan dalam penelitian ini yaitu U-Net (Ronneberger, Fischer & Brox, 2015) generator. U-Net generator merupakan model *encoder-decoder* dengan *skip connections* seperti yang terlihat pada Gambar 4. *Skip connections* digunakan untuk menyambungkan *layer* di dalam *encoder* dengan *layer* pada *decoder* yang memiliki *feature map* sama. Generator akan menerima masukan citra berukuran 256x256x3. Kemudian citra memasuki 8 (delapan) *layer encoder* dan 8 (delapan) *layer decoder*. Setelah itu generator menghasilkan citra berukuran 256x256x3. *Skip connections* ditambahkan pada *layer* i hingga $n-i$, dengan n merupakan jumlah total *layer*. *Layer* terakhir pada *encoder* merupakan *layer bottleneck* untuk mengurangi jumlah parameter dan perkalian matriks karena jumlah neuron yang lebih sedikit pada *layer* ini.

Pada penelitian ini, modifikasi pada U-Net dilakukan dengan menggunakan *instance normalization* daripada menggunakan *batch normalization*. *Instance normalization* bekerja lebih sederhana dengan cara menstandarisasi nilai pada setiap fitur *output*, bukan di seluruh fitur dalam satu *batch*. *Encoder* menggunakan blok *Convolution - InstanceNorm* (kecuali *layer* pertama dan terakhir) - *LeakyReLU* (kecuali *layer* terakhir). Fungsi aktivasi *LeakyReLU* digunakan untuk mempertahankan bobot kurang dari 0.

Di sisi lain, *decoder* menggunakan blok *Convolution Transpose - InstanceNorm* (kecuali *layer* terakhir) - *Dropout* (0,5) pada 3 (tiga) *layer* pertama - *ReLU*. *Layer* terakhir menggunakan konvolusi 3x3 dan fungsi aktivasi *tanh* digunakan untuk memetakan setiap fitur ke jumlah kelas yang diinginkan. Semua *layer convolution* menggunakan ukuran kernel 4x4 dan stride 2x2.

2.2.2. Diskriminator: PatchGAN

Arsitektur PatchGAN (Isola dkk., 2016) digunakan sebagai diskriminator pada penelitian ini. PatchGAN yang digunakan memiliki ukuran *patch* 70x70 yang berfungsi mengklasifikasikan apakah *patch* tersebut asli atau palsu. Blok konvolusi terdiri dari *Convolution - InstanceNorm - LeakyReLU*. Mengikuti implementasi pada (Radford, Metz & Chintala, 2015), *LeakyReLU* digunakan pada semua *layer* untuk menghasilkan model dengan resolusi yang tinggi dan proses pelatihan yang stabil. Seperti yang terlihat pada Gambar 5, citra berukuran 256x256x3 dimasukkan ke dalam diskriminator. Kemudian citra akan melalui 5 (lima) blok konvolusi. Citra tersebut selanjutnya dipetakan ke *patch* 70x70, untuk selanjutnya pada *layer* terakhir *patch* tersebut diklasifikasikan sebagai citra asli atau palsu. Pada *layer* pertama *instance normalization* tidak digunakan. Semua *layer* konvolusi menggunakan ukuran kernel 4x4 dan stride 2x2.



Gambar 5. Arsitektur PatchGAN sebagai diskriminator

2.4. Fungsi Objektif

Fungsi objektif CycleGAN (Zhu dkk., 2017) memiliki banyak perbedaan dibandingkan dengan GAN. Alih-alih menggunakan *cross-entropy* untuk *adversarial loss*, CycleGAN meminimalkan *mean squared error*. (Zhu dkk., 2017) mengklaim bahwa meminimalkan *loss* seperti itu sama dengan meminimalkan Pearson χ^2 *divergence*. Selain itu, (Zhu dkk., 2017) mengklaim pelatihan lebih stabil dan menghasilkan citra berkualitas lebih tinggi. (Zhu dkk., 2017) juga menghilangkan aktivasi sigmoid yang biasanya ada di lapisan terakhir diskriminator, sehingga kodomain untuk diskriminator adalah \mathbb{R} bukan (0, 1). Gradien fungsi sigmoid mendekati nol ketika input semakin meningkat nilainya. Artinya, jika diskriminator yakin dengan keluarannya, generator akan menerima lebih sedikit umpan balik dalam bentuk gradien. Dengan menghapus aktivasi sigmoid, masalah tersebut tidak terjadi lagi.

CycleGAN juga mengenakan fungsi *cycle consistency loss*. Fungsi ini memastikan bahwa G_A dan G_B saling terbalik (*inverse*). Fungsi *loss* ini mencegah jatuhnya model dengan memastikan bahwa tidak terlalu banyak informasi yang hilang saat translasi antar domain.

Selain itu, CycleGAN juga mengenakan *identity loss*. Sebagai contoh pada kasus menerjemahkan lukisan ke foto. *Artificial Neural Network* (ANN) dapat belajar memetakan lukisan siang hari ke foto matahari terbenam. *Cycle consistency loss* dan *adversarial loss* tidak mencegah pemetaan ini, sedangkan *identity loss* mencegahnya. Citra sampel dari domain A, ketika dipetakan ke domain yang sama, harus memberikan citra yang sama. Fungsi *loss* ini bersifat opsional dan disarankan untuk menggunakannya saat jaringan tampaknya

menghasilkan citra yang memiliki warna yang berbeda dari yang diharapkan.

Adversarial Loss

Adversarial loss (Goodfellow dkk., 2014) merupakan fungsi *loss* utama yang diaplikasikan pada generator dan diskriminator. Generator $G_B(x)$ berusaha menghasilkan citra x yang sama dengan citra yang berasal dari A . Diskriminator D_A selanjutnya mencoba membedakan citra hasil dari $G_B(x)$ dan citra asli di A . Nilai *loss* untuk proses ini disimpan dalam $L_A(D_A)$. Proses yang serupa juga terjadi untuk generator $G_A(x)$ dan diskriminator D_B , dengan nilai *loss* disimpan dalam $L_B(D_B)$. Fungsi tersebut dapat dituliskan secara matematis seperti pada persamaan (1) dan (2). Perhitungan *adversarial loss* pada penelitian ini menggunakan $L2$ norm atau *mean squared error*.

$$L_A(D_A) = D_{x \sim A}[(D_A(x) - 1)^2] + D_{x \sim B}[(D_A(G_B(x)))^2] \quad (1)$$

$$L_B(D_B) = D_{x \sim B}[(D_B(x) - 1)^2] + D_{x \sim A}[(D_B(G_A(x)))^2] \quad (2)$$

Cycle consistency loss

CycleGAN menggunakan *unpaired data* untuk melakukan proses pemetaan dari citra domain A ke domain B dan sebaliknya. Jika hanya menggunakan fungsi *adversarial loss* (Goodfellow dkk., 2014) saja, ada kemungkinan input citra domain A dan target domain B tidak akan terpetakan dengan benar. *Cycle consistency loss* dapat dihitung menggunakan persamaan (3). *Cycle consistency loss* akan menghitung perbedaan antara input citra dengan citra yang dihasilkan generator menggunakan $L1$ norm atau *mean absolute error*.

$$L_{cycle}(G_A, G_B) = \mathbb{E}_{x \sim A} [\|G_B(G_A(x)) - x\|_1] + \mathbb{E}_{x \sim B} [\|G_A(G_B(x)) - x\|_1] \quad (3)$$

Identity Loss

Identity Loss (Taigman, Polyak & Wolf, 2016) ditambahkan untuk membantu mempertahankan warna asli pada *output* citra. *Identity loss* tidak akan merubah banyak pada citra. Citra asli dan citra hasil rekonstruksi dihitung perbedaannya menggunakan $L1$ norm atau *mean absolute error*. Persamaan (4) menunjukkan cara perhitungan *identity loss*.

$$L_{id}(G_A, G_B) = D_{x \sim A}[\|G_B(x) - x\|_1] + D_{x \sim B}[\|G_A(x) - x\|_1] \quad (4)$$

Total Loss

Semua fungsi *loss* di atas selanjutnya digabungkan dengan menggunakan bobot yang berbeda sehingga menghasilkan *total loss* seperti terlihat pada persamaan (5).

$$L(D_A, D_B, G_A, G_B) = \lambda_{gg}[L_A(D_A) + L_B(D_B)] + \lambda_c L_{cycle}(G_A, G_B) + \lambda_{id} L_{id}(G_A, G_B) \quad (5)$$

Penelitian ini menggunakan λ_{gg} mewakili bobot untuk *adversarial loss* dengan $\lambda_{gg} = 1$. *Cycle consistency loss* dengan $\lambda_c = 10$ seperti implementasi pada (Zhu dkk., 2017). Sedangkan λ_{id} mewakili bobot untuk *identity loss* dengan $\lambda_{id} = 5$. Pemberian bobot pada fungsi *loss* dimaksudkan untuk memberikan tingkat prioritas kepada masing-masing fungsi *loss*, di mana bobot 10 akan dinilai lebih penting daripada bobot 1.

2.5. Pengukuran Performa Sistem

Pengukuran performa sistem dilakukan dengan menghitung nilai *Frechet Inception Distance* (FID) (Heusel dkk., 2017). FID masih relevan digunakan untuk mengevaluasi kualitas dan keberagaman citra yang dihasilkan oleh berbagai arsitektur berbasis GAN sehingga dapat digunakan untuk mengukur performa sistem yang diusulkan (Borji, 2022; Sharma, Sharma & Jindal, 2022; Korgialas dkk., 2023). FID bekerja dengan memetakan sekumpulan citra ke ruang fitur yang ditentukan oleh *layer* tertentu dari model *Inception*. Nilai aktivasi dari *layer* tsb digunakan untuk memperkirakan statistik seperti vektor rata-rata dan matriks kovarians, yang digunakan untuk membuat distribusi Gaussian multidimensi. Selanjutnya, jarak Fréchet antara distribusi yang diestimasi dari citra riil dan citra yang dibangkitkan, dihitung dan dilambangkan dengan skor FID (Heusel dkk., 2017). Semakin rendah skor FID, berarti citra yang dihasilkan sangat mirip dengan citra aslinya, sehingga menunjukkan bahwa citra visual yang dihasilkan berkualitas tinggi. Perhitungan nilai FID menggunakan persamaan (6).

$$FID(x, g) = \|\mu_x - \mu_g\|_2^2 + Tr(\Sigma_x + \Sigma_g - 2\sqrt{\Sigma_x \Sigma_g}) \quad (6)$$

x dan g merupakan *embedding* asli dan palsu, yang diasumsikan berupa distribusi normal *multivariate*.

μ_x dan μ_g merupakan *magnitude* vektor x dan g . Tr merupakan *trace of matrix*, sedangkan Σ_x dan Σ_g

adalah matriks kovarian dari vektor x dan g .

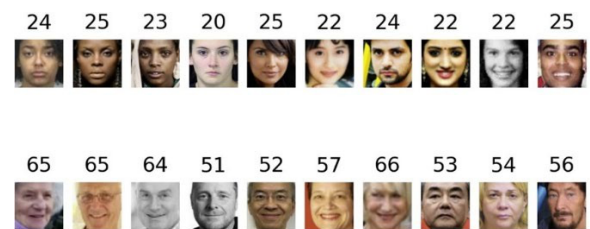
3. EKSPERIMEN DAN ANALISIS

Pada bagian ini dipaparkan data yang digunakan pada eksperimen dan hasil evaluasi beserta analisis atas eksperimen untuk memprediksi penuaan wajah.

3.1 Data

Data yang digunakan dalam penelitian ini yaitu data UTKFace (Zhang, Song & Qi, 2017) yang merupakan data berisi 23.000 citra dengan anotasi usia, jenis kelamin dan etnis dari individu yang berbeda-beda. Penelitian ini hanya menggunakan label usia dan mengabaikan label jenis kelamin dan etnis. Data yang digunakan merupakan data dengan citra yang telah dilakukan *alignment* dan *cropping* sehingga menyisakan citra bagian wajah saja. Citra pada data ini berukuran 200x200x3.

Data ini memiliki citra dengan rentang usia dari 0 hingga 116 tahun. Data dibagi ke dalam 2 (dua) domain yaitu domain A dengan citra usia muda 18-25 tahun (Sun dkk., 2020) dan domain B dengan citra usia tua 50-70 tahun. Terdapat 3.142 data untuk masing-masing domain. Total jumlah citra yang digunakan adalah 6.284 citra wajah. Penelitian ini hanya berfokus kepada prediksi penuaan wajah dengan input citra wajah RGB pada domain A menjadi citra wajah RGB domain B. Contoh citra dapat dilihat pada Gambar 6.



Gambar 6. Contoh dataset UTKFace [8] beserta label usia

3.2 Evaluasi

Dalam pelatihan dilakukan 2 (dua) skenario dengan tujuan menghasilkan model terbaik. Skenario ini dibuat untuk menemukan parameter terbaik yang dapat menghasilkan citra dengan nilai FID yang lebih kecil. Berikut adalah skenario yang dilakukan selama proses pelatihan.

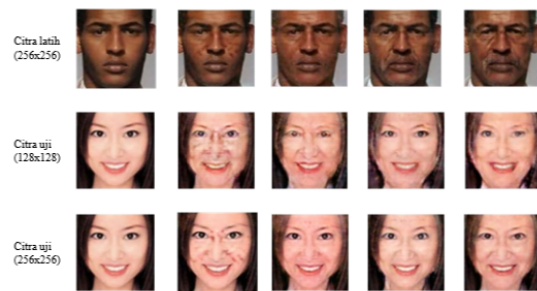
- Skenario 1: Membandingkan kinerja sistem pada input citra berukuran 128x128x3 dengan citra berukuran 256x256x3.
- Skenario 2: Membandingkan nilai FID antara learning rate 0,0002 dengan 0,0003.

Setiap skenario diimplementasikan menggunakan arsitektur yang sama dengan Adam optimizer (Kingma & Ba, 2014), $\beta = 0.5$ dan ukuran *batch*=1, sebanyak 20 epoch menggunakan NVIDIA Tesla T4 13 GB GPU selama 20 jam. Perlu diketahui

bahwa lamanya waktu training menjadi tantangan dalam melakukan eksperimen dengan berbagai skenario tersebut.

Setelah proses pelatihan model selesai, model generator akan diuji menggunakan data uji yang sudah disiapkan. Generator yang digunakan dalam pengujian ini yaitu generator G yang dapat mentranslasikan citra domain A ke target domain B.

Pada skenario pertama, generator dapat menghasilkan citra seperti terlihat pada Gambar 7.



Gambar 7. Perbandingan hasil *training* dan *testing* pada setiap *epoch*. *Epoch* yang diujicoba adalah 5, 10, 15, dan 20.

Pada Gambar 7 dapat dilihat bahwa citra dengan ukuran 256x256x3 menghasilkan citra wajah domain B lebih baik daripada citra dengan ukuran 128x128. Citra berukuran 128x128x3 masih terlihat tidak realistis dengan adanya bayangan pada citra. Hal ini dikarenakan citra ukuran 256x256x3 memiliki detail piksel yang lebih banyak, sehingga citra yang dihasilkan cukup detail. Meskipun untuk melatih model menggunakan citra 256x256x3 harus menggunakan memori yang cukup besar, namun hasil citra dengan ukuran 256x256x3 terlihat lebih realistis.

Di sisi lain, *learning rate* (lr) pada proses pelatihan berfungsi untuk mengatur seberapa cepat model belajar. Semakin besar nilai *learning rate*, maka semakin cepat model akan belajar untuk menghasilkan citra.

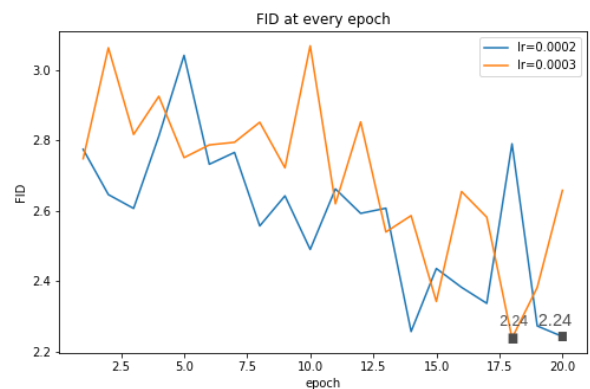
Untuk skenario 2, citra yang digunakan yaitu citra berukuran 256x256x3. *Learning rate* yang pertama diujikan yaitu lr = 0,0002 sesuai dengan pengaturan pada (Zhu dkk., 2017). Dalam skenario ini, lr = 0,0003 juga diujikan untuk melihat perkembangan model dalam belajar. Pada Tabel 1 dapat dilihat bahwa nilai FID terkecil terdapat pada lr = 0,0003 saat epoch ke-18 yaitu sebesar 2,24. Sedangkan pada lr = 0,0002 model baru mencapai nilai FID terkecil pada epoch ke-20. Dengan memperbesar nilai *learning rate*, model dapat menghasilkan citra dengan kualitas yang baik dengan *epoch* lebih sedikit. Pada Gambar 8 dilampirkan diagram perbandingan FID untuk setiap *epoch* dengan *learning rate* yang berbeda.

Berdasarkan hasil pengujian menggunakan data uji, sistem berhasil menghasilkan citra dengan nilai FID 2,24. Dari sampel citra pada Gambar 9 dapat dilihat bahwa masih terdapat sedikit artefak pada citra. Pada citra di bagian baris terbawah di Gambar

9 terdapat artefak seperti kumis di bawah hidung yang mana pada citra masukan sebelumnya tidak memiliki kumis. Selain itu generator juga dapat menghasilkan citra dengan warna rambut pada citra hasil rekonstruksi terlihat lebih pudar dan memutih sebagai efek penuaan pada domain B.

Tabel 1. Perbandingan nilai FID di setiap epoch pada *learning rate* (lr) yang berbeda. Nilai FID terkecil dengan lr = 0,0002 didapatkan pada *epoch* 20, sedangkan nilai FID terkecil dengan lr = 0,0003 didapatkan pada *epoch* 18.

Epoch	lr= 0,0002	lr= 0,0003
1	2,77	2,75
2	2,65	3,06
3	2,61	2,82
4	2,81	2,93
5	3,05	2,75
6	2,73	2,79
7	2,76	2,79
8	2,56	2,85
9	2,65	2,72
10	2,49	3,07
11	2,66	2,62
12	2,59	2,85
13	2,61	2,54
14	2,25	2,59
15	2,44	2,34
16	2,38	2,65
17	2,34	2,58
18	2,79	2,24
19	2,28	2,38
20	2,24	2,66

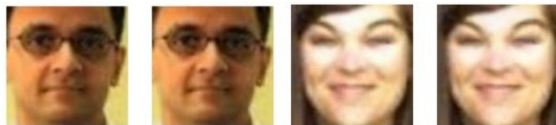


Gambar 8. Perbandingan FID untuk setiap *epoch* dengan *learning rate* yang berbeda



Gambar 9. Hasil citra rekonstruksi generator

Pada hasil citra rekonstruksi, terdapat beberapa citra yang gagal dipetakan oleh model ke target citra. Contoh citra yang gagal di translasi dapat dilihat pada Gambar 10 di mana citra tidak mengalami perubahan apapun. Setelah diobservasi semua citra wajah yang memiliki atribut lain seperti kacamata dan kualitas foto buram pada citra wajah gagal ditranslasikan oleh model.



Gambar 10. Contoh citra yang gagal ditranslasi ke domain target

4. KESIMPULAN

Dari proses pelatihan dan pengujian yang dilakukan, dapat diambil kesimpulan bahwa CycleGAN dapat digunakan untuk membuat sistem yang dapat melakukan prediksi penuaan wajah dengan menggunakan *unpaired data*. Namun untuk beberapa kondisi citra, CycleGAN belum bisa mentranslasikan citra sesuai target domain. Model generator dengan input citra 256x256x3 dan learning rate=0,0003 menghasilkan citra dengan efek penuaan yang lebih realistis dan nilai FID sebesar 2,24.

Saran untuk penelitian selanjutnya yaitu untuk melatih model dengan lebih banyak data dengan kondisi berbeda dan *epoch* yang lebih banyak agar generator lebih terlatih dalam menghasilkan citra yang sesuai dengan domain target. Penuaan wajah pada jenis kelamin yang berbeda memiliki efek penuaan yang berbeda pula, sehingga untuk penelitian selanjutnya dapat dicoba dengan memisahkan jenis kelamin pada data yang ada.

DAFTAR PUSTAKA

- ANON. 2023. *How digital de-aging is changing the face of movies | CBC News*. [online] Available at: <<https://www.cbc.ca/news/entertainment/digital-de-aging-1.5397657>> [Accessed 7 September 2023].
- ANTIPOV, G., BACCOUCHE, M. & DUGELAY, J.L., 2017. Face Aging With Conditional Generative Adversarial Networks. *Proceedings - International Conference on Image Processing, ICIP*, [online] 2017-September, pp.2089–2093. <https://doi.org/10.48550/arxiv.1702.01983>
- BORJI, A., 2022. Pros and cons of GAN evaluation measures: New developments. *Computer Vision and Image Understanding*, 215, p.103329. <https://doi.org/10.1016/J.CVIU.2021.103329>.
- DESPOIS, J., FLAMENT, F. & PERROT, M., 2020. AgingMapGAN (AMGAN): High-Resolution Controllable Face Aging with Spatially-Aware Conditional GANs. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, [online] 12537 LNCS, pp.613–628. https://doi.org/10.1007/978-3-030-67070-2_37/FIGURES/12.
- FU, Y., GUO, G. & HUANG, T.S., 2010. Age synthesis and estimation via faces: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(11), pp.1955–1976. <https://doi.org/10.1109/TPAMI.2010.36>.
- GOODFELLOW, I.J., POUGET-ABADIE, J., MIRZA, M., XU, B., WARDE-FARLEY, D., OZAIR, S., COURVILLE, A. & BENGIO, Y., 2014. Generative Adversarial Nets. *Advances in Neural Information Processing Systems*, 27.
- HEUSEL, M., RAMSAUER, H., UNTERTHINER, T., NESSLER, B. & HOCHREITER, S., 2017. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. *Advances in Neural Information Processing Systems*, [online] 2017-December, pp.6627–6638. <https://doi.org/10.48550/arxiv.1706.08500>
- KEMELMACHER-SHLIZERMAN, I., SUWAJANAKORN, S. & SEITZ, S.M., 2014. Illumination-aware age progression. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp.3334–3341. <https://doi.org/10.1109/CVPR.2014.426>.
- KINGMA, D.P. & BA, J.L., 2014. Adam: A Method for Stochastic Optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*. [online] <https://doi.org/10.48550/arxiv.1412.6980>.
- KORGIALAS, C., PANTRAKI, E., BOLARI, A., SOTIROUDI, M. & KOTROPOULOS, C., 2023. Face Aging by Explainable

- Conditional Adversarial Autoencoders. *Journal of Imaging* 2023, Vol. 9, Page 96, [online] 9(5), p.96. <https://doi.org/10.3390/JIMAGING9050096>.
- LIU, S., SUN, Y., ZHU, D., BAO, R., WANG, W., SHU, X. & YAN, S., 2017. Face aging with contextual generative adversarial nets. *MM 2017 - Proceedings of the 2017 ACM Multimedia Conference*, [online] pp.82–90. <https://doi.org/10.1145/3123266.3123431>.
- RADFORD, A., METZ, L. & CHINTALA, S., 2015. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*. [online] <https://doi.org/10.48550/arxiv.1511.06434>
- REED, S., AKATA, Z., YAN, X., LOGESWARAN, L., SCHIELE, B. & LEE, H., 2016. Generative Adversarial Text to Image Synthesis. *33rd International Conference on Machine Learning, ICML 2016*, [online] 3, pp.1681–1690. <https://doi.org/10.48550/arxiv.1605.05396>
- REN, L., LIU, S., SUN, Y., DONG, J., LIU, L. & YAN, S., 2017. Time traveler: A real-time face aging system. *MM 2017 - Proceedings of the 2017 ACM Multimedia Conference*, [online] pp.1245–1246. <https://doi.org/10.1145/3123266.3127922>.
- RONNEBERGER, O., FISCHER, P. & BROX, T., 2015. U-net: Convolutional networks for biomedical image segmentation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, [online] 9351, pp.234–241. https://doi.org/10.1007/978-3-24574-4_28/COVER.
- SAWANT, M.M. & BHURCHANDI, K.M., 2019. Age invariant face recognition. *Artificial Intelligence Review*, [online] 52(2), pp.981–1008. <https://doi.org/10.1007/S10462-018-9661-Z>.
- SHARMA, N., SHARMA, R. & JINDAL, N., 2022. Comparative analysis of CycleGAN and AttentionGAN on face aging application. *Sadhana - Academy Proceedings in Engineering Sciences*, [online] 47(1), pp.1–20. <https://doi.org/10.1007/S12046-01807-4/FIGURES/15>.
- SHENG, M., MA, Z., JIA, H., MAO, Q. & DONG, M., 2020. Face Aging with Conditional Generative Adversarial Network Guided by Ranking-CNN. *Proceedings - 3rd International Conference on Multimedia Information Processing and Retrieval, MIPR 2020*, pp.314–319. <https://doi.org/10.1109/MIPR49039.2020.00071>.
- SHU, X., TANG, J., LAI, H., LIU, L. & YAN, S., 2015. Personalized Age Progression with Aging Dictionary. In: *2015 IEEE International Conference on Computer Vision (ICCV)*. IEEE. pp.3970–3978. <https://doi.org/10.1109/ICCV.2015.452>.
- SUN, H., XI, Q., FAN, R., -, AL, CHEN, X., YANG, B., LI, J., JIA, P., HUANG, Y., CAI, B., HAMZAH, N. & HAFIZHELMI KAMARU ZAMAN, F., 2020. Face Aging on Realistic Photo in Cross-Dataset Implementation. *IOP Conference Series: Materials Science and Engineering*, [online] 917(1), p.012080. <https://doi.org/10.1088/1757899X/917/1/012080>.
- TAIGMAN, Y., POLYAK, A. & WOLF, L., 2016. Unsupervised Cross-Domain Image Generation. *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings*. [online] <https://doi.org/10.48550/arxiv.1611.02200>
- ZHANG, Z., SONG, Y. & QI, H., 2017. Age Progression/Regression by Conditional Adversarial Autoencoder. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, [online] 2017-January, pp.4352–4360. <https://doi.org/10.48550/arxiv.1702.08423>
- ZHU, J.Y., KRÄHENBÜHL, P., SHECHTMAN, E. & EFROS, A.A., 2016. Generative visual manipulation on the natural image manifold. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, [online] 9909 LNCS, pp.597–613. https://doi.org/10.1007/978-3-319-46454-1_36/TABLES/1.
- ZHU, J.Y., PARK, T., ISOLA, P. & EFROS, A.A., 2017. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. *Proceedings of the IEEE International Conference on Computer Vision*, [online] 2017-October, pp.2242–2251. <https://doi.org/10.48550/arxiv.1703.10593>

Halaman ini sengaja dikosongkan