



Implementasi Generative Adversarial Networks (GANs) Sistem Presensi Berbasis Deteksi Wajah (SIDEWA)

Implementation of Generative Adversarial Networks (GANs) in Face Detection-Based Attendance System (SIDEWA)

Tati Suprapti¹, Dian Ade Kurnia², Doni Anggara³, Rananda Deva Rian⁴, Aldi Setiawan⁵

^{1,3,4,5}Teknik Informatika, STMIK IKMI Cirebon

²Manajemen Informatika, STMIK IKMI Cirebon

¹tatisuprapti112004@gmail.com, ²dianade2014@gmail.com, ³titanese.dot@gmail.com, ⁴ranandapssr@gmail.com,

⁵marionata56@gmail.com

Abstract

This study aims to measure the success rate of the Generative Adversarial Network (GAN) algorithm model on facial image objects implemented in a face detection-based attendance system at the BPS Office in Cirebon City. GAN consists of two separate networks, between which are targeted at each other. The first set of tissues is a classifier that needs to be trained to know if it is real or false, and the second set of networks is a generator that generates a random sample similar to a real sample and uses it as a fake sample. GAN is a deep learning technique used to process unstructured data, one of which is facial images. Some of the stages in conducting this research include dataset preparation, data set processing, making generator models, making discriminator models, combining generator and discriminator models, making a GAN training process and analyzing the capabilities of generators and discriminators. Based on the results of experiments that have been carried out on facial image samples through 1000 epochs with 10 iterations every 1 epoch, it takes 5 minutes of training time, by producing an average accuracy of 66.06%. From the results of the training process carried out, the images obtained can be said to be successful because they are visible even though they are not yet very clearly visible.

Keywords : generative adversarial networks, deep learning, epoch, face

Abstrak

Penelitian ini bertujuan untuk mengukur tingkat keberhasilan model algoritma Generative Adversarial Networks (GANs) terhadap objek gambar wajah yang diimplementasikan pada sistem presensi berbasis deteksi wajah di Kantor BPS Kota Cirebon. GANs terdiri dari dua jaringan terpisah, di antaranya ditargetkan satu sama lain. Kumpulan jaringan pertama adalah pengklasifikasi yang perlu dilatih untuk mengetahui apakah itu nyata atau salah, dan kumpulan jaringan kedua adalah generator yang menghasilkan sampel acak yang mirip dengan sampel nyata dan menggunakannya sebagai sampel palsu. GANs merupakan teknik deep learning yang digunakan untuk memproses data yang tidak terstruktur salah satunya gambar wajah. Tahapan dari teknik GANs ini terdiri dari penyiapan dataset, pemrosesan data set, membuat model generator, membuat model diskriminator, menggabungkan model generator dan diskriminator, membuat proses training GANs dan menganalisis kemampuan generator dan diskriminator. Berdasarkan hasil eksperimen yang telah dilakukan pada sampel gambar wajah melalui 1000 epoch dengan 10 iterations setiap 1 epoch diperlukan waktu training selama 5 menit, dengan menghasilkan rata rata akurasi 72, %. Dari hasil dari proses training yang dilakukan, gambar yang diperoleh dapat dikatakan berhasil karena terlihat walaupun belum nampak dengan sangat jelas.

Kata Kunci : generative adversarial networks, deep learning, epoch, face

1. Pendahuluan

Situasi Covid-19 memiliki dampak selain secara langsung kepada kesehatan manusia, juga mengubah perilaku hidup sehat dan pola bekerja yang memerlukan perilaku menjaga jarak (social distancing).

Semaksimal mungkin menghindari dengan sentuhan benda baik secara langsung maupun tidak langsung. Berdasar dari kondisi itulah Badan Pusat Statistik (BPS) Kota Cirebon melakukan inovasi dengan membangun sistem presensi pegawai yang semula berbasis Finger-

print menjadi sistem presensi berbasis deteksi wajah (SIDEWA). Perubahan sistem tersebut selain didasari dengan tuntutan protokol kesehatan dimasa covid-19 yang harus melakukan social distance, juga didasari karena sistem sebelumnya mengalami kendala diantaranya sulitnya pimpinan dalam memantau kehadiran secara real-time, seringnya kesalahan dalam merekap data kehadiran pegawai dan lain sebagainya. Pengembangan SIDEWA menjawab terhadap tuntutan protokol kesehatan dimasa pandemik Covid-19 yaitu menjaga jarak (social-distance), karena pegawai tidak perlu menyentuh benda apapun, hanya dengan cara menghadapkan muka ke kamera yang telah dipasang, selanjutnya dideteksi wajahnya oleh aplikasi, kemudian dicocokkan dengan database wajah yang telah disimpan sebelumnya dan selanjutnya di rekam sebagai data kehadiran. SIDEWA telah berjalan kurang lebih 2 tahun sejak tahun 2020 sampai dengan sekarang. Namun demikian, masih terdapat kendala dari sisi tingkat akurasi yang masih belum optimal, performance mesin yang kurang maksimal. Hal ini diakibatkan karena keterbatasan mesin komputer yang belum mampu untuk melakukan proses algoritma yang lebih canggih.

Saat ini banyak teknik machine learning yang digunakan untuk menghasilkan performance yang lebih baik, sekalipun data yang diprosesnya adalah data yang tidak terstruktur [1]. Bahkan, teknik pembelajaran yang paling dalam atau disebut dengan teknik *deep learning* saat ini sudah mulai diaplikasi diberbagai sektor kehidupan, karena seiring dengan ukuran data dan jenis data yang berkembang yang disebut dengan *big data* [2].

Teknik *deep learning* yang digunakan untuk memproses *big data* salah satunya adalah *Generative Adversarial Network (GAN)*. GAN merupakan teknik machine learning yang ditemukan oleh Ian Goodfellow pada tahun 2014 [3]. GAN terdiri dari dua sistem, yaitu sistem generatif dan sistem diskriminatif. Sistem generatif menghasilkan pesaing atau tandingan sedangkan sistem diskriminatif menilai mereka [4]. Tantangannya adalah sejauh mana informasi tersebut menyebar. Secara teratur, sistem generatif mencari cara untuk menghasilkan informasi yang menarik berdasarkan kode objek, sedangkan diskriminatif mengenali output yang dikirim oleh generator dari informasi asli [5].

Penelitian lain telah membuktikan bahwa GAN digunakan untuk menghasilkan kajian tentang model topeng penghilang wajah, yang terdiri dari generator dekode yang disandikan dan jaringan tekstur, untuk mensintesis bagian wajah yang hilang [6] [7]. GANs juga berhasil diimplementasikan pada penelitian yang berfokus pada usia berdasarkan wajah, dengan dibuktikan bahwa efektif dalam mempelajari transformasi antara kelompok usia dengan bantuan

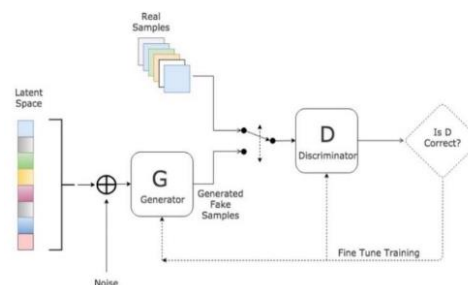
sample berpasangan yakni melalui perbedaan dari setiap gambar wajah yang sama [6].

Kontribusi dari penelitian ini adalah pertama menentukan tugas yang tepat untuk permasalahan big data, kedua menerapkan model GANs terhadap sistem deteksi wajah pada presensi pegawai dilingkungan Kantor Statistik Kota Cirebon, ketiga mengevaluasi kinerja model GANs melalui pengukuran tingkat akurasi generatif dan akurasi diskriminatif..

2. Metode Penelitian

2.1. Arsitektur Generative Adversarial Network (GAN)

GAN terdiri dari dua jaringan terpisah, di antaranya ditargetkan satu sama lain. Kumpulan jaringan pertama adalah pengklasifikasi yang perlu kita latih untuk mengetahui apakah itu nyata atau salah, dan kumpulan jaringan kedua adalah generator yang menghasilkan sampel acak yang mirip dengan sampel nyata dan menggunakannya sebagai sampel palsu [5].



Gambar 1. Cara Kerja Generative Adversarial Network (GAN)

Pada gambar 1 dijelaskan perihal D sebagai pengklasifikasi gambar, untuk rangkaian gambar untuk membedakan wajah yang berbeda [8]. Tujuan Generator G adalah untuk menggambar gambar palsu yang sangat dekat untuk mengelabui D dengan memilih elemen di ruang potensial dari data *training* untuk komposisi dan menambahkan noise acak sebagai data palsu. Selama *training*, D menerima data asli dan data palsu yang dihasilkan G, dan tugasnya adalah menentukan apakah gambar itu milik data asli atau data palsu.

2.2. Formulasi GAN

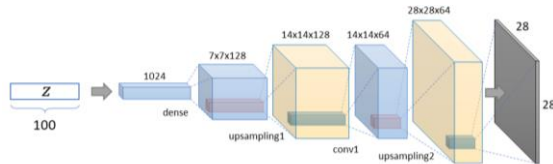
Setiap model Deep learning sebuah fungsi harus didefinisikan untuk memperoleh suatu hasil [9]. Pada Generatif Adversarial Networks memilih suatu Fungsi Objective yang digunakan bersamaan dengan model yang dibangun, dengan rumus persamaan 1 [10] :

$$\begin{array}{lll} \text{Min} & \text{Max} & \text{Ex} \sim p_x[\log D(x)] + \\ \text{G} & \text{D} & \text{Ez} \sim p_z[\log(1-D(G(z)))] \end{array} \quad (1)$$

Fungsi objektif (1) terdiri dari x sebagai real data (Sampel data) dan z adalah noise vektor, G adalah Generator dan D adalah Diskriminator, p_x adalah distribusi dari x (real data) dan p_z adalah masing-masing dari noise vektor. $D(x)$ merupakan hasil dari training Diskriminator, $G(z)$ adalah hasil dari Generator dan $D(G(z))$ adalah Hasil dari Discriminator yang memberikan nilai/hasil dari $G(z)$. Diskriminator dilatih untuk memaksimalkan pemberian nilai terhadap hasil dari generator dengan memaksimalkan fungsi $\log D(x) + \log(1 - D(G(z)))$. Kemudian Generator dilatih untuk dapat menipu Diskriminator dengan meminimalkan fungsi $\log(1 - D(G(z)))$. Melalui pelatihan inilah Generator akan berusaha semaksimal mungkin untuk menghasilkan gambar tidak semirip mungkin dengan dataset (real data), dan Diskriminator juga akan berusaha untuk mengidentifikasi hasil dari Generator untuk mengetahui gambar tersebut asli atau palsu.

2.3. Model Jaringan Generator

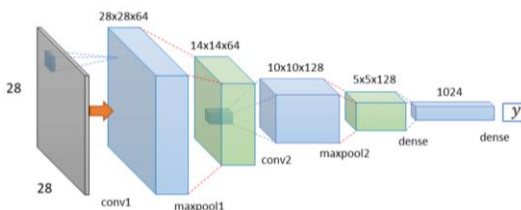
Model jaringan generator terdiri dari input generator dimana ada sebuah array yang dibangkitkan secara random yang disebut dengan *seed*. Disini akan ditetapkan untuk ukuran input seed adalah $[1 \times 100]$. Dari masukan seed tersebut, Generator akan mengubahnya menjadi sebuah gambar berukuran $[32 \times 32]$ menggunakan Convolutional Neural Network. Secara detil, model jaringan genator ini dapat digambarkan seperti gambar 2:



Gambar 2. Arsitektur Generator

2.4. Model Jaringan Diskriminator

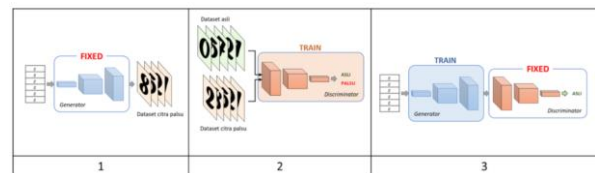
Dikriminator merupakan CNN yang menerima input image berukuran $[28, 28]$ dan menghasilkan angka biner yang menyatakan apakah input gambar merupakan gambar dari dataset asli (kelas 1) atau merupakan gambar baru/gambar palsu (kelas 0) [9]. Sebagai ilustrasi, model jaringan diskriminator dapat digambarkan pada gambar 3:



Gambar 3. Arsitektur Diskriminator

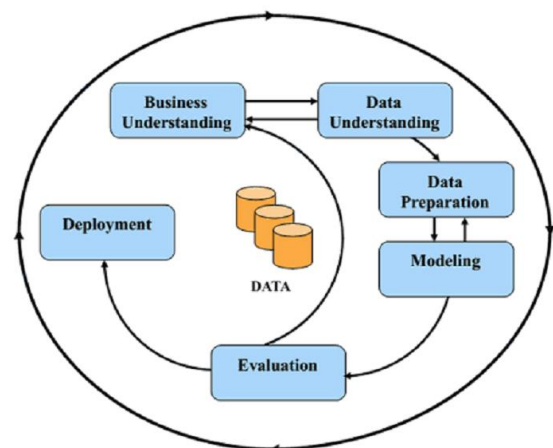
2.5. Proses Pelatihan GAN

Pada proses pelatihan (training) ini dilakukan secara berurutan, untuk setiap epoch pelatihan akan melakukan tiga tahapan yaitu membangkitkan dari citra palsu menggunakan generator sejumlah dataset citra asli [10], melatih discriminator untuk bisa membedakan dataset citra asli dari dataset citra palsu, menggunakan discriminator yang sudah dilatih untuk melatih Generator agar bisa membangkitkan dataset citra palsu yang dinilai asli oleh discriminator. Tahapan tersebut dapat digambarkan pada gambar 4.



Gambar 4. Tahapan Proses Pelatihan (Training) GAN

Metode yang digunakan adalah Cross Industri Standar Process Data Mining (CRISP-DM) dengan tahapan antara lain *Business Understanding*, *Data Understanding*, *Data Preparation*, *Modelling*, *Evaluation* dan *Deployment* [7]. Tahapan tersebut dapat ilustrasikan pada gambar 5.



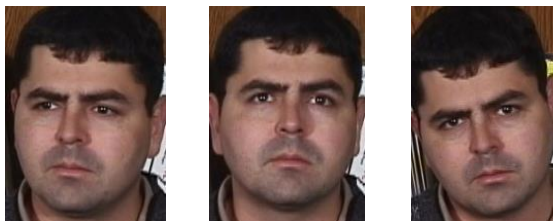
Gambar 5. Diagram Proses CRISP-DM
 Sumber : Pete Chapman et. Al (2000)

Penjelasan dari masing-masing tahapan gambar 5 dapat dijabarkan: *Business Understanding*, pada tahapan ini harus dilakukan tentang menentukan tujuan diimplementasikannya GANs dalam rangka penyempurnaan Sistem Presensi Berbasis Deteksi Wajah khususnya dilingkungan Kantor Badan Pusat Statistik (BPS) Kota Cirebon, sehingga dapat diperoleh rekaman informasi yang diketahui dari sebuah Kantor BPS tersebut. *Data Understanding*, pada tahapan ini

dilakukan aktivitas pengumpulan data pegawai yang ada dilingkungan Kantor BPS Kota Cirebon dengan memahami atribut yang akan dijadikan masukan ke dalam sistem, setelah itu data dideskripsikan dan dieksplorasi sehingga dapat diperoleh data yang valid dan data yang berkualitas. *Data Preparation*, pada tahapan ini dilakukan deskripsi dataset dari atribut pegawai yang ada dilingkungan kantor tersebut, berdasarkan foto wajah yang diambil dari kamera sehingga diharapkan dataset tersebut telah siap untuk digunakan pada pembuatan model. Dataset tersebut harus melalui tahapan seleksi data, pembersihan data, konstruksi data dan integritas data serta format data. *Modeling*, tahapan ini memilih teknik GANs yang tepat dari sekian teknik deep learning lainnya selanjutnya mendesain model GANs dan mengujinya untuk dapat digunakan seoptimal mungkin. *Evaluation*, dilakukan aktivitas mengevaluasi hasil model GANs, mereview proses model GANs. *Deployment*, yaitu aktivitas membangun aplikasi GANs untuk implementasi SIDEWA [11]

2.6. Dataset

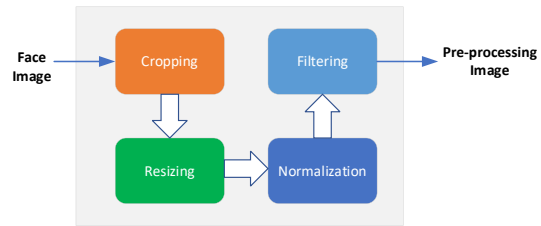
Dataset diambil dari bagian umum yang menangani kepegawaian berupa file foto pegawai di Kantor BPS Kota Cirebon. Petugas harus mengambil foto wajah melalui kamera yang telah disediakan. Setiap pegawai di capture sebanyak 3 pose, yakni pose tegak menghadap kamera, pose miring ke kiri dan pose miring ke kanan dengan kecukupan pencahayaan. Sebagai data sampel akan diambil 6 orang data foto pegawai dengan masing-masing pegawai 3 pose tersebut, sehingga total sampelnya sebanyak 18 foto.



Gambar 6. Contoh Sampel Citra Wajah 3 pose

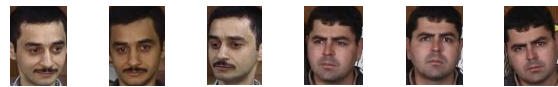
2.7. Image Pre-Processing

Tahapan pada image pre-processing bertujuan sebagai bentuk pengolahan atau pemrosesan sinyal dengan input berupa gambar (image) dan ditransformasikan menjadi gambar lain [12] dengan menggunakan teknik tertentu. Beberapa teknik pengolahan citra yang digunakan dalam penelitian ini adalah *Cropping*, *Resizing*, *Normalization*, *Filtering* seperti gambar 7.



Gambar 7 Tahapan Pemrosesan Gambar (Citra)

Teknik *cropping* adalah proses penghapusan bagian sudut dari gambar untuk memotong /mengambil / mengeluarkan sebagian isi gambar guna memperoleh hasil yang diinginkan [13]. Hasil dari teknik ini seperti tampil pada gambar 8 citra wajah.



Gambar 8 Teknik Cropping

Teknik *resizing* merupakan proses mengubah ukuran besar gambar dalam satuan piksel [14]. Ukuran piksel yang ditentukan untuk seluruh citra yaitu dengan dimensi 32 x 32 , bit depth 24 dengan hasil gambar 9.



Gambar 9 Citra berdimensi 32 x 32 dengan bit depth 24 RGB

Teknik *normalization* merupakan proses distribusi intensitas pada gambar yang dipengaruhi oleh pencahayaan yang tidak sama. Teknik *filtering* merupakan proses untuk menghilangkan derau (noise) pada gambar [15] . Teknik filtering disini menggunakan teknik pengubahan dari RGB ke Black & White, dengan hasil pada gambar 10.



Gambar 10 Teknik Filtering Citra RGB ke Gray

Sebagai tools untuk melakukan teknik diatas, digunakan tools ilove image yang dapat diakses secara online melalui alamat url <https://www.iloveimg.com/>

3. Hasil dan Pembahasan

Berdasarkan hasil eksperimen yang telah dilakukan, beberapa hasil telah diperoleh diantaranya hasil teknik pre-processing seperti cropping, resizing, normalisasi dan filtering. Selain teknik pre-processing juga diperoleh hasil pemodelan generator, pemodelan diskriminator serta pemodelan gabungan antara generator dan diskriminator.

Hasil dari model diskriminator meliputi layer flatten, dense, leaky_re_lu dan dense 1, seperti yang digambarkan pada tabel 1.

Tabel 1. Hasil Model Diskriminator

Layer (type)	Output Shape	Param #
flatten (Flatten)	(None, 2352)	0
dense (Dense)	(None, 128)	301184
leaky_re_lu (LeakyReLU)	(None, 128)	0
dense_1 (Dense)	(None, 1)	129
Total params: 301,313 Trainable params: 301,313 Non-trainable params: 0		

Hasil dari model generator meliputi dense_2, leaky_re_lu_1, dense_3 dan reshape seperti pada tabel 2.

Tabel 2. Hasil Model Generator

Layer (type)	Output Shape	Param #
dense_2 (Dense)	(None, 128)	12928
leaky_re_lu_1 (LeakyReLU)	(None, 128)	0
dense_3 (Dense)	(None, 2352)	303408
reshape (Reshape)	(None, 28, 28, 3)	0
Total params: 316,336 Trainable params: 316,336 Non-trainable params: 0		

Sedangkan hasil model gabungan antara model generator dan diskriminator dapat ditampilkan seperti tabel 3.

Tabel 3 : Hasil Model Gabungan (Generator dan Diskriminator)

Layer (type)	Output Shape	Param #
sequential_1 (Sequential)	(None, 28, 28, 3)	316336
sequential (Sequential)	(None, 1)	301313
Total params: 617,649 Trainable params: 316,336 Non-trainable params: 301,313		

Model generator dan model diskriminator telah berhasil dibuat dan digabungkan sehingga telah menjadi satu kolom atau vektor tunggal yang disebut dengan. Selanjutnya dihasilkan fungsi aktivasi yang berfungsi untuk menampung berbagai jenis jaringan syaraf untu memudahkan proses training dan menghasilkan kinerja model lebih baik.

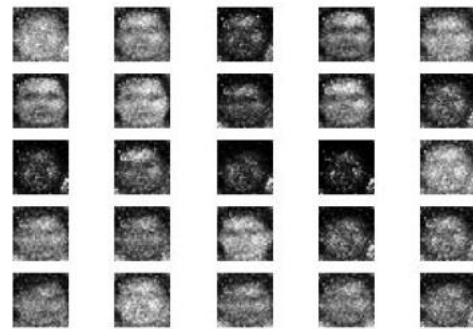
Setelah dibuat model generator dan model diskriminator dan model gabungan antara generator dan diskriminator, selanjutnya akan diukur tingkat *generator loss* dan *diskriminator loss* serta tingkat akurasi terhadap salah satu sample citra melalui proses training yang ditetapkan dengan nilai parameter epoch 1000, batch size 128 dan interval 100. Dari hasil proses training dapat diperoleh tabel 4.

Tabel 4. Akurasi Diskriminator dan Generator dengan 1000 epoch

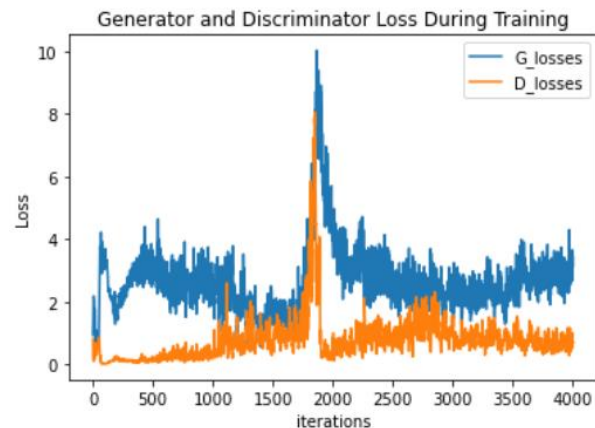
Literasi	Epoch	D Loss	Accuracy	G Loss
1	100	0.996641	50.78	1.92073
2	200	1.077202	56.25	1.924441

Literasi	Epoch	D Loss	Accuracy	G Loss
3	300	1.516468	43.36	2.371171
4	400	0.466807	78.52	2.622881
5	500	0.828703	58.98	2.219804
6	600	0.407905	79.69	2.864509
7	700	0.669072	73.44	2.802930
8	800	0.657398	76.17	2.93177
9	900	0.546918	74.22	3.047591
10	1000	0.691887	72.27	3.147644

Dari tabel 4 diatas dapat divisualisasikan ke dalam bentuk Grafik Generator dan Diskriminator Loss selama proses training sebagaimana tampilan gambar 11.



Gambar 11. Hasil Training sebanyak 1000 epoch



Gambar 12. Training Generator Loss dan Diskriminator Loss

Dengan training 1000 epoch diperoleh hasil seperti yang terdapat pada gambar diatas, hal ini masih kurang memuaskan jika dilihat, karena hasil setiap gambar tergolong sama. Namun jika dilihat dari visualisasi grafik diskriminator dan generator loss seperti dilihat pada gambar tersebut sebenarnya sudah baik karena generator berhasil menghasilkan gambar yang sangat berbeda dengan datasetnya sehingga diskriminator sulit untuk mengenali gambar yang dihasilkan generator, tetapi untuk gambar yang dihasilkan generator terlihat kurang baik atau kurang jelas. Waktu yang dibutuhkan untuk melatih model dengan 1000 epoch adalah sekitar 5 menit.

4. Kesimpulan

Implementasi algoritma deeplearning dengan teknik Generative Adversarial Networks (GANs) telah berhasil dilakukan untuk dapat dimanfaatkan pada aplikasi SDEWA dengan tingkat akurasi pengenalan wajah sebesar 72.27%. Penelitian ini belum dikatakan maksimal karena masih terlihat gambar yang belum nampak dengan sangat jelas. Hal ini dipengaruhi oleh rendahnya resolusi atau pixel dari gambar yang dihasilkan disebabkan dimensi gambar yang didefinisikan diawal hanya 32 pixel. Oleh karena itu, penetapan nilai 1000 epoch dengan 10 iterations setiap 1 epoch sudah cukup baik untuk melatih model yang saat ini digunakan. Untuk meningkatkan pelatihan model ini dimasa yang akan datang, peningkatan ukuran konvolusi dan dekonvolusi dari 512 x 512 menjadi 1024 x 1024 akan sangat berguna untuk menghasilkan gambar yang lebih baik.

Masa yang akan datang, model ini tidak hanya diimplementasikan pada sistem presensi bidang kepegawaian saja, tentu saja dapat digunakan dibidang pendidikan seperti presensi siswa, dibidang hukum dan kriminalitas dan bidang lainnya yang relevan.

Ucapan Terimakasih

Terimakasih kepada Dirjen Pendidikan Tinggi Kemendikbud Ristek yang telah membiayai terhadap publikasi melalui Program Kompetisi Hibah Penelitian Dosen Pemula tahun anggaran 2022.

Daftar Rujukan

- [1] S. Guan and M. Loew, "Evaluation of Generative Adversarial Network Performance Based on Direct Analysis of Generated Images," *Proc. - Appl. Imag. Pattern Recognit. Work.*, vol. 2019-Octob, 2019, doi: 10.1109/AIPR47015.2019.9174595.
- [2] R. Mawan, "Klasifikasi motif batik menggunakan convolutional neural network," *J. Open Access Yayasan Lentera Dua Indones.*, vol. 1, pp. 45–50, 2020, doi: 10.36802/janaloka.
- [3] T. Mukhiddin, W. Lee, S. Lee, and T. Rashid, "Research issues on generative adversarial networks and applications," *Proc. - 2020 IEEE Int. Conf. Big Data Smart Comput. BigComp 2020*, pp. 487–488, 2020, doi: 10.1109/BigComp48618.2020.00-19.
- [4] S. E. Awan, M. Bennamoun, F. Sohel, F. Sanfilippo, and G. Dwivedi, "Imputation of missing data with class imbalance using conditional generative adversarial networks," *Neurocomputing*, vol. 453, pp. 164–171, 2021, doi: 10.1016/j.neucom.2021.04.010.
- [5] C. Min, Y. Li, L. Fang, and P. Chen, "Conditional Generative Adversarial Network on Semi-supervised Learning Task," *2019 IEEE 5th Int. Conf. Comput. Commun. ICCCC 2019*, pp. 1448–1452, 2019, doi: 10.1109/ICCC47050.2019.9064268.
- [6] Z. J. Yu and S. C. Yang, "Research on the application of jigsaw generative adversarial network to face generation," *Proc. - 2020 Int. Symp. Comput. Consum. Control. IS3C 2020*, pp. 26–29, 2020, doi: 10.1109/IS3C50286.2020.00014.
- [7] Prabhat, Nishant, and D. K. Vishwakarma, "Comparative Analysis of Deep Convolutional Generative Adversarial Network and Conditional Generative Adversarial Network using Hand Written Digits," *Proc. Int. Conf. Intell. Comput. Control Syst. ICICCS 2020*, no. Iciccs, pp. 1072–1075, 2020, doi: 10.1109/ICICCS48265.2020.9121178.
- [8] M. Toshpulatov, W. Lee, and S. Lee, "Generative adversarial networks and their application to 3D face generation: A survey," *Image Vis. Comput.*, vol. 108, p. 104119, 2021, doi: 10.1016/j.imavis.2021.104119.
- [9] S. A. Israel et al., "Generative adversarial networks for classification," *Proc. - Appl. Imag. Pattern Recognit. Work.*, vol. 2017-Octob, pp. 23–26, 2018, doi: 10.1109/AIPR.2017.8457952.
- [10] K. X. Chen, Z. X. Zhao, and S. Yamane, "Enhanced Conditions Based Deep Convolutional Generative Adversarial Networks," *2021 IEEE 10th Glob. Conf. Consum. Electron. GCCE 2021*, pp. 663–665, 2021, doi: 10.1109/GCCE53005.2021.9621858.
- [11] V. B. Raj and K. Hareesh, "Review on Generative Adversarial Networks," *Proc. 2020 IEEE Int. Conf. Commun. Signal Process. ICCSP 2020*, pp. 479–482, 2020, doi: 10.1109/ICCSP48568.2020.9182058.
- [12] K. Park, M. Chae, and J. H. Cho, "Image pre-processing method of machine learning for edge detection with image signal processor enhancement," *Micromachines*, vol. 12, no. 1, pp. 1–13, 2021, doi: 10.3390/mi12010073.
- [13] N. Petrellis, "A Review of Image Processing Techniques Common in Human and Plant Disease Diagnosis," *Symmetry (Basel)*, vol. 10, no. 7, p. 270, Jul. 2018, doi: 10.3390/sym10070270.
- [14] X. Yan, L. Wen, L. Gao, and M. Perez-Cisneros, "A Fast and Effective Image Preprocessing Method for Hot Round Steel Surface," *Math. Probl. Eng.*, vol. 2019, 2019, doi: 10.1155/2019/9457826.
- [15] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 6, pp. 1397–1409, 2013, doi: 10.1109/TPAMI.2012.213.