Jurnal Telematika vol. 20 no. 1 2025
e-ISSN: 2579-3772
doi: 10.61769/telematika.v20i1.730

Model *Deep Learning* untuk Face Anti-Spoofing dalam Mengatasi *Domain Generalization* dengan *Depth Estimation* dan *Generative Adversarial Network*

Tio Dewantho Sunoto[1], Daniel Setiadikarunia[2], Riko Arlando Saragih[3,*], Elia Moses[4]

[12,3,4]*Program Studi Teknik Elektro, Universitas Kristen Maranatha*
Jl. Suria Sumantri No. 65, Bandung, Indonesia

[1]tiodewantho@yahoo.com
[2]blessed_dsk@yahoo.com
[3]2122004@eng.maranatha.edu

*Korespondensi: riko.as@eng.maranatha.edu

Abstract*— The use of facial biometrics to gain access to a security system is common in communication/computing devices. However, this convenience comes with a vulnerability to security breaches, where facial images can be falsified using photos or videos of someone with access rights. The availability of photos or videos of individuals on social media can exacerbate this. A face anti-spoofing system (FAS) is a crucial component for determining whether an input image is genuine or synthetic in biometric systems that utilize facial image information. Many methods have been used to realise this system, both with a hand-crafted method-based approach and deep learning (DL). However, research on the distribution differences between the test dataset and the training dataset is still rare. This article discusses the use of deep learning (DL)-based models for face anti-spoofing (FAS) applications. This study employs a model that utilizes depth map estimation to identify discriminative features and a generative adversarial network (GAN) to address the issue of distribution differences through a data generation approach. For models implemented with an intra-set simulation scenario, test results for two public datasets, NUAA and CASIA, provided the best results in terms of half total error rate (HTER) metrics, at 2.97% and 2.7% respectively. Meanwhile, simulations comparing the characteristics of the test dataset and the training dataset revealed that applying GAN to enhance the model's generalization ability could reduce the bona fide presentation classification error rate (BPCER) by 9.75%.

***Keywords*— face anti-spoofing (FAS), deep learning (DL), generative adversarial neural network (GAN), domain generalization (DG), FAS dataset**

***Abstrak*— Penggunaan biometrik wajah untuk memperoleh akses suatu sistem keamanan adalah hal yang lazim ditemukan dalam perangkat komunikasi/komputasi. Walaupun demikian, kemudahan ini berakibat kepada kerentanan terjadinya penerobosan ke dalam sistem keamanan, di mana citra wajah dapat dipalsukan dengan memanfaatkan foto atau video seseorang yang memiliki hak akses. Hal ini dapat diperburuk dengan tersedianya foto atau video seseorang di media sosial. Sistem *face anti-spoofing* (FAS) adalah suatu sistem yang penting untuk mendeteksi apakah citra masukan adalah citra riil atau citra palsu dalam suatu sistem biometrik yang menggunakan informasi citra wajah. Banyak metode yang sudah digunakan untuk merealisasikan sistem ini, baik dengan pendekatan berbasis metode *hand-crafted* maupun *deep learning* (DL). Walaupun demikian, penelitian mengenai perbedaan distribusi antara *dataset* uji dengan *dataset* latih masih jarang dilakukan. Artikel ini membahas penggunaan model berbasis *deep learning* (DL) untuk aplikasi *face anti-spoofing* (FAS). Penelitian ini mengimplementasikan model menggunakan estimasi peta kedalaman untuk menemukan fitur diskriminatif dan *generative adversarial network* (GAN) untuk mengatasi isu perbedaan distribusi yang**

menggunakan pendekatan berupa pembangkitan (pembentukan) data. Untuk model yang diimplementasikan dengan skenario simulasi *intraset*, hasil pengujian untuk dua *dataset* publik, yaitu NUAA dan CASIA, memberikan hasil terbaik dari segi metrik *half total error rate* (HTER), berturut-turut 2,97% dan 2,7%. Sementara simulasi untuk adanya perbedaan antara karakteristik *dataset* uji dengan dataset latih, hasil dengan menerapkan GAN untuk meningkatkan kemampuan generalisasi model, dapat menurunkan *bonafide presentation classification error rate* (BPCER) sebesar 9,75%.

***Kata Kunci*— *face anti-spoofing* (FAS), *deep learning* (DL), *generative adversarial neural network* (GAN), *domain generalization* (DG), *dataset* FAS**

I. PENDAHULUAN

Sistem pengenalan wajah (*face recognition*/FR) merupakan suatu teknik biometrik praktis yang telah banyak diterapkan dalam kehidupan sehari-hari [1]. Hal ini menyebabkan FR menjadi salah satu area penelitian yang paling populer, dikarenakan penerapannya yang luas dalam berbagai skenario *autentikasi*, seperti pengendalian akses *mobile* dan pembayaran elektronik [2].

Walaupun demikian, sistem FR rentan terhadap serangan presentasi yang juga dikenal sebagai *face spoofing attacks* (FAS), seperti *print attack*, *video replay*, dan *3D mask* yang mengancam keamanan sistem dan meningkatkan risiko pelanggaran privasi [1]. Untuk meningkatkan keandalan sistem FR, deteksi FAS menjadi bagian penting untuk melindungi keamanan sistem sehingga secara bertahap FAS menjadi fokus dalam bidang *computer vision*, pengenalan pola, dan bidang terkait lainnya [3].

Face anti-spoofing (FAS) adalah komponen yang sangat penting untuk memastikan keamanan sistem FR [4]. Secara umum FAS dapat diklasifikasikan menjadi dua kategori, yaitu metode tradisional berbasis *handcrafted* dan metode berbasis *deep learning* [5]. FAS dapat menggunakan fitur *hand-crafted* untuk klasifikasi, seperti LBP, HOG, dan SIFT untuk menggambarkan pola terkait *spoofing*. Fitur *hand-crafted* bekerja dengan baik pada sampel skala kecil, tetapi sensitif terhadap variasi pencahayaan, skala, dan postur [4]. Di sisi lain, dengan berkembangnya metode *deep learning*, jaringan mendalam mulai digunakan untuk deteksi FAS. Fitur-fitur yang diekstraksi melalui metode pembelajaran ini lebih *robust*. Oleh karena itu, CNN sebagai salah satu arsitektur neural network pada *deep learning* juga banyak digunakan dalam *face anti-spoofing* [6].

Seiring dengan perkembangan arsitektur CNN dan munculnya *dataset* FAS secara bertahap, metode *deep learning* telah menjadi dominan dalam bidang FAS. Metode FAS berbasis pembelajaran mendalam dapat diklasifikasikan menjadi tiga kategori, yaitu *classification supervision*, *auxiliary pixel-wise supervision*, dan *generative pixel-wise supervision* [7]. Metode berbasis klasifikasi biasanya menggunakan pengawasan entropi silang biner. Namun, model klasifikasi biner cenderung *overfitting* dan kurang *robust* terhadap serangan dalam skenario dengan pergeseran domain minor [7]. *Auxiliary pixel-wise supervision* dengan fungsi tambahan dengan mahir memanfaatkan pengetahuan manusia sebelumnya. Teknik ini memerlukan model yang dalam untuk memprediksi kedalaman asli untuk sampel hidup, sementara menghasilkan peta nol untuk yang *spoof*. Berbeda dengan *auxiliary pixel-wise supervision, generative pixel-wise supervision* tidak memberlakukan batasan tegas yang dirancang oleh para ahli. Secara umum, *generative pixel-wise supervision* memberikan fleksibilitas yang lebih dalam memodelkan *spoofing*, tetapi dapat mengakibatkan model kurang stabil dan lebih rentan terhadap gangguan dibandingkan dengan *auxiliary pixel-wise supervision* yang lebih terstruktur [7].

Terdapat berbagai penelitian yang telah dilakukan dengan menerapkan *auxiliary pixel-wise supervision* menggunakan CNN, khususnya yang menggunakan fitur kedalaman wajah. Yang, dkk. [8] pertama kali menerapkan CNN untuk FAS dan mencapai hasil yang baik, berdasarkan pengetahuan

sebelumnya, wajah asli memiliki kedalaman sementara wajah palsu hanyalah kertas atau layar datar. Atoum, dkk. [9] mengekstrak fitur area lokal dan memperkirakan peta kedalaman keseluruhan dari gambar wajah. Liu, dkk. [10] menggunakan analisis pantulan cahaya untuk mengekstrak detail normal guna memperkirakan kedalaman wajah. Penelitian-penelitian tersebut menunjukkan bahwa nilai kedalaman setiap objek dalam peta kedalaman wajah palsu adalah sama.

Namun, pendekatan berbasis CNN yang telah memperoleh hasil baik menghadapi tantangan lainnya, yaitu penurunan kinerja yang signifikan ketika model menghadapi domain yang belum pernah dilihat (*unseen domain)* sebelumnya. Model *machine learning* (ML) umumnya dilatih berdasarkan asumsi *independent and identically distributed* (i.i.d.)) bahwa data pelatihan dan pengujian didistribusikan secara identik dan independen. Namun, pada praktiknya asumsi ini tidak selalu berlaku. Ketika distribusi probabilitas data pelatihan dan data pengujian berbeda, kinerja model ML sering kali menurun karena adanya kesenjangan distribusi domain (*domain distribution gap*) [11].

Pengumpulan data dari semua domain yang mungkin untuk melatih model ML sangat mahal, bahkan hampir mustahil [12]. Oleh karena itu, meningkatkan kemampuan generalisasi model ML menjadi sangat penting.

Terdapat banyak topik penelitian yang terkait dengan *generalization*, seperti *domain adaptation*, *meta-learning*, *transfer learning*, dan *covariate shift*. Masing-masing dari topik tersebut memiliki karakteristiknya tersendiri [12], tetapi memiliki perbedaan utama apabila dibandingkan dengan *domain generalization* (DG), yaitu akses terhadapap target domain, DG tidak dapat melihat target domain saat pelatihan. Hal ini menyebabkan DG lebih menantang dibandingkan pembelajaran generalisasi lainnya, tetapi juga lebih realistis dan diinginkan dalam aplikasi praktis [12].

Dalam beberapa tahun terakhir, *domain generalization (DG)* telah mendapatkan banyak perhatian. Tujuan *domain generalization* adalah mempelajari model dari satu atau beberapa domain yang berbeda namun terkait yang akan dapat digeneralisasikan dengan baik pada domain pengujian yang belum pernah terlihat. Metode *domain generalization* (DG) untuk FAS diusulkan dengan tujuan untuk mempelajari model dari beberapa *dataset* sumber sehingga model tersebut dapat digeneralisasi ke *dataset* yang belum pernah dilihat sebelumnya.

Beberapa penelitian mengenai DG untuk fungsi FAS telah banyak dilakukan pada tahun-tahun belakangan ini. Liu, dkk. mengusulkan metode *source-free domain adaptation* (SFDA) untuk meningkatkan kemampuan generalisasi *dari deep learning-based face anti-spoofing (FAS*) [2]. Zheng, dkk. mengusulkan metode *masked frequency autoencoders* (MFAE) yang menyelidiki generalisasi domain FAS dari perspektif baru, yaitu dalam domain frekuensi [4]. Liu, dkk. mengusulkan domain generalisasi tanpa pengawasan (*unsupervised*) untuk *face anti-spoofing* [13]. Zhou, dkk. mengusulkan perspektif baru dalam *domain generalization* (DG) untuk *face anti-spoofing (FAS)* yang menyelaraskan representasi fitur pada tingkat instansi yang halus [14]. Beberapa metode *domain generalization* (DG) untuk fungsi FAS telah diusulkan dalam penelitian-penelitian terbaru untuk meningkatkan kemampuan generalisasi sistem FAS di berbagai domain yang belum pernah dilihat sebelumnya.

Berdasarkan uraian tersebut, penelitian ini akan berfokus pada permasalahan *face anti-spoofing* dengan memanfaatkan fitur *depth estimation* sebagai fitur diskriminatif wajah dan menerapkan model *generative adversarial network* (GAN) untuk mengatasi untuk mengakomodir perbedaan karakteristik (distribusi) antara data latih dan data uji (permasalahan DG). Pengujian dilakukan untuk empat dataset yang biasa dilakukan untuk menguji metode untuk mendeteksi FAS, yaitu NUAA, CASIA-MFSD, MSU-MFSD, dan IDIAP *Replay Attack*. Diharapkan dari pendekatan ini kesalahan prediksi pada sistem yang diimplementasikan dapat menurun (berkurang).

II. METODOLOGI

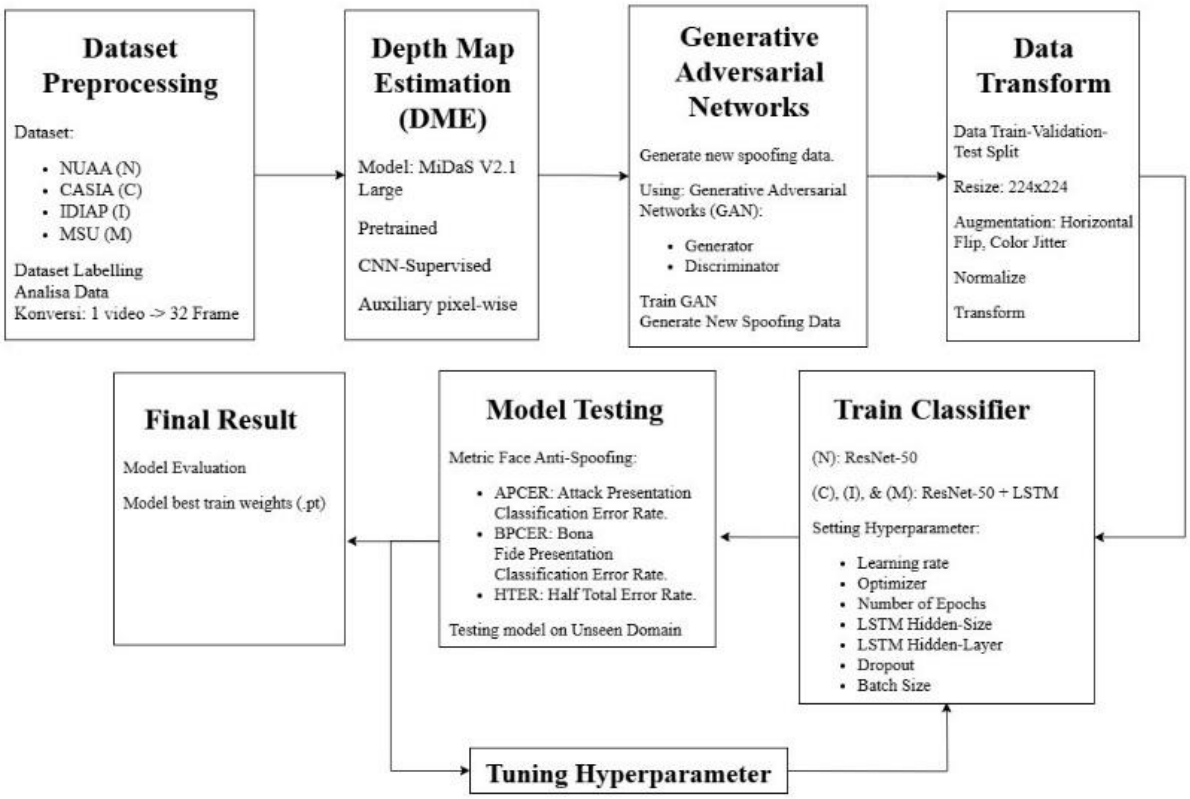
Secara ringkas alur dalam penelitian ini dapat dilihat pada Gambar 1. Tahap pertama alur penelitian diawali dengan *dataset preprocessing*. Proses utama tahapan ini *labelling*, khususnya untuk dataset NUAA dan dataset CASIA-MFSD. Proses ini dilakukan berdasarkan panduan dari *publisher dataset*.

Untuk *dataset* berbentuk video dilakukan *preprocessing* berupa *sampling frame*. Tiga dari empat *dataset* memiliki data dalam bentuk video, di antaranya CASIA-MFSD, Idiap Replay-Attacks, dan MSU-MFSD. Setiap video berdurasi kurang lebih 9 detik dengan total frame rata-rata adalah 240 *frame*. Pada penelitian ini setiap video diambil 32 *frame*. *Preprocessing* ini dilakukan untuk mempercepat proses komputasi model. Gambar 2 menunjukkan contoh proses *sampling* video dari video keseluruhan menjadi 32 *frame* gambar.

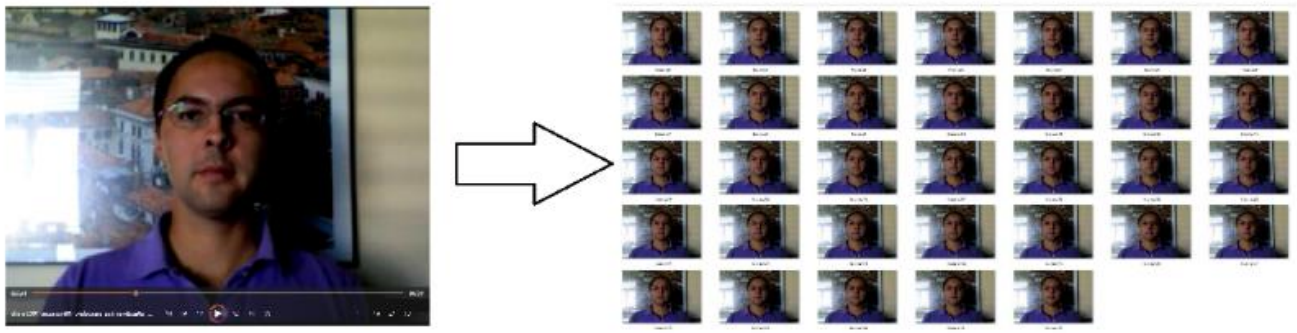
Tahap selanjutnya adalah melakukan ekstraksi fitur kedalaman citra wajah (*depth map estimation*/DME). Proses ini untuk membantu sistem deteksi FAS dalam menentukan *real* atau *fake* citra wajah yang dimasukkan ke dalam sistem FA.

DME dilakukan dengan menggunakan *pretrained model* MiDaS v.2.1. Model ini berbasis *convolutional neural network* (CNN) sehingga mengadopsi prinsip *supervised learning*, di mana *ground truth* diperoleh dari *dataset* untuk aplikasi FAS. Contoh *input* dan *output* dari tahap ekstraksi fitur berbasis DME ini dapat dilihat pada Gambar 3.

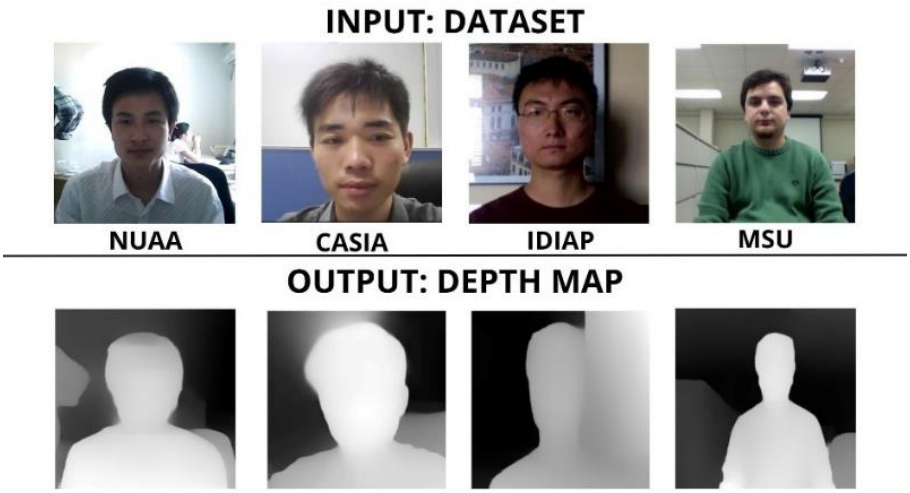
Metode DG dalam penelitian ini diterapkan melalui metode *data generation* (DG). DG merupakan salah satu submetode dari *data manipulation* (DM). Dalam penelitian ini DG dilakukan menggunakan *generative adversarial network* (GAN). Data yang dihasilkan oleh model GAN diharapkan dapat membantu model untuk mempelajari fitur-fitur general atau umum yang tidak bergantung pada fitur-fitur dari *dataset face anti-spoofing* (FAS) tertentu.



Gambar 1 Diagram blok alur proses penelitian



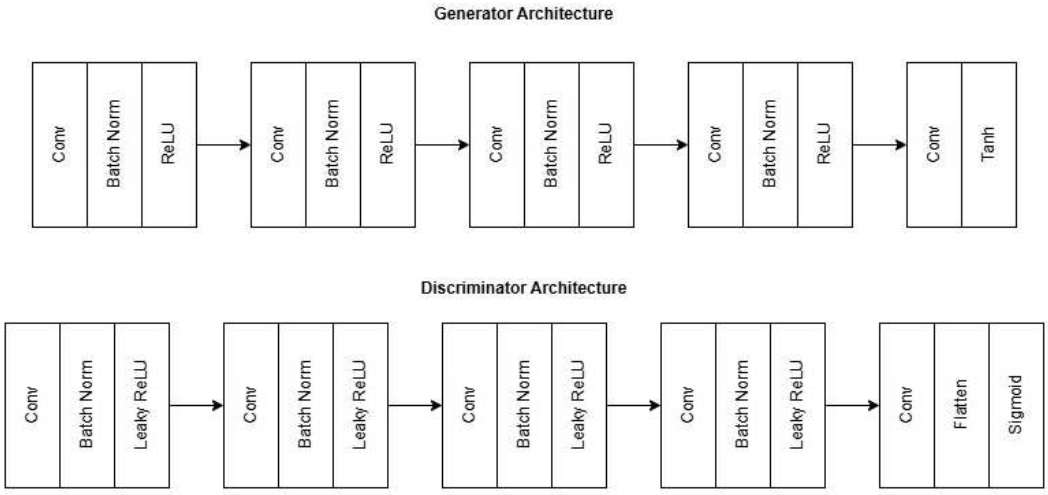
Gambar 2 Contoh konversi video menjadi 32 frame



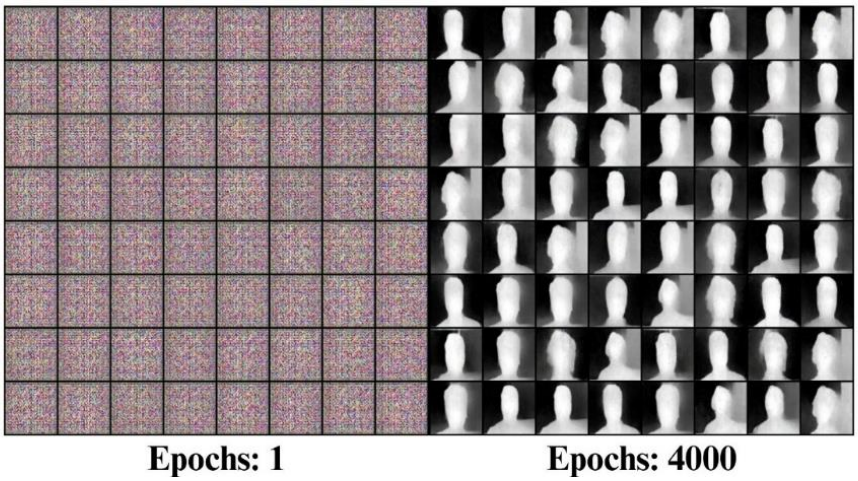
Gambar 3 Contoh hasil ekstraksi ciri estimasi peta kedalaman (*depth map*) dari suatu citra masukan

GAN terdiri dari dua jaringan, yaitu *generator* dan *discriminator*. Dikarenakan terdapat dua jaringan, maka terdapat dua *loss function* yang diterapkan pada GAN. *Loss function* yang digunakan adalah *binary cross entropy*. *Loss* pada *generator* diukur untuk memperbaiki data hasil *generate* agar lebih mirip dengan *data* spoof FAS, sedangkan *loss* pada *discriminator* yang berfungsi untuk mengukur kemampuan *discriminator* dalam menentukan data hasil *generate* dari *generator* adalah data '*real'* (data *spoof* dari *dataset* FAS) atau data '*fake'* (bukan data *spoof* dari *dataset* FAS). Arsitektur dari *generator* dan *discriminator* dapat dilihat dalam Gambar 4, sedangkan data hasil *generate* oleh model GAN ditunjukkan pada Gambar 5.

Tahap berikutnya adalah *data transform*. Proses ini bertujuan untuk menormalisasi (*resize*) ukuran citra wajah menjadi 224×224 piksel (disesuaikan dengan ukuran *output depth map*). Untuk menambahkan variasi data, maka dilakukan *data generation* (DG) yang dalam penelitian untuk *computer vision* biasanya menggunakan teknk seperti *horizontal flip*, *random affine,* dan *color jitter*. Data hasil luaran proses DG diubah ke dalam bentuk tensor untuk mempermudah pemrosesan lebih lanjut. Sampel data hasil DG diperlihatkan dalam Gambar 6.



Gambar 4 Arsitektur *generator* dan *discriminator*



Gambar 5 Data hasil *generate* menggunakan GAN

Gambar 6 Data hasil DG

Pada tahapan *training classifier*, model FAS mempelajari pola data yang sudah dalam bentuk tensor menggunakan *pretrained model* ResNet-50. Model dilatih untuk mendapatkan bobot yang tepat melalui

backpropagation. Layer terakhir pada ResNet-50 mengubah *output layer* sebelumnya yang berbentuk multidimensi menjadi satu dimensi menggunakan operasi *flatten*. *Flatten feature map* kemudian dioperasikan dengan *activate function*. Jumlah fitur dilakukan *down sampling* untuk menghasilkan fitur yang lebih sederhana. Terakhir model melakukan klasifikasi data wajah '*real'* atau '*fake'*.

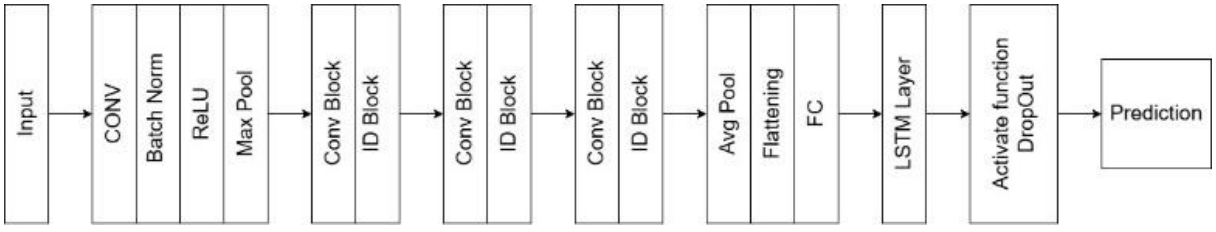
Pada *dataset* FAS berupa video, yaitu CASIA-MFSD (C), Idiap Replay-Attacks, dan MSU-MFSD (M) model FAS mempelajari fitur temporal dengan bantuan tambahan jaringan LSTM. Video merupakan sekuensial dari *frame* demi *frame*. Jaringan LSTM dimanfaatkan untuk mempelajari fitur temporal tersebut. Berbeda dengan model FAS untuk *dataset* gambar, *fully connected layer* dari ResNet-50 tidak langsung melakukan klasifikasi. Fitur dari lapisan *fully connected* menjadi *input* untuk jaringan LSTM. *Input* tersebut kemudian diekstrak pada jaringan LSTM. *Output* dari jaringan LSTM kemudian mengklasifikasikan data sebagai wajah '*real'* atau '*fake'*. Gambar 7 menunjukkan arsitektur ResNet-50 + LSTM yang digunakan dalam penelitian ini.

Pada penelitian ini model FAS diuji dengan dua jenis pengujian. Pertama, model diuji dengan data uji intradataset. Pengujian pertama dimaksudkan untuk menguji kemampuan model dalam memanfaatkan *depth map* sebagai fitur diskriminatif untuk fungsi FAS. Pengujian kedua, dilakukan dengan menguji model pada *data test* yang berbeda dengan *dataset* pelatihan. Pengujian kedua bermaksud untuk mengukur kemampuan generalisasi model pada '*unseen domain'*. *Unseen domain* berarti domain yang belum pernah dilihat model selama pelatihan.

Hyperparameter yang diatur pada model FAS di dalam penelitian ini adalah *learning rate*, jumlah *epoch*, *dropout*, *regulariasi*, *optimizer*, jumlah *hidden layer*, *hidden size*, dan *batch size*. *Tuning hyperparameter* disesuaikan berdasarkan hasil pengujian atau evaluasi model. Misalkan, untuk hasil evaluasi atau pengujian model yang mengindikasikan model '*overfit'* (model dengan hasil yang baik pada pelatihan, tetapi hasilnya buruk ketika divalidasi atau diuji), maka akan dilakukan peningkatan *dropout*, penurunan jumlah *epoch*, penambahan regularisasi, penurunan *learning rate*, dan lainnya. *Tuning hyperparameter* dilakukan hingga model dapat mempelajari pola data dengan baik.

Final result adalah *output* akhir dari proses pelatihan dan pengujian model. Pada penelitian ini hasil akhir dari pelatihan dan pengujian model berupa hasil pengujian model dalam *metric* FAS. Selain itu, bobot model disimpan dalam bentuk '.pt'. Model dapat digunakan untuk melakukan klasifikasi wajah '*real'* atau '*fake'* dengan memanggil model dan memberikan bobot pelatihan model.

Performansi model yang direalisasikan dan disimulasikan dievaluasi dengan menghitung metrik yang lazim dalam menilai kinerja model FAS, yaitu *attack presentation classification error rate* (APCER), *bona fide presentation classification error rate* (BPCER), dan *half-total error rate* (HTER). Metrik APCER digunakan untuk mengukur kesalahan klasifikasi yang terjadi ketika sistem salah mengklasifikasikan presentasi serangan (*spoof*) sebagai wajah asli. Sementara itu, BPCER didefinisikan sebagai rasio dari jumlah presentasi wajah *real* yang salah diklasifikasikan sebagai *fake* (*false negatives*) terhadap total jumlah presentasi wajah asli. HTER adalah metrik yang digunakan untuk menilai performa sistem *face anti-spoofing* secara keseluruhan dalam mengklasifikasikan wajah *real* dan *fake* secara tepat. Ketiga metrik ini dihitung



Gambar 7 Arsitektur ResNet-50 + LSTM

dengan terlebih dahulu empat metrik prediksi untuk mengevaluasi suatu model *classifier*, yaitu *true positive* (TP), *false positive* (FP), *true negative* (TN), dan *false negative* (FN). Dari keempat metrik prediksi ini, maka rumus untuk menghitung APCER, BPCER, dan HTER adalah sebagai berikut:

$$APCER = \frac{FP}{TN + FP} \quad (1)$$

$$BPCER = \frac{FN}{FN + TP} \quad (2)$$

$$HTER = \frac{APCER + BPCER}{2} \quad (3)$$

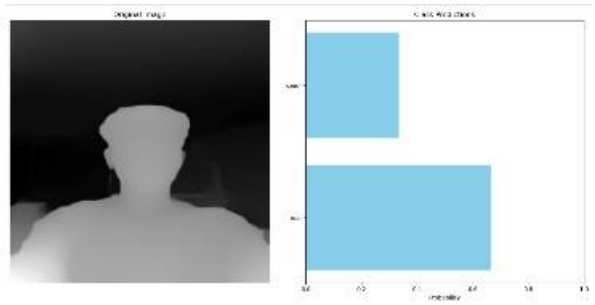
III. HASIL DAN PEMBAHASAN

Model deteksi FAS yang dirancang akan diuji untuk kasus riil (dilakukan oleh peneliti). Pengujian dilakukan untuk membuktikan bahwa model FAS dapat membedakan wajah asli dan palsu dengan baik, pada kasus nyata. Gambar 8 memperlihatkan sampel citra masukan (*real* dan *fake*), sedangkan Gambar 9 menyajikan hasil DME dan hasil prediksi oleh model.

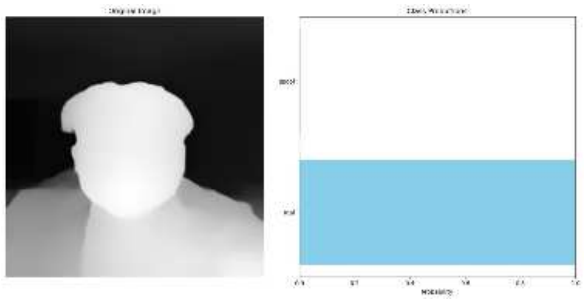
Untuk menguji performa model *face anti-spoofing* dalam mengklasifikasikan wajah *real* dan wajah *fake* dilakukan pengujian model pada *intradataset*. Pengujian *intradataset* berarti pelatihan dan pengujian model dilakukan menggunakan *dataset* yang sama. Pengujian ini menunjukkan performa model FAS dengan *depth map* sebagai fitur diskriminatif. Performa diukur berdasarkan *error rate* pada tiga metrik FAS yang digunakan pada penelitian ini, yaitu *attack presentation classification error rate* (APCER), *bona fide presentation classification error rate* (BPCER), dan *half total error rate* (HTER). Tabel I memperlihatkan model, *training* parameter, dan model parameter yang digunakan untuk melatih model pada masing-masing *dataset*. Hasil simulasi dalam Tabel I, nilai *dropout*-nya adalah 0,5. Tabel II menunjukkan hasil percobaan *intradataset* pada keempat publik *dataset* yang digunakan pada penelitian ini. Seluruh hasil simulasi dalam tulisan ini diperoleh dengan menggunakan PC Intel Core i-9 dengan RAM Corsair DDR 4 ukuran 32 GB (4 buah) serta VGA Card tipe MSI RTX 3090 Gaming X Trio.

(a) Aktual: *Real* (b) Aktual: *Real* (c) Aktual: *Fake* (d) Aktual: *Fake*

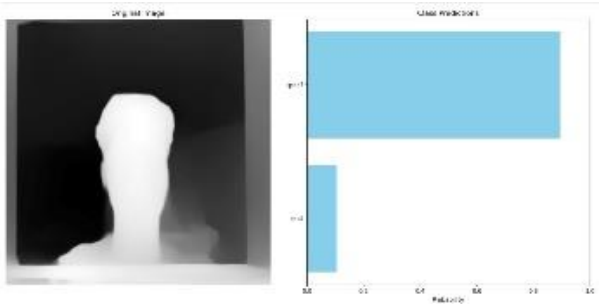
Gambar 8 Contoh pengujian deteksi FAS



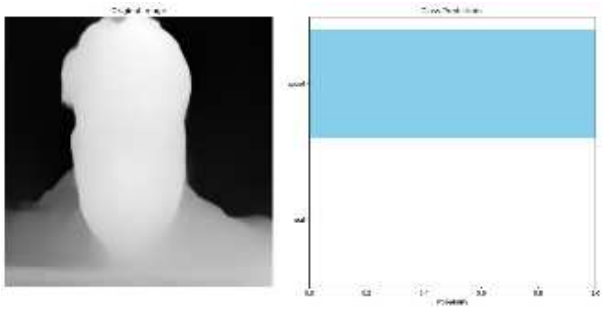
(a) Prediksi: *Real* (*true decision*)



(b) Prediksi: *Real* (*true decision*)



(c) Prediksi: *Fake* (*true decision*)



(d) Prediksi: *Fake* (*true decision*)

Gambar 9 Hasil DME dan deteksi FAS untuk *input* pada Gambar 8

TABEL I
PARAMETER MODEL DAN TRAINING UNTUK PENGUJIAN *INTRADATASET*

Dataset	***Model***	***Parameter Model***		***Parameter Training***		
		***Layer* LSTM**	**LSTM *Hidden Size***	***Optimizer***	***Learning Rate***	***Epoch Number***
NUAA	ResNet–50	–	–	Adam	0,0001	30
CASIA	ResNet–50 + LSTM	1	128	AdamW	*OneCycleR*(1e–4)	50
IDIAP	ResNet–50 + LSTM	1	128	AdamW	*OneCycleR*(1e–4)	50
MSU	ResNet–50 + LSTM	1	128	AdamW	*OneCycleR*(5e–2)	20

TABEL III
HASIL PENGUJIAN MODEL *INTRADATASET*

Dataset	***Data Split***			***Metric* (%)**		
	Train	***Validation***	***Test***	***APCER***	***BPCER***	***HTER***
NUAA	2545	2545	2000	1,25	4,69	2,97
CASIA	154	31	124	5,41	0	2,7
IDIAP	105	45	160	12,5	30	21,25
MSU	140	28	112	14,29	4,76	9,52

Berdasarkan hasil pengujian model FAS pada *intradatset*, didapati bahwa karakteristik *dataset* lebih mempengaruhi performa model dibandingkan banyaknya data latih model. Model FAS yang dilatih dengan *dataset* NUAA merupakan model dengan data latih paling sedikit dan data uji paling banyak. Ketiga model FAS yang dilatih dengan masing-masing 3 *dataset* lainnya, yaitu CASIA, IDIAP, dan MSU memiliki data latih yang lebih banyak dan data uji yang lebih sedikit dibandingkan NUAA.

Hasil metrik FAS pada hasil pengujian dalam Tabel II menunjukkan bahwa model FAS yang dilatih dan diuji pada *dataset* NUAA dan CASIA merupakan model FAS yang paling optimal. Ditinjau dari karakteristiknya, *dataset* NUAA dan CASIA memiliki karakteristik *dataset* yang sama untuk wajah *fake*. Wajah *fake* pada *dataset* NUAA dan CASIA merupakan *spoofing* jenis *print attack* atau *replay attack* yang diumpankan (di-*input*-kan) dari jarak tertentu (cukup jauh) dari layar kamera. Hal tersebut mengakibatkan pemanfaatan *depth map* lebih optimal pada *dataset* NUAA dan CASIA, dibandingkan pada *dataset* IDIAP *replay attack* dan MSU. Pernyataan tersebut dibuktikan dengan nilai *error* (APCER, BPCER, dan HTER) yang rendah pada pengujian model *intradataset* NUAA dan CASIA. Sebaliknya, nilai *error* pada (APCER, BPCER, dan HTER) pengujian model intra-*dataset* IDIAP *Replay Attack* dan MSU yang cenderung lebih tinggi.

Sementara itu, untuk kondisi adanya perbedaan gap distribusi antara data latih dan data uji (kondisi *unseen domain*), simulasi dilakukan dengan menetapkan dataset NUAA dan IDIAP sebagai data latih serta untuk data uji diambil dari *dataset* CASIA. Melalui penetapan ini diharapkan model akan dilatih untuk dapat mempelajari fitur dari 2 *dataset* yang memiliki karakteristik berbeda. Harapannya adalah model dapat mempelajari karakteristik *spoofing* pada data latih dengan baik sehingga ketika diuji pada *dataset* yang berbeda, model dapat mengklasifikasikan wajah *real* dan *fake* dengan baik. Apabila *goal* atau tujuan tersebut tercapai, maka artinya model tidak kebingungan dengan fitur yang dipelajarinya selama pelatihan. Tabel III menyajikan *hyperparameter* model yang digunakan pada pengujian lintas *dataset*, sedangkan hasil dari pengujian lintas *dataset* ditunjukkan pada Tabel III pada baris pertama (NUAA dan IDIAP).

TABEL IIIII
HYPERPARAMETER MODEL UNTUK PENGUJIAN LINTAS *DATASET*

Dataset	***Model***	***Hyperparameter Model***			
		Dropout	***Optimizer***	***LearningRate***	***Epoch Number***
NUAA + IDIAP	ResNet–50	0,3	AdamW	10^{-6}	20
NUAA + IDIAP + GAN	ResNet–50	0,3	AdamW	10^{-6}	20

TABEL IVV
KOMPOSISI DATASET DAN METRIK FAS

Dataset	***Data Split***				***Metric* (%)**		
	***Train* (NUAA)**	***Train* (IDIAP)**	***Train* (GAN)**	***Test* (CASIA)**	***APCER***	***BPCER***	***HTER***
NUAA + IDIAP	700	700	–	600	38,5	12,5	25,5
NUAA + IDIAP + GAN	700	700	256	600	38,5	2,75	20,63

Pengujian model FAS dengan data latih NUAA dan IDIAP menunjukkan hasil yang tidak cukup baik. Hasil pengujian memperlihatkan evaluasi metrik FAS masih memiliki persentase *error* cukup besar, dengan nilai minimum evaluasi metrik FAS untuk BPCER adalah sebesar 12,5% dan nilai metrik FAS untuk APCER adalah sebesar 38,5% (baris pertama Tabel IV). Hasil pengujian lintas dataset menunjukkan bahwa model sudah cukup baik dalam mengklasifikasikan wajah *real* sebagai wajah *real*, tetapi cukup kesulitan dalam mengklasifikasikan wajah *fake* sebagai wajah *fake*.

Oleh karena itu, dalam penelitian ini perlu dilakukan sesuatu agar dapat meningkatkan kemampuan generalisasi model. Dalam tulisan ini, *generative adversarial network* (GAN) dimanfaatkan untuk menambah jumlah data latih agar dapat memperkaya variasi data dalam pelatihan model. Pendekatan ini dilakukan sehingga model FAS dapat mempelajari fitur yang lebih general dan tidak bergantung pada karakteristik *dataset* tertentu saja (NUAA dan IDIAP). GAN dilatih untuk men-*generate* data *spoof* dari *dataset* pelatihan, yaitu NUAA dan IDIAP *replay attack.*

Setelah data latih tambahan hasil luaran dari GAN dimasukkan ke dalam data latih model FAS, pengujian yang sama (pengujian lintas *dataset*) kembali dilakukan. Sekarang data latih terdiri atas *dataset* NUAA, IDIAP *replay attack*, dan data hasil GAN, sedangkan data uji masih sama, yaitu *dataset* CASIA. Tabel III menunjukkan *hyperparameter* model yang digunakan pada pengujian lintas *dataset* (sama untuk tanpa dan dengan GAN).

Pada pengujian model FAS dengan data latih NUAA dan IDIAP *replay attack* saja (tanpa data tambahan hasil luaran GAN), metrik BPCER yang didapatkan mencapai angka12,5%. Pada pengujian model FAS dengan data latih NUAA, IDIAP *replay attack*, ditambah data hasil GAN, nilai metrik BPCER yang didapatkan bisa mencapai hanya 2,75%. Penambahan data hasil GAN pada model memberikan penurunan nilai BPCER sebesar 9,75% dan HTER sebesar 4,87%. Penambahan data hasil GAN memperkaya kemampuan generalisasi model sehingga hasil pengujian untuk *setting* simulasi, seperti untuk kondisi lintas dataset, menunjukkan bahwa model sudah cukup baik dalam mengklasifikasikan wajah *real* sebagai wajah *real*, demikian pula dalam mengklasifikasikan wajah *fake* sebagai wajah *fake*.

IV. SIMPULAN

Tulisan ini membahas tentang bagaimana mengatasi adanya perbedaan (*gap*) distribusi dari *dataset* latih dan *dataset* uji untuk persoalan *face anti-spoofing* (FAS). Ekstraksi ciri untuk mendeteksi *spoofing* adalah dengan mencari peta kedalaman citra masukan. Pendekatan yang digunakan adalah dengan memakai metode *data generation* dan diimplementasikan dengan mengadopsi model GAN. Hasil yang diperoleh khususnya untuk adanya perbedaan distribusi dapat menurunkan metrik BPCER hingga sebesar 9,75% dan HTER 4,87% dengan data latih NUAA dan IDIAP *replay attack*, ditambah data hasil GAN serta *dataset* CASIA sebagai data uji. Dalam penelitian selanjutnya, bisa diujikan skenario pendekatan selain *data generation* sebagai sarana untuk mengatasi isu perbedaan distribusi antara *dataset* uji dengan *dataset* latih, seperti *learning strategy* dan *representation learning* [12] atau mengkombinasikan ketiga pendekatan tersebut.

UCAPAN TERIMA-KASIH

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada Universitas Kristen Maranatha (UKM) yang sudah memberikan dukungan dana dalam pelaksanaan penelitian ini.

DAFTAR REFERENSI

[1] R. Cai *et al.*, “S-Adapter: generalizing vision transformer for face anti-spoofing with statistical tokens,” *IEEE Trans. Inf. Forensics Secur.*, vol. 19, pp. 8385–8397, 2024, doi: 10.1109/TIFS.2024.3420699.

[2] Y. Liu, Y. Chen, W. Dai, M. Gou, C. T. Huang, and H. Xiong, “Source-free domain adaptation with domain generalized pretraining for face anti-spoofing,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 8, pp. 5430–5448, 2024, doi: 10.1109/TPAMI.2024.3370721.

[3] W. Liu and Y. Pan, “Spatio-temporal-based action face anti-spoofing detection via fusing dynamics and texture face keypoints cues,” *IEEE Transactions on Consumer Electronics*, vol. 70, no. 2, pp. 2401-2413, Feb. 2024, doi: 10.1109/TCE.2024.3361480.

[4] T. Zheng et al., “MFAE: masked frequency autoencoders for domain generalization face anti-spoofing,” *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 4058-4069, 2024, doi: 10.1109/TIFS.2024.3371266.

[5] C. Kong, K. Zheng, Y. Liu, S. Wang, A. Rocha, and H. Li, “M3FAS: an accurate and robust multimodal mobile face anti-spoofing system,” *IEEE Transactions on Dependable and Secure Computing*, vol. 21, no. 6, pp. 1230-1242, Jun. 2024, doi: 10.1109/TDSC.2024.3381598.

[6] W. Zheng, M. Yue, S. Zhao, and S. Liu, “Attention-based spatial-temporal multi-scale network for face anti-spoofing,” *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol. 5, no. 2, pp. 296-307, Apr. 2023, doi: 10.1109/TBIOM.2021.3066983.

[7] D. Wang et al., “Wild face anti-spoofing challenge 2023: benchmark and results,” in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, 2023, pp. 10209048, doi: 0.1109/CVPRW59228.2023.00679.

[8] J. Yang, Z. Lei, and S. Z. Li, “Learn convolutional neural network for face anti-spoofing,” *arXiv preprint arXiv:1408.5601*, 2014. [Online]. Available: https://arxiv.org/abs/1408.5601

[9] Y. Atoum, Y. Liu, A. Jourabloo, and X. Liu, “Face anti-spoofing using patch and depth-based CNNs,” in *Proc. IEEE Int. Joint Conf. Biometrics (IJCB)*, 2017, pp. 8272713, doi: 10.1109/BTAS.2017.8272713.

[10] Y. Liu, Y. Tai, J. Li, S. Ding, C. Wang, and F. Huang, “Aurora Guard: Real-Time Face Anti-Spoofing via Light Reflection,” *arXiv preprint arXiv:1902.10311*, 2019. [Online]. Available: https://arxiv.org/abs/1902.10311

[11] N. M. Adams, "Dataset shift in machine learning," *Journal of the Royal Statistical Society Series A: Statistics in Society*, vol. 173, no. 1, p. 274, 2010. doi: 10.1111/j.1467-985x.2009.00624_10.x.

[12] J. Wang *et al.*, “Generalizing to unseen domains: a survey on domain generalization,” *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 8, 2023, pp. 8052–8072, doi: 10.1109/TKDE.2022.3178128.

[13] Y. Liu *et al.*, “Towards unsupervised domain generalization for face anti-spoofing,” *Proc. IEEE Int. Conf. Comput. Vis.*, pp. 20597–20607, 2023, doi: 10.1109/ICCV51070.2023.01888.

[14] Q. Zhou *et al.*, “Instance-aware domain generalization for face anti-spoofing,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2023-June, pp. 20453–20463, 2023, doi: 10.1109/CVPR52729.2023.01959.

Tio Dewantho Sunoto, memperoleh gelar Sarjana Teknik Elektro di Universitas Kristen Maranatha (UKM) Bandung dan gelar Magister Elektroteknik di Institut Teknologi Bandung (ITB). Bidang yang ditekuni teknik sistem kontrol. Saat ini sebagai pengajar di UKM.

Daniel Setiadikarunia, memperoleh gelar Sarjana Teknik Elektro di Universitas Kristen Maranatha (UKM) Bandung dan gelar Magister Elektroteknik dan Doktor di Institut Teknologi Bandung (ITB). Bidang yang ditekuni teknik komputer dan teknik telekomunikasi. Saat ini sebagai pengajar di UKM.

Riko Arlando Saragih, memperoleh gelar Sarjana Teknik Elektro di Institut Teknologi Bandung (ITB), gelar Magister Teknik Elektro di ITB, dan gelar Doktor di Universitas Indonesia (UI). Bidang yang ditekuni *pattern recognition*, *machine lerning*, dan *digital signal processing*. Saat ini sebagai pengajar di Universitas Kristen Maranatha (UKM).

Elia Moses, mahasiswa Program Studi Teknik Elektro Universitas Kristen Maranatha (UKM) Bandung. Studinya telah selesai.