

PENGEMBANGAN SIAMESE NEURAL NETWORK DALAM IMPLEMENTASI FACE RECOGNITION

Fernando¹, Muhammad Ezar Al Rivan²
fernand0753@mhs.mdp.ac.id¹, meedzhar@mdp.ac.id²
Universitas Multi Data Palembang

Abstrak

Pengenalan wajah merupakan salah satu teknologi biometrik yang terus berkembang, terutama untuk aplikasi verifikasi identitas yang cepat dan akurat. Penelitian ini mengembangkan sistem face recognition menggunakan arsitektur Siamese Neural Network (SNN) dengan ekstraktor fitur SqueezeNet dan menggunakan fungsi loss contrastive loss. Dataset yang digunakan terdiri dari 1200 gambar wajah milik 30 orang, dengan berbagai sudut pandang dan kondisi pencahayaan. Eksperimen dilakukan dengan empat kombinasi hyperparameter, mencakup variasi optimizer (SGD dan Adam) serta learning rate (0,01 dan 0,001). Hasil pengujian menunjukkan bahwa konfigurasi terbaik diperoleh saat menggunakan learning rate 0,001, baik dengan optimizer SGD maupun Adam, dengan nilai F1-score sebesar 1,00. Model ini juga menunjukkan kecepatan dan akurasi tinggi dalam proses verifikasi wajah, menjadikannya cocok untuk implementasi pada perangkat mobile dan aplikasi presensi berbasis wajah.

Kata Kunci: Face Recognition, Siamese Neural Network, SqueezeNet.

Abstract

Face recognition is a rapidly evolving biometric technology, especially for fast and accurate identity verification applications. This study develops a face recognition system using the Siamese Neural Network (SNN) architecture, employing SqueezeNet as the feature extractor and contrastive loss as the loss function. The dataset consists of 1200 face images from 30 individuals, captured from various angles and under uniform lighting conditions. Experiments were conducted using four hyperparameter combinations, including variations in optimizers (SGD and Adam) and learning rates (0.01 and 0.001). The results show that the best configuration was achieved using a learning rate of 0.001 with either SGD or Adam, resulting in an F1-score of 1.00. The model demonstrated high speed and accuracy in face verification, making it suitable for implementation on mobile devices and face-based attendance applications.

Keywords: Face Recognition, Siamese Neural Network, SqueezeNet.

1. PENDAHULUAN

Kemajuan pada teknologi kecerdasan buatan (Artificial Intelligence/AI) telah memberikan pengaruh besar di berbagai sektor, salah satunya dalam sistem keamanan dan identifikasi biometrik. Wajah dan sidik jari manusia menjadi objek biometrik yang mudah untuk dipindai serta diproses [1]. Pengenalan wajah atau face recognition masih menjadi tantangan tersendiri [2]. Metode biometrik ini sangat populer karena sifatnya yang tidak invasif dan dapat mengidentifikasi seseorang secara otomatis dengan efisien.

Teknologi face recognition telah banyak dimanfaatkan dalam sistem kehadiran, kontrol akses, pengawasan, hingga verifikasi identitas pada aplikasi digital.

Secara sederhana, sistem pengenalan wajah bekerja dengan membandingkan gambar wajah yang dimasukkan dengan data wajah yang tersimpan di database, kemudian menentukan tingkat kemiripan antara keduanya [3]. Teknologi face recognition telah digunakan secara luas dalam berbagai bidang, seperti keamanan, kehadiran pegawai, layanan perbankan digital, serta layanan

publik [7]. Di lingkungan kerja maupun pendidikan, teknologi ini diimplementasikan pada sistem presensi otomatis berbasis aplikasi mobile atau kamera cerdas, sehingga mampu menggantikan metode manual yang rawan kecurangan [20].

Untuk mendukung efektivitas sistem pengenalan wajah, digunakan berbagai metode kecerdasan buatan, salah satunya adalah Convolutional Neural Network (CNN). CNN merupakan tipe Deep Neural Network yang dirancang khusus untuk mengolah data dua dimensi dan memiliki kedalaman jaringan yang kompleks, sehingga sangat efektif dalam pengolahan citra [4]. Selain CNN, terdapat juga pendekatan lain yang terbukti efektif, yaitu Siamese Neural Network (SNN). SNN merupakan model jaringan saraf yang mempelajari kesamaan antara dua gambar input dengan cara menghitung jarak pada ruang fitur untuk menentukan apakah kedua gambar wajah tersebut berasal dari orang yang sama [8]. SNN banyak digunakan untuk mencari kesamaan atau keterkaitan antara dua objek yang sebanding [5]. Proses pencocokan tersebut dapat dilakukan dengan dua metode, yaitu verifikasi (menentukan apakah dua wajah identik) dan identifikasi (mencari identitas satu wajah di antara banyak wajah) [19].

Arsitektur SNN banyak diaplikasikan, termasuk untuk pengenalan wajah, di mana sistem membandingkan dua gambar wajah untuk menentukan apakah keduanya adalah orang yang sama atau tidak. Pada SNN, dilakukan beberapa tahapan seperti pembelajaran fitur dan perhitungan jarak antara dua gambar untuk mengukur tingkat kemiripan [16]. SNN memiliki dua input untuk membandingkan dua pola dan menghasilkan satu output berupa nilai kesamaan antara kedua pola tersebut. Ekstraksi fitur dilakukan melalui lapisan konvolusi, di mana tiap cabang menghasilkan representasi fitur dari gambar masukan [5]. Tujuan proses ini adalah untuk menemukan

pola serta karakteristik penting dalam gambar. Hasil ekstraksi fitur berupa vektor embedding, yang selanjutnya digunakan untuk menghitung tingkat kemiripan antara dua vektor fitur menggunakan fungsi jarak atau kesamaan [6].

Salah satu metode pengukuran yang digunakan adalah Euclidean Distance, yang menghitung seberapa jauh dua embedding dalam ruang vektor. Semakin kecil jarak antar vektor, maka kemungkinan kedua gambar merupakan wajah dari orang yang sama akan semakin besar. Beberapa penelitian terbaru menunjukkan bahwa SNN lebih fleksibel dibandingkan model klasifikasi konvensional, karena tidak membutuhkan pelabelan kelas yang rumit dan tetap efektif meski data yang tersedia terbatas. Inilah yang menjadikan SNN sesuai untuk skenario few-shot learning, di mana sistem harus mengenali wajah baru meski hanya ada satu atau dua gambar referensi [18].

Penelitian oleh Suwindra dkk. Pada tahun 2021 menunjukkan bahwa SNN mampu mengenali kemiripan antarspesies burung secara efektif dengan akurasi pengujian mencapai 93,90% dan AUC sebesar 93%, menggunakan dataset besar dengan 260 spesies [6]. Arum dkk. Pada tahun 2023 menerapkan SNN dalam pendekatan one-shot learning untuk pengenalan wajah dan berhasil meraih akurasi 98,5%, melampaui metode PCA dan LDA [24]. Kartarina & Imam pada tahun 2019 juga membuktikan efektivitas SNN yang dikombinasikan dengan arsitektur ResNet-50 dalam verifikasi wajah, dengan akurasi 92% dan AUC 97% pada dataset LFW [25]. Ketiga penelitian ini memperkuat bahwa SNN merupakan pendekatan yang andal, fleksibel, dan akurat untuk berbagai kebutuhan pengenalan citra dalam kondisi data terbatas maupun kompleksitas objek tinggi.

Penelitian ini bertujuan untuk mengembangkan sistem verifikasi wajah berbasis Siamese Neural Network menggunakan arsitektur SqueezeNet sebagai

ekstraktor fitur. Tujuan utamanya adalah menghasilkan model verifikasi wajah yang ringan, cepat, dan akurat, serta mampu bekerja efektif meskipun hanya dengan jumlah data terbatas. Kontribusi utama dari penelitian ini terletak pada penggunaan kombinasi SNN dan SqueezeNet yang dioptimalkan melalui pengujian berbagai hyperparameter, termasuk learning rate dan optimizer, serta evaluasi kinerja model menggunakan metrik presisi, recall, dan F1-score. Penelitian ini juga mendemonstrasikan performa model dengan waktu verifikasi di bawah lima detik, yang sangat relevan untuk aplikasi presensi mobile dan real-time. Namun, keterbatasan dari penelitian ini terletak pada jumlah dataset yang masih relatif kecil dan kondisi pencahayaan yang homogen. Penelitian lanjutan disarankan untuk menguji model pada lingkungan yang lebih kompleks dan variatif guna mengukur robust-nya sistem secara menyeluruh.

2. METODE PENELITIAN

Data

Data yang digunakan dalam Pengembangan Siamese Neural Network dalam implementasi Face Recognition berupa gambar dengan format jpg. Dataset yang digunakan terdiri dari 30 orang, dengan berbagai sudut pandang seperti hadap depan, hadap kiri dan hadap kanan, serta dalam kondisi pencahayaan yang sama. Masing – masing orang diambil gambarnya sebanyak 40 gambar. Dengan total gambar dataset sebanyak 1200 gambar. kemudian akan dilakukan augmentasi data pada gambar tersebut. Kemudian dilakuakn resize untuk semua gambar menjadi ukuran 224 x 224 pixel. Kemudian pada dataset tersebut dilakukan augmentation data. dataset ini akan menggunakan pembagian data pelatihan dan data validasi dengan perbandingan 8:2.

Algoritma

Dalam penelitian ini, algoritma utama yang digunakan adalah Siamese Neural Network (SNN), sebuah arsitektur deep

learning yang dirancang untuk membandingkan dua input dan menentukan tingkat kemiripannya. SNN bekerja dengan dua cabang jaringan saraf identik yang berbagi bobot dan parameter. Setiap cabang menerima gambar wajah sebagai input dan mengekstraksi fitur (embedding) dari gambar tersebut. Kemudian, jarak antara dua embedding dihitung menggunakan metrik seperti Euclidean Distance untuk menentukan apakah kedua gambar berasal dari orang yang sama[8].

Keunggulan SNN terletak pada kemampuannya menangani permasalahan verifikasi wajah dalam lingkungan dinamis, di mana jumlah identitas dapat berubah seiring waktu. Hal ini juga menjadikan SNN cocok untuk aplikasi real-time seperti sistem absensi mobile, di mana data wajah terus bertambah tanpa perlu pelatihan ulang untuk seluruh model. arsitektur Siamese juga sangat cocok digunakan karena model ringan serta memiliki akurasi yang cukup baik dan efisiensi komputasi, sehingga cocok diimplementasikan pada perangkat edge atau smartphone [9]. Sehingga SNN juga dapat dikombinasikan dengan arsitektur CNN yang ringan seperti MobileFace atau EfficientNet, sehingga cocok diimplementasikan pada perangkat edge seperti smartphone dan kamera cerdas. Ini menjadikan SNN sangat ideal untuk sistem verifikasi real-time dengan waktu respon yang cepat dan penggunaan daya yang rendah [15]. Siamese juga mampu mengenali wajah baru dengan jumlah data yang sangat terbatas karena fokus pembelajarannya adalah pada hubungan antar pasangan gambar, bukan klasifikasi individual [13]. Kelebihan lain dari SNN adalah ketahanannya terhadap variasi pencahayaan, sudut pandang, dan ekspresi wajah. Karena model ini mempelajari jarak atau kemiripan antara representasi fitur wajah, ia lebih robust terhadap gangguan visual dan noise dibandingkan model klasifikasi konvensional [14]. Untuk melatih jaringan, digunakan Contrastive Loss, yaitu fungsi

kerugian yang mengatur agar pasangan citra positif (wajah dari orang yang sama) memiliki jarak embedding yang kecil, sementara pasangan negatif (wajah dari orang berbeda) memiliki jarak yang lebih besar dari margin tertentu [17]. Pendekatan ini terbukti efektif dalam meningkatkan kemampuan diskriminatif model terhadap wajah-wajah dengan karakteristik yang mirip [7].

Implementasi SNN dalam sistem face recognition telah terbukti mampu mencapai kinerja yang kompetitif dengan waktu verifikasi di bawah lima detik, bahkan dalam kondisi pencahayaan dan latar belakang yang bervariasi. Dengan kombinasi arsitektur ringan dan loss function yang tepat, model ini menjadi pilihan ideal untuk solusi autentikasi berbasis wajah yang cepat, ringan, dan akurat [10].

Dalam implementasinya, penelitian ini mengintegrasikan SNN dengan SqueezeNet sebagai modul feature extractor. SqueezeNet adalah arsitektur CNN ringan yang diperkenalkan oleh Iandola pada tahun 2016 dan dirancang untuk mencapai akurasi setara AlexNet tetapi dengan ukuran model yang jauh lebih kecil—hanya sekitar 0.5MB. Arsitektur ini menggunakan blok khusus bernama Fire Module, yang terdiri dari dua tahap utama: Squeeze layer dan Expand layer. Squeeze layer menggunakan konvolusi 1×1 untuk mengurangi jumlah channel (saluran fitur) dari input, sehingga menghemat jumlah parameter dan komputasi. Setelah proses squeeze, data masuk ke expand layer yang terdiri dari kombinasi konvolusi 1×1 dan 3×3 untuk mengekspansi kembali fitur ke bentuk representasi yang lebih kompleks dan mendalam. Hasil dari kedua filter ini kemudian digabung (concatenated) untuk membentuk output akhir dari Fire Module. Struktur ini dirancang agar mampu menangkap pola spasial dan tekstural dengan efisiensi tinggi.

SqueezeNet juga mengimplementasikan teknik late downsampling, yakni pengurangan resolusi spasial yang dilakukan di tahap akhir

jaringan, bukan di awal seperti kebanyakan CNN. Strategi ini memungkinkan lebih banyak informasi spasial dipertahankan selama proses ekstraksi fitur awal. Dengan jumlah parameter yang sangat rendah, SqueezeNet tetap mampu menangkap fitur kompleks dari citra wajah dan menghasilkan embedding yang representatif namun tetap ringan, menjadikannya sangat cocok untuk sistem verifikasi wajah berbasis SNN yang ditujukan untuk perangkat mobile atau edge computing.

Integrasi antara SNN dan SqueezeNet pada penelitian ini bertujuan untuk menciptakan sistem verifikasi wajah yang tidak hanya akurat, tetapi juga memiliki performa cepat, ringan, dan dapat diterapkan secara efisien dalam kondisi nyata, termasuk pada perangkat dengan keterbatasan sumber daya.

Perancangan Model

Perancangan model dalam penelitian ini bertujuan untuk menghasilkan sistem face recognition yang mampu melakukan verifikasi wajah secara efisien dan akurat dengan waktu respon yang cepat. Model ini dikembangkan untuk memenuhi kebutuhan sistem autentikasi yang ringan namun tetap andal, khususnya pada lingkungan komputasi terbatas. Model dirancang berdasarkan pendekatan Siamese Neural Network (SNN) yang terbukti efektif dalam menangani permasalahan verifikasi citra wajah, terutama dalam situasi dengan data terbatas dan jumlah identitas yang dinamis. SNN bekerja berdasarkan prinsip pembelajaran kemiripan antar pasangan data, bukan klasifikasi langsung ke dalam label tertentu, sehingga sangat fleksibel dalam menghadapi data baru.

Siamese Neural Network merupakan arsitektur pembelajaran yang terdiri dari dua jaringan identik (kembar) yang berbagi bobot dan struktur yang sama. Kedua jaringan ini menerima sepasang gambar wajah sebagai input dan mengekstraksi fitur penting dari masing-masing gambar. Dengan memproses gambar input (wajah) dan mengekstrak fitur-

fitur penting dalam bentuk vektor embedding. Setiap vektor embedding merupakan representasi numerik dari fitur wajah yang telah disaring dan dipadatkan oleh jaringan. Untuk mengekstraksi fitur penting dari masing-masing gambar, pada penelitian ini digunakan algoritma SqueezeNet yang berfungsi sebagai feature extractor utama. Penggunaan SqueezeNet didasarkan pada pertimbangan efisiensi komputasi dan jumlah parameter yang rendah tanpa mengorbankan akurasi. Dengan memproses gambar input (wajah) dan mengekstrak fitur-fitur penting dalam bentuk vektor embedding [12].

Pada tahap ekstraksi fitur, gambar pertama kali diproses melalui lapisan konvolusi awal dengan kernel 7×7 dan stride 2 yang bertujuan untuk menangkap fitur dasar seperti tepi dan tekstur. Lapisan ini bertindak sebagai penyaring awal terhadap informasi spasial dalam citra, memfokuskan perhatian jaringan pada elemen-elemen penting sejak awal proses. Selanjutnya, gambar melewati lapisan MaxPooling (3×3 , stride 2) untuk mengurangi dimensi spasial tanpa kehilangan informasi penting. Proses downsampling ini membantu mempercepat proses komputasi sekaligus menjaga kestabilan fitur.

SqueezeNet menggunakan komponen utama yang disebut Fire Module. Fire Module adalah komponen inti dalam arsitektur SqueezeNet yang dirancang untuk meningkatkan efisiensi komputasi dengan mengurangi jumlah parameter jaringan tanpa mengorbankan akurasi [21]. Fire Module terdiri dari dua bagian utama, yaitu Squeeze Layer dan Expand Layer. Squeeze Layer berfungsi sebagai tahap pertama dalam Fire Module yang bertujuan untuk mengurangi jumlah channel fitur sebelum diproses lebih lanjut. Lapisan ini terdiri dari konvolusi 1×1 , yang bertugas menyaring dan menurunkan kompleksitas input. Tujuan dari lapisan ini adalah untuk mengurangi kedalaman input, yaitu jumlah saluran atau depth sebelum data tersebut diteruskan ke tahap berikutnya.

Setelah proses squeeze, lapisan expand menggunakan dua set filter konvolusi: filter 1×1 dan filter 3×3 yang diterapkan pada output dari squeeze layer. Kombinasi keduanya membantu menangkap informasi lokal maupun global dari citra. Output dari kedua jenis filter ini kemudian digabungkan dalam depth untuk membentuk output akhir dari Fire Module [22]. Gabungan ini memperkaya representasi fitur tanpa menambah beban parameter secara signifikan. Dengan menggunakan SqueezeNet sebagai feature extractor memungkinkan jaringan menangkap pola fitur dengan kompleksitas yang semakin meningkat di setiap tahapannya.

Setelah melalui seluruh Fire Module, gambar melewati lapisan konvolusi akhir (1×1) untuk menyesuaikan jumlah channel, sebelum diterapkan Adaptive Average Pooling, yang mengubah fitur menjadi vektor embedding [12]. Pooling ini berfungsi menyatukan semua informasi spasial menjadi satu vektor tetap yang mewakili keseluruhan fitur gambar secara kompak. Vektor embedding dari dua gambar kemudian dibandingkan menggunakan metrik jarak, dengan Euclidean Distance. Euclidean Distance digunakan untuk mengukur jarak kemiripan pada output dari masing-masing jaringan [23]. Jika jarak antar fitur rendah, maka gambar dianggap berasal dari orang yang sama. Sebaliknya, jarak yang tinggi mengindikasikan bahwa gambar berasal dari individu yang berbeda [11]. Pemilihan SqueezeNet sebagai backbone CNN yang ringan namun kuat memperkuat keseluruhan sistem sehingga model tetap responsif meski dijalankan di perangkat dengan keterbatasan sumber daya. Dengan demikian, integrasi SNN dan SqueezeNet menghasilkan solusi pengenalan wajah yang ringan, cepat, dan akurat, sesuai dengan tujuan utama dari perancangan model dalam penelitian ini.

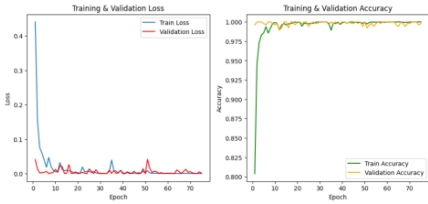
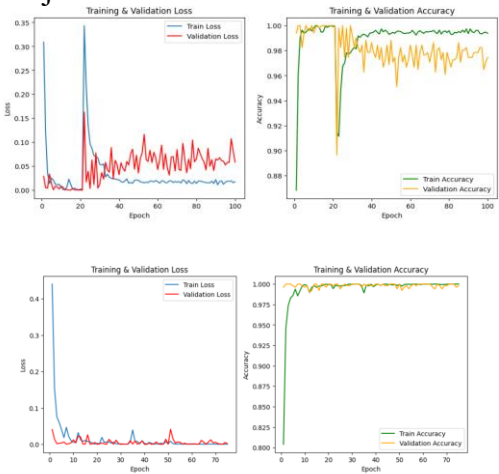
3. HASIL DAN PEMBAHASAN

Dari hasil pengujian model yang dilakukan dengan melakukan hyperparameter tuning yang mencakup penyesuaian parameter seperti learning rate, optimizer, dan epoch untuk menghasilkan metrik evaluasi. Metrik evaluasi yang dihasilkan berupa precision, recall, dan accuracy. Serta untuk mencari kombinasi terbaik untuk model yang di kembangkan.

Tabel 1

Hyperparameter Tuning	
Paramter	Nilai
Epoch	50,75,100
Learning rates	0.001, 0.01
Optimizer	SGD, Adam

Pada dataset pertama, dilakukan pengujian terhadap hyperparameter pertama untuk dataset pertama epoch dengan nilai 50, 75, dan 100. Pengujian dilakukan menggunakan konfigurasi SGD dengan initial learning rate (lr) sebesar 0,01. Berdasarkan hasil pengujian, konfigurasi terbaik diperoleh pada epoch ke-100. Rincian hasil pengujian disajikan pada Tabel 2. Visualisasi performa pelatihan model ditunjukkan pada Gambar 1 yang memperlihatkan grafik loss dan akurasi selama proses training dan validasi. Gambar 2 menampilkan hasil confusion matrix yang memberikan gambaran akurasi klasifikasi model terhadap masing-masing kelas pada data uji.

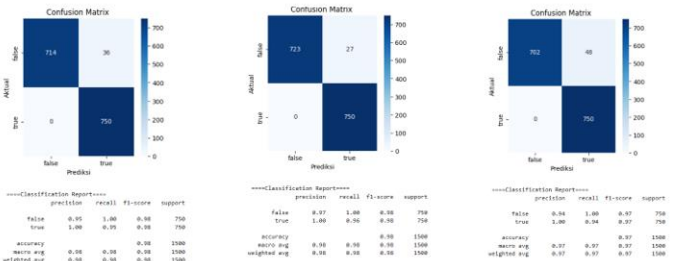


Gambar 1 Visualisasi Performa Pelatihan Model Hyperparameter Pertama

Tabel 2

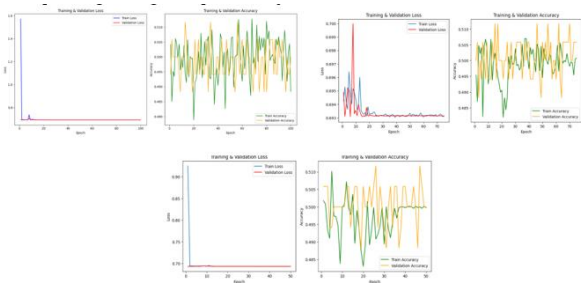
Hasil Hyperparameter Pertama

Epoch	Learning Rate	Optimizer	label	Precision	Recall	F1-score
50	0.01	SGD	True	1.00	0.95	0.98
			False	0.95	1.00	0.98
75			True	1.00	0.94	0.97
			False	0.94	1.00	0.97
100			True	1.00	0.96	0.98
			False	0.97	1.00	0.98



Gambar 2 Hasil Confusion Matrix Hyperparameter Kedua

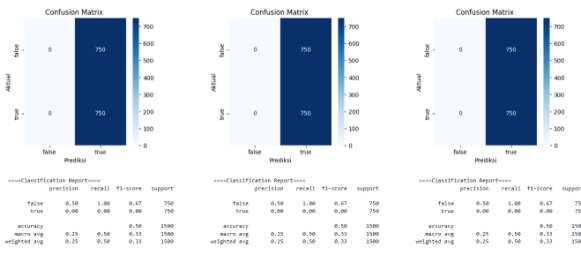
Pada dataset pertama, dilakukan pengujian terhadap hyperparameter kedua untuk dataset pertama epoch dengan nilai 50, 75, dan 100. Pengujian dilakukan menggunakan konfigurasi optimizer ADAM dengan initial learning rate (lr) sebesar 0,01. Berdasarkan hasil pengujian, konfigurasi terbaik diperoleh pada epoch ke-50. Rincian hasil pengujian disajikan pada Tabel 2. Visualisasi performa pelatihan model ditunjukkan pada Gambar 3 yang memperlihatkan grafik loss dan akurasi selama proses training dan validasi. Gambar 4 menampilkan hasil confusion matrix yang memberikan gambaran akurasi klasifikasi model terhadap masing-masing kelas pada data uji.



Gambar 3 Visualisasi Performa Pelatihan Model Hyperparameter Kedua

Tabel 3
 Hasil Hyperparameter Kedua

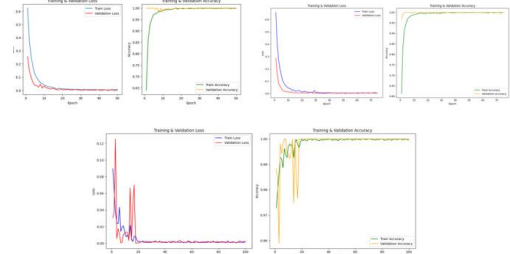
Epoch	Learning Rate	Optimizer	label	Precision	Recall	F1-score
50	0.01	Adam	True	0.00	0.00	0.00
			False	0.50	1.00	0.67
75			True	0.00	0.00	0.00
			False	0.50	1.00	0.67
100			True	0.00	0.00	0.00
			False	0.50	1.00	0.67



Gambar 4 Hasil Confusion Matrix Hyperparameter Kedua

Pada dataset pertama, dilakukan pengujian terhadap hyperparameter ketiga untuk dataset pertama epoch dengan nilai 50, 75, dan 100. Pengujian dilakukan menggunakan konfigurasi optimizer SGD dengan initial learning rate (lr_0) sebesar 0,01. Berdasarkan hasil pengujian, konfigurasi terbaik diperoleh pada epoch ke-50. Rincian hasil pengujian disajikan pada Tabel 4. Visualisasi performa pelatihan model ditunjukkan pada Gambar 5 yang memperlihatkan grafik loss dan akurasi

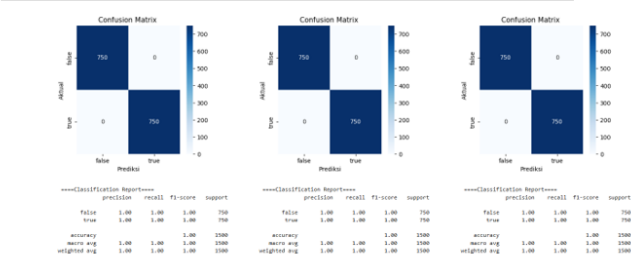
selama proses training dan validasi. Gambar 6 menampilkan hasil confusion matrix yang memberikan gambaran akurasi klasifikasi model terhadap masing-masing kelas pada data uji.



Gambar 5 Visualisasi Performa Pelatihan Model Hyperparameter Ketiga

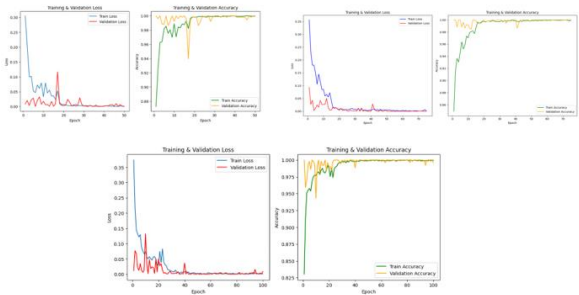
Tabel 4
 Hasil Hyperparameter Ketiga

Epoch	Learning Rate	Optimizer	label	Precision	Recall	F1-score
50	0.001	SGD	True	1.00	1.00	1.00
			False	1.00	1.00	1.00
75			True	1.00	1.00	1.00
			False	1.00	1.00	1.00
100			True	1.00	1.00	1.00
			False	1.00	1.00	1.00



Gambar 6 Hasil Hyperparameter Ketiga Pada dataset pertama, dilakukan pengujian terhadap hyperparameter keempat untuk dataset pertama epoch dengan nilai 50, 75, dan 100. Pengujian dilakukan menggunakan konfigurasi optimizer ADAM dengan initial learning rate (lr_0) sebesar 0,001. Berdasarkan hasil pengujian, konfigurasi terbaik diperoleh pada epoch ke-50. Rincian hasil pengujian disajikan pada Tabel 4. Visualisasi performa pelatihan model ditunjukkan pada Gambar 7 yang memperlihatkan grafik loss dan akurasi selama proses training dan validasi. Gambar 8

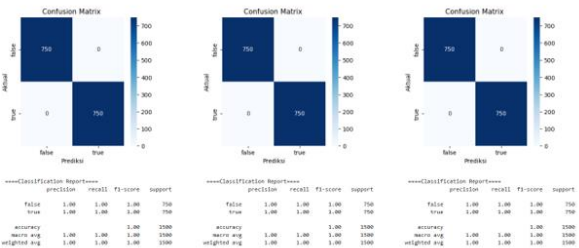
menampilkan hasil confusion matrix yang memberikan gambaran akurasi klasifikasi model terhadap masing-masing kelas pada data uji.



Gambar 7 Visualisasi Performa Pelatihan Model Hyperparameter Keempat

Tabel 5
Hasil Hyperparameter Keempat

Epoch	Learning Rate	Optimizer	label	Precision	Recall	F1-score
50	0.001	SGD	True	1.00	1.00	1.00
			False	1.00	1.00	1.00
75			True	1.00	1.00	1.00
			False	1.00	1.00	1.00
100			True	1.00	1.00	1.00
			False	1.00	1.00	1.00



Gambar 8 Hasil Hyperparameter Keempat
Berdasarkan hasil pengujian terhadap empat kombinasi hyperparameter pada dataset pertama, dapat disimpulkan bahwa pemilihan optimizer dan learning rate sangat memengaruhi performa model. Pada pengujian pertama yang menggunakan optimizer SGD dengan learning rate 0,01, hasil terbaik diperoleh pada epoch ke-100 dengan nilai F1-score sebesar 0,98 untuk kedua kelas, yang menunjukkan performa cukup stabil dan meningkat seiring bertambahnya epoch. Sebaliknya, pada

pengujian kedua yang menggunakan optimizer Adam dengan learning rate yang sama (0,01), model gagal mengenali kelas True pada seluruh epoch, dan hanya menunjukkan performa sedang untuk kelas False dengan F1-score sebesar 0,67. Hal ini menandakan bahwa konfigurasi Adam dengan learning rate tinggi kurang cocok untuk dataset ini. Pada pengujian ketiga dan keempat, baik SGD maupun Adam dengan learning rate 0,001 mampu memberikan hasil sempurna dengan F1-score sebesar 1,00 di semua epoch dan untuk kedua kelas. Ini menunjukkan bahwa penggunaan learning rate yang lebih kecil secara signifikan meningkatkan performa model, serta memperlihatkan bahwa kedua optimizer dapat bekerja dengan baik jika dikonfigurasi dengan benar. Dengan demikian, dapat disimpulkan bahwa konfigurasi learning rate 0,001, baik dengan SGD maupun Adam, merupakan pilihan terbaik untuk model ini.

4. KESIMPULAN

Berdasarkan hasil pengujian terhadap kombinasi hyperparameter pada dataset pertama, dapat disimpulkan bahwa pemilihan optimizer dan learning rate sangat berpengaruh terhadap performa model. Penggunaan optimizer SGD dengan learning rate sebesar 0,01 menunjukkan hasil terbaik pada epoch ke-100, dengan nilai F1-score mencapai 0,98 untuk kedua kelas (True dan False), yang menunjukkan bahwa model mampu belajar lebih baik seiring bertambahnya jumlah epoch. Sebaliknya, penggunaan optimizer Adam dengan learning rate yang sama menghasilkan performa yang buruk, di mana model tidak mampu mengenali kelas True sama sekali dan hanya memberikan hasil sedang pada kelas False dengan F1-score sebesar 0,67. Hasil ini mengindikasikan bahwa learning rate yang terlalu tinggi tidak sesuai digunakan bersama optimizer Adam pada dataset ini. Selanjutnya, pengujian dengan learning rate yang lebih kecil, yaitu 0,001, baik menggunakan SGD

maupun Adam, memberikan hasil sempurna dengan nilai Precision, Recall, dan F1-score sebesar 1,00 pada seluruh epoch dan kedua kelas. Ini menunjukkan bahwa penurunan learning rate secara signifikan meningkatkan kestabilan dan akurasi model. Dengan demikian, dapat disimpulkan bahwa konfigurasi terbaik dalam pelatihan model ini adalah penggunaan learning rate 0,001, baik dengan optimizer SGD maupun Adam, karena mampu memberikan performa optimal dan konsisten.

Ucapan Terima Kasih

Terima Kasih kepada Universitas Multi Data Palembang yang telah memberikan kesempatan bagi kami untuk mengembangkan diri serta memperdalam ilmu kami dibidang Informatika, dan juga Terima kasih kepada Bapak M. Ezar Al Rivan, M.Kom. selaku Dosen pembimbing.

5. DAFTAR PUSTAKA

- A. Santoso, "Implementasi Deep Learning berbasis Keras untuk Pengenalan Wajah," Skripsi, Program Studi Informatika, Fakultas Komunikasi dan Informatika, Universitas Muhammadiyah Surakarta, 2018.
- Arum, P. R., Dzeaulfath, M., & Winarta, A. O. (2023). One-Shot Learning Menggunakan Siamese Neural Network Untuk Pendeteksian Wajah. *Prosiding Seminar Nasional Sains Data*, 3(1), 290–295. <https://doi.org/10.33005/senada.v3i1.125>
- Ayu, D. W., & Pradipta, G. A. (2024). SqueezeNet Feature Extraction dan Gradient Boosting untuk Klasifikasi Penyakit Monkeypox pada Citra Kulit. *Jurnal Sistem dan Informatika (JSI)*, 18(2), 177–183. <https://doi.org/10.30864/jsi.v18i2.612>
- Chen, S., Liu, Y., Gao, X., & Han, Z. (2021). MobileFace: Lightweight Deep Face Recognition for Mobile Devices. *IEEE Access*, 9, 50148–50156. <https://doi.org/10.1109/ACCESS.2021.3068640>
- Fitriani, N., Setiawan, I. A., & Saputra, R. (2022). Penerapan Siamese Neural Network untuk Verifikasi Wajah pada Sistem Presensi Mahasiswa. *Jurnal Teknologi dan Sistem Komputer*, 10(2), 165–172.
- G. Koch, R. Zemel, dan R. Salakhutdinov, "Detection of Ovulation," *British Medical Journal*, vol. 2, no. 5108, p. 1355, 2015, doi: 10.1136/bmj.2.5108.1355-c.
- He, M., Liang, D., Li, X., & Li, W. (2020). An Efficient Deep Siamese Network for Face Verification on Resource-Constrained Devices. *Computers, Materials & Continua*, 63(1), 23–39. <https://doi.org/10.32604/cmc.2020.010473>
- Howard, A. et al. (2021). Searching for MobileNetV3. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11), 4152–4162.
- Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., & Keutzer, K. (2016). SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size. 1–13. <http://arxiv.org/abs/1602.07360>
- Kartarina, & Imam, H. (2019). Deep Learning Verifikasi Kemiripan Wajah Dengan Arsitektur Jaringan Siamese. *Jurnal Bumigora Information Technology (BITE)*, 1(2), 116–125. <https://doi.org/10.30812/bite.v1i2.605>
- Kumar, P., & Singh, R. (2022). Deep Siamese Neural Networks for Face Verification in Unconstrained Environments. *Neural Computing and Applications*, 34(12), 9453–9467.
- M. Zufar, "Convolutional Neural Networks untuk Pengenalan Wajah secara Real-Time," Skripsi, Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Teknologi Sepuluh Nopember, Surabaya, 2016.
- Putra, Y. R., Nugroho, A. S., & Hartanto, R. (2022). Optimasi Face Recognition Berbasis Siamese Network dengan Pendekatan Few-Shot Learning. *Jurnal Ilmu Komputer dan Informatika (JIKI)*, 9(1), 45–53.
- Rahman, M. M., Barua, S., & Ahmed, F. (2022). Siamese Neural Network-Based Face Verification System for Real-Time Applications. *Sensors*, 22(10), 3763. <https://doi.org/10.3390/s22103763>

- Ranjan, R., Sankaranarayanan, S., Bansal, A., Xu, H., Chen, J.-C., Castillo, C. D., & Chellappa, R. (2021). A Fast and Accurate System for Face Recognition in Unconstrained Images. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 3(1), 1–12.
- Santos, M. S., Valadao, C. T., Resende, C. Z., & Cavalieri, D. C. (2024). Predicting diabetic retinopathy stage using Siamese Convolutional Neural Network. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging and Visualization*, 12(1).
<https://doi.org/10.1080/21681163.2023.2297017>
- Sari, R. P., & Hartati, S. (2021). Siamese Neural Network untuk Sistem Absensi Otomatis Berbasis Wajah. *Jurnal Teknologi Informasi dan Ilmu Komputer*, 8(3), 511–518.
- Sriyati, A. Setyanto, dan E. T. Luthfi, “Literature Review: Pengenalan Wajah Menggunakan Algoritma Convolutional Neural Network,” *Jurnal TIKomSiN*, vol. 8, no. 2, pp. 63–70, Okt. 2020.
- Sun, Y., Wang, X., & Tang, X. (2020). Deep Learning Face Representation by Joint Identification-Verification. *International Journal of Computer Vision*, 128, 1175–1193. <https://doi.org/10.1007/s11263-019-01247-5>
- Suwindra, M. T., Ernawati, & Erlansari, A. (2021). Analisis Kemiripan Jenis Burung Menggunakan Siamese Neural Network Analysis of Bird Species Similarity Using Siamese Neural Network. *Jurnal Rekursif*, 9(2).
<http://ejournal.unib.ac.id/index.php/rekursif/193>
- Wang, T., Du, H., Li, X., & Zhang, S. (2022). Lightweight Face Verification via Siamese Network and Contrastive Learning. *Journal of Visual Communication and Image Representation*, 85, 103572.
- Wang, Z., Zhu, Z., Li, Y., & Gong, M. (2021). A comprehensive survey on face recognition technologies. *Information Fusion*, 72, 1–35.
<https://doi.org/10.1016/j.inffus.2021.01.004>
- Y. Hartiwi, E. Rasywir, Y. Pratama, dan P. A. Jusia, “Eksperimen Pengenalan Wajah dengan Fitur Indoor Positioning System menggunakan Algoritma CNN,” *Paradigma – Jurnal Informatika dan Komputer*, vol. 22, no. 2, pp. 248–255, Sep. 2020.
- Zhang, W., Qi, C., & Li, P. (2021). Lightweight Siamese Network for Real-Time Face Verification on Mobile Devices. *IEEE Access*, 9, 135876–135885.
- Zulkifli, I. M. R. (2023). KLIK: Kajian Ilmiah Informatika dan Komputer Image Classification pada Kasus American Sign Language Menggunakan Support Vector Machine. *Media Online*, 4(2), 1184–1191.
<https://doi.org/10.30865/klik.v4i2.1242>