

Penerapan Metode CNN Pada Sistem Pendeteksi Arah Gerak Mata Untuk Optimalisasi *Smart Door Lock*

Application of the CNN Method in an Eye Movement Detection System to Optimize Smart Door Lock

Hendra Kurniawan¹, Ika Karlina Laila Nur Suciningtyas²

^{1,2}Program Studi Teknologi Rekayasa Elektronika Teknik Elektro Politeknik Negeri Batam
^{1,2}Jl. Ahmad Yani, Tlk. Tering, Kec. Batam Kota, Kota Batam, Kepulauan Riau 29461.
telp. [+62 778 469860](tel:+62778469860)

hendrakurniawann46@gmail.com¹, ikakarlina@polibatam.ac.id²

Abstrak – Kemajuan teknologi telah membawa inovasi dalam berbagai bidang, termasuk sistem keamanan. Salah satu kemudahan yang diharapkan adalah kemampuan untuk mengakses pintu secara otomatis tanpa perlu membawa kunci fisik. Penelitian ini menggunakan Convolutional Neural Network (CNN) untuk mendeteksi gerakan arah mata sebagai metode autentikasi pada smart door lock. Sistem ini dirancang untuk mendeteksi gerakan mata, mengimplementasikan metode tersebut dalam sistem kunci pintar, dan mengevaluasi kinerjanya dari segi keandalan dan responsivitas. Metode penelitian meliputi pengumpulan data gambar mata, pra-pemrosesan citra, pelatihan model CNN, serta pengujiannya. Hasil pelatihan menunjukkan akurasi tinggi pada dataset validasi, namun pengujian sistem dalam kondisi nyata masih menghadapi beberapa tantangan. Hasil penelitian menunjukkan bahwa pencahayaan sangat memengaruhi performa model prediksi. Pada 40 Lux, akurasi tercatat sebesar 83,33%, meningkat menjadi 88,71% pada 60 Lux, dan mencapai 93,33% pada 80 dan 100 Lux, dengan performa terbaik di 120 Lux (akurasinya 98,33%, presisi 100%, recall 93,33%). Namun, pengujian pada jarak 4 cm dengan pencahayaan 120 Lux hanya mencapai akurasi 65%, menunjukkan jarak deteksi memengaruhi kinerja model karena perbedaan dengan dataset awal. Sistem Open Lock dan Close Lock berhasil dalam 27 dari 30 pengujian, dengan waktu rata-rata 17-18 detik, meski terdapat 3 kegagalan akibat kondisi yang tidak terpenuhi. Sistem Close Lock melalui sinyal pushbutton selalu berhasil dengan waktu eksekusi 1 detik, mencatat keberhasilan keseluruhan 90%. Hasil ini menunjukkan bahwa sistem belum sepenuhnya optimal karena performa masih bergantung pada pencahayaan dan posisi pengguna. Pengembangan lebih lanjut diperlukan untuk meningkatkan jarak deteksi, adaptasi pencahayaan, dan kecepatan eksekusi agar sistem lebih akurat.

Kata Kunci: Convolution Neural Network, Smart Door lock, Deteksi Mata

Abstract – Technological advancements have brought innovations across various fields, including security systems. One of the conveniences anticipated is the ability to access doors automatically without the need to carry physical keys. This study employs Convolutional Neural Network (CNN) to detect eye movement direction as an authentication method for a smart door lock system. The system is designed to detect eye movements, implement this method in a smart lock system, and evaluate its reliability and responsiveness. The research methods include eye image data collection, image preprocessing, CNN model training, and testing. The training results showed high accuracy on the validation dataset, but the system's performance in real-world conditions still faces several challenges. The study results indicate that lighting significantly

affects the model's prediction performance. At 40 Lux, accuracy was recorded at 83.33%, increasing to 88.71% at 60 Lux, and reaching 93.33% at 80 and 100 Lux, with the best performance achieved at 120 Lux (accuracy: 98.33%, precision: 100%, recall: 93.33%). However, testing at a 4 cm distance under 120 Lux lighting only achieved 65% accuracy, indicating that detection distance affects the model's performance due to discrepancies with the initial dataset. The Open Lock and Close Lock systems succeeded in 27 out of 30 tests, with an average execution time of 17-18 seconds, despite 3 failures caused by unmet conditions. The Close Lock system using a pushbutton signal consistently succeeded with a 1-second execution time, achieving an overall success rate of 90%. These findings indicate that the system is not yet fully optimized, as its performance still relies on lighting and user positioning. Further development is needed to enhance detection distance, adapt to various lighting conditions, and improve execution speed, ensuring a more accurate and responsive system.

Keywords: Convolutional Neural Network, Smart Door Lock, Eye Detection.

1. Pendahuluan

Kemajuan teknologi dalam berbagai bidang telah membawa perubahan signifikan, termasuk dalam sektor sistem keamanan. Salah satu kebutuhan yang semakin meningkat adalah sistem akses pintu otomatis yang tidak memerlukan kunci fisik. Konsep tradisional penguncian pintu menggunakan kunci logam telah berkembang seiring dengan adanya teknologi modern, seperti penggunaan kode PIN dan RFID. Meskipun teknologi-teknologi ini menawarkan kemudahan, masih terdapat ketergantungan pada perangkat fisik yang harus dibawa pengguna, yang mengurangi efisiensi [1], [2].

Seiring dengan kebutuhan akan solusi yang lebih praktis, sistem identifikasi berbasis biometrik mulai diperhitungkan, mengingat ciri khas unik setiap individu seperti sidik jari dan telapak tangan yang sulit untuk diduplikasi. Namun, tantangan muncul ketika tangan pengguna dalam kondisi tertentu seperti basah atau berminyak, yang dapat mengganggu akurasi sistem biometrik. Selain itu, penggunaan sistem ini secara massal dapat menimbulkan risiko penyebaran virus, yang menjadi kendala utama dalam penggunaannya [3], [4]. Sebagai alternatif, teknologi pengenalan wajah menawarkan solusi yang lebih aman dan tanpa kontak fisik. Meskipun demikian, teknologi ini memungkinkan terjadinya pembobolan melalui foto atau model 3D wajah pengguna [5].

Pengolahan citra menggunakan teknik machine learning telah berkembang dengan berbagai metode seperti *Convolutional Neural Network* (CNN), *Artificial Neural Network* (ANN), *K-Nearest Neighbors* (KNN), dan *Support Vector Machine* (SVM). Di antara metode-metode tersebut, CNN terbukti unggul dalam berbagai tugas pengolahan dan klasifikasi citra. Sebuah studi perbandingan menunjukkan bahwa CNN mencapai performa terbaik dalam klasifikasi citra dengan akurasi 0,942, presisi 0,943, recall 0,942, dan F1 Score 0,942 [6]. Sementara itu, ANN menunjukkan tingkat akurasi 90% [7]. Penggunaan CNN untuk mendeteksi citra gerakan mata juga memberikan hasil yang menjanjikan. Penelitian sebelumnya mencatat akurasi sebesar 95% dalam mendeteksi arah pandangan mata [8]. Selain itu, teknologi ini memiliki potensi untuk meningkatkan aksesibilitas, seperti pada kursi roda pintar yang memanfaatkan gerakan mata untuk navigasi menu, dengan tingkat akurasi mencapai 96% [9].

CNN telah terbukti sangat efektif dalam menganalisis pola visual yang kompleks, seperti pengenalan wajah dan gerakan tubuh. Dengan menggunakan gambar sebagai input, algoritma ini mampu mengidentifikasi dan membedakan berbagai objek dalam sebuah gambar, sehingga sering digunakan untuk membantu mesin mengenali dan membedakan satu gambar dari yang lain [10]. CNN mampu menangani berbagai variasi dalam gambar, seperti perubahan ukuran, rotasi, dan pergeseran, yang membuatnya sangat berguna untuk aplikasi keamanan [11].

Mengingat keberhasilan teknologi CNN dalam berbagai aplikasi, penelitian ini bertujuan untuk mengembangkan sistem autentikasi berbasis gerakan mata yang dapat diterapkan pada *smart door lock*. Sistem yang ingin dikembangkan diharapkan bisa memberikan solusi yang lebih aman dan baik dibandingkan dengan metode autentikasi konvensional, seperti penggunaan kunci fisik atau sidik jari. Dengan memanfaatkan keunikan pola gerakan mata, sistem ini tidak hanya dapat meningkatkan tingkat keamanan, tetapi juga menawarkan kemudahan akses yang lebih baik

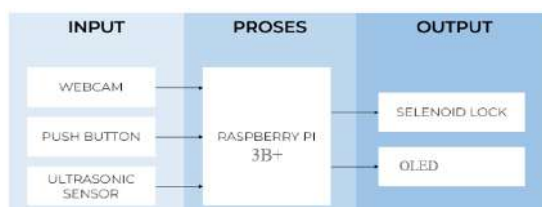
tanpa memerlukan kontak fisik. terutama bagi penyandang disabilitas yang memiliki keterbatasan fisik.

2. Metode Penelitian

Metode penelitian pada alat meliputi perancangan, pengumpulan dan pengolahan dataset, inialisasi model CNN, integrasi model dengan perangkat lunak dan perangkat keras, serta pengujian fungsi dan akurasi.

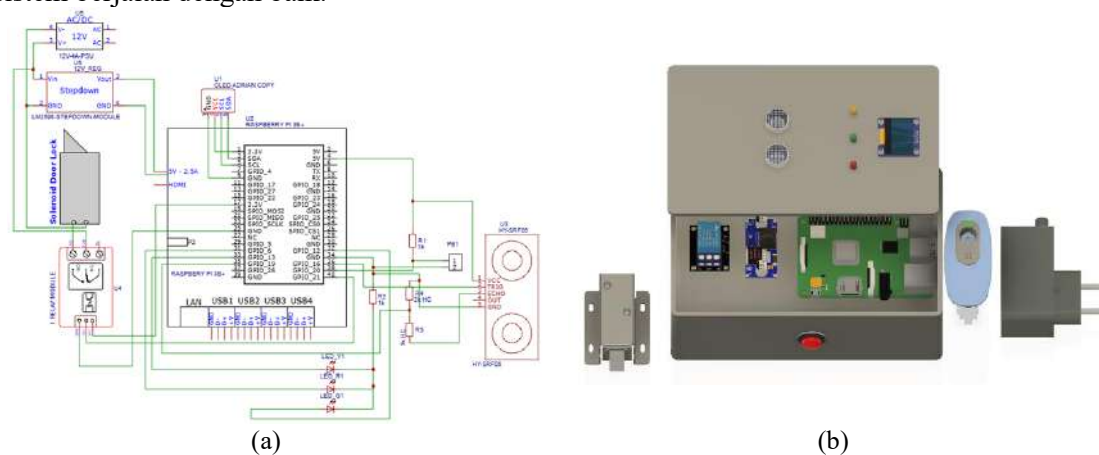
2.1. Perancangan Sistem

Perancangan sistem deteksi gerakan mata untuk optimalisasi *smart door lock* dimulai dengan penyusunan diagram blok, seperti yang ditampilkan pada Gambar 1. Diagram ini menggambarkan rancangan sistem yang terdiri dari tiga bagian utama *input*, *processing*, dan *output*. Pada bagian *input*, sistem memanfaatkan tiga komponen utama, yaitu webcam, *push button*, dan sensor ultrasonik. Raspberry Pi Model 3B+ digunakan sebagai *controller* untuk memproses data yang diterima dari *input*. Sementara itu, bagian *output* terdiri dari dua komponen utama, yang pertama solenoid lock, yang berfungsi untuk membuka dan menutup kunci, dan kedua layar OLED, yang digunakan untuk menampilkan status sistem secara *real-time*.



Gambar 1. Diagram Blok Sistem

Setelah menetapkan diagram blok, selanjutnya dibuat desain *schematic* dari sistem yang dirancang. Desain *schematic* pada Gambar 2a ini menampilkan semua komponen yang digunakan dalam sistem. Setiap komponen dihubungkan sesuai fungsinya untuk memastikan sistem berjalan dengan baik.

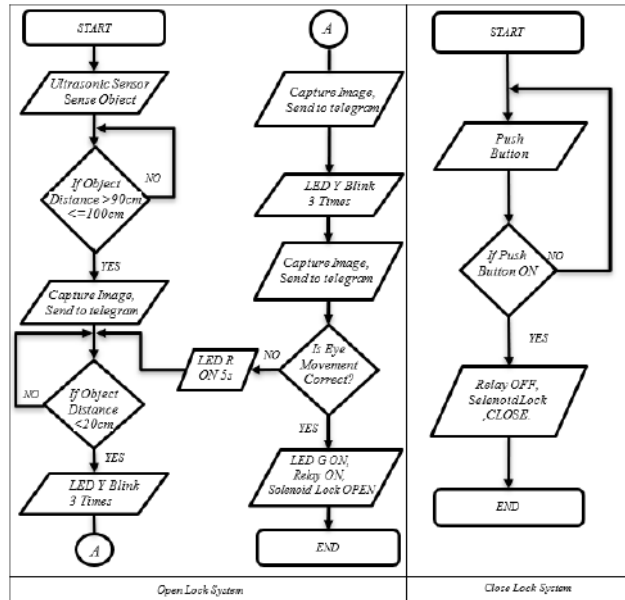


Gambar 2. Desain *Schematic* (a) dan mekanik (b)

Seluruh komponen seperti raspberry Pi, relay dan modul *step down* dirancang pada sebuah kotak yang memiliki ukuran 180x110x65mm, rancangan dapat dilihat pada Gambar 2b. Bagian *cover* dipasang webcam, sensor ultrasonik dan OLED untuk memungkinkan akses yang mudah dan visualisasi status sistem, serta *push button* yang terletak pada sisi kotak.

Proses kerja sistem atau *flowchart* dari alat ini dapat dilihat pada Gambar 3. *Flowchart* sistem ini terdiri dari dua bagian utama: *open lock system* dan *close lock system*. *Open lock system* adalah sistem utama dalam penelitian ini. Sistem *open lock* akan aktif ketika sensor ultrasonik mendeteksi keberadaan objek. Selanjutnya, webcam akan mengambil gambar dari kondisi mata pengguna sebanyak dua kali yang bertujuan agar sistem lebih aman. Setelah citra ini ditangkap, Raspberry Pi akan memprosesnya. Jika pola gerakan mata yang dikenali sesuai

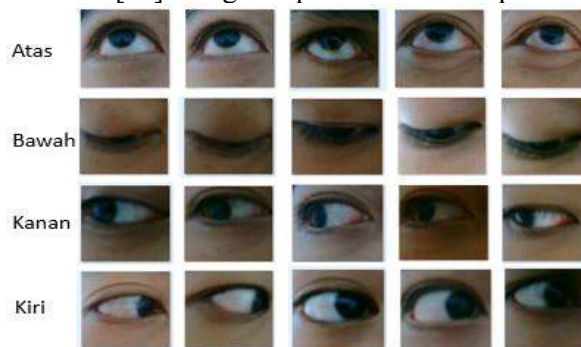
dengan yang telah ditetapkan, Raspberry Pi akan mengaktifkan relay yang mengendalikan *solenoid lock* untuk membuka pintu. Sementara itu, sistem *close lock* diaktifkan dengan menekan *push button*. Aksi ini akan mengirimkan sinyal ke Raspberry Pi untuk menonaktifkan relay, sehingga *solenoid lock* akan menutup pintu kembali. Sistem *close lock* akan bekerja jika sistem *open lock* berhasil. Seluruh status sistem, termasuk pembukaan dan penutupan pintu, akan ditampilkan secara pada OLED untuk memberikan informasi kepada pengguna.



Gambar 3. Flowchart Sistem

2.1. Persiapan Dataset

Diperlukan dataset yang memadai dan bervariasi untuk mencapai tingkat akurasi yang optimal dalam sistem deteksi gerakan mata menggunakan CNN[12]. Dalam penelitian ini, tujuan utama adalah untuk mendeteksi 4 arah pandang mata, yaitu: kanan, kiri, bawah, dan atas. Dataset menggunakan citra dari mata kanan yang berjumlah 1000 citra untuk setiap arah pandang mata diambil dari jarak 2 CM. Total dataset mencapai 4000 gambar dari 5 orang partisipan yang disimpan di google drive agar mempermudah proses pelatihan. Dataset akan dibagi menjadi dua bagian yang digunakan untuk tahap pelatihan model (*training*) dan untuk tahap pengujian model (*testing*). Penggunaan banyaknya data tanpa didukung oleh kualitas gambar yang baik sangat berdampak pada tingkat akurasi[13]. Pengelompokan data set dapat dilihat pada Gambar 4.

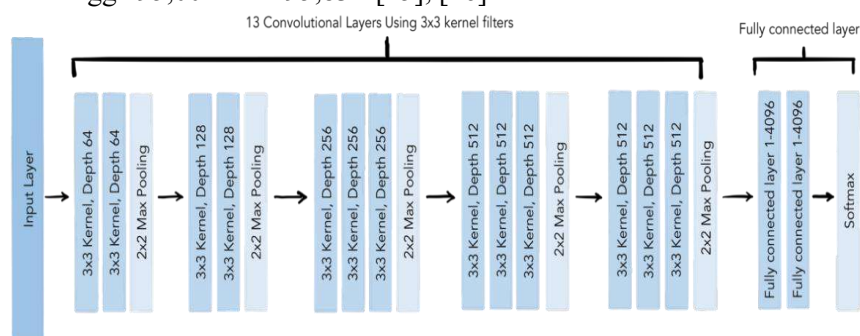


Gambar 4. Data Set

2.2. Pelatihan Model CNN

Model CNN yang digunakan pada pelatihan model kali ini adalah VGG16. VGG adalah singkatan dari Visual Geometric Group, sebuah arsitektur jaringan yang dikembangkan oleh Universitas Oxford, struktur dari model dapat dilihat pada Gambar 7. VGG16 merujuk pada versi model ini yang terdiri dari 16 lapisan dalam strukturnya[14]. Penelitian sebelumnya yang

menggunakan arsitektur VGG16 untuk keperluan identifikasi dan klasifikasi menghasilkan tingkat akurasi hingga 95,00% dan 95,83%[15], [16].



Gambar 5. Arsitektur VGG16

Pelatihan dilakukan untuk membuat dan memastikan bahwa model yang akan digunakan mampu mengidentifikasi objek yang ditargetkan[17]. Pelatihan model dilakukan di Google Colab yang terhubung dengan google drive menggunakan perangkat PC. Google Colab menyediakan akses ke GPU gratis, yang dapat mempercepat proses pelatihan model secara signifikan[12]. Performa model selama pelatihan dipantau menggunakan fungsi loss untuk menilai kemampuan mendeteksi data baru. Setelah selesai, model disimpan dalam format TensorFlow Lite (.tflite) untuk digunakan di Raspberry Pi. TensorFlow Lite adalah pustaka machine learning berformat ringan yang dirancang untuk mendeteksi dan mengenali objek dengan kebutuhan penyimpanan yang minim[18], [19].

2.3. Metode Pengujian

Sebagai upaya untuk merancang pengujian keefektifan dan kinerja alat pengenalan gerakan mata sebagai kunci pintar, terdapat beberapa kriteria penting yang perlu diperhatikan:

a. Pengujian Komponen Alat

Pengujian komponen dilakukan untuk memastikan setiap bagian perangkat keras dalam sistem dapat berfungsi sesuai dengan spesifikasi yang telah dirancang. Proses pengujian ini menggunakan metode *black box testing*, yang berfokus pada keluaran (output) dari perangkat keras berdasarkan masukan (input) yang diberikan[20].

b. Ketahanan Kinerja Sistem Alat

Pengujian kinerja sistem dibagi menjadi 3:

1. Kondisi Pencahayaan

Pengujian dilakukan diberbagai kondisi pencahayaan dari 40 Lux, 60 Lux, 80 Lux, 100 Lux, dan 120 Lux sebanyak 30 kali per kondisi cahaya. Tujuan dari pengujian ini adalah untuk mengevaluasi sejauh mana sistem dapat mempertahankan performa pengenalan gerakan mata pada berbagai tingkat pencahayaan dengan kondisi nyata.

2. Jarak

Pengujian dilakukan pada jarak 4cm yang mana jarak tersebut berbeda dengan jarak dari data training. Proses ini dilakukan sebanyak 30 kali menggunakan level cahaya yang menghasilkan akurasi tinggi bertujuan untuk mengevaluasi apakah jarak memengaruhi akurasi deteksi model yang telah dilatih. Hasil dari pengujian ini akan memberikan informasi penting tentang kemampuan sistem mendeteksi gerakan mata.

3. Sistem *Open Lock* dan *Close Lock*

Pengujian sistem *Open Lock* dan *Close Lock* dilakukan menggunakan metode *black box testing* untuk memastikan seluruh proses pada Gambar 5 berjalan sesuai rancangan. Pengujian memanfaatkan jarak dan level pencahayaan optimal serta melibatkan dua partisipan berbeda dari dataset awal guna menguji generalisasi alat. Partisipan pertama menggunakan input arah Kiri-Kanan, sedangkan partisipan kedua menggunakan arah Atas-Bawah. Selain itu, pengujian kecepatan sistem dilakukan untuk mengukur waktu yang diperlukan dari pengambilan citra hingga aktivasi solenoid lock. Proses ini diulang untuk menghitung rata-

rata waktu dan mengevaluasi konsistensi kinerja sistem. Persentase keberhasilan dihitung menggunakan persamaan (1)

$$\% = \frac{\text{Total Berhasil}}{\text{Total Uji}} \times 100\% \quad (1)$$

3. Hasil dan Pembahasan

3.1. Hasil Rancangan Alat

Hasil rancangan alat dapat dilihat pada Gambar 6. Alat ini dirancang mempertimbangkan fungsi utama, yaitu membuka dan menutup solenoid. Selain itu, rancangan ini juga dilengkapi dengan indikator LED dan sensor ultrasonik untuk memastikan sistem alat yang sedang berjalan.



Gambar 6. Hasil Rancangan Alat

3.2. Hasil Pengujian Komponen Alat

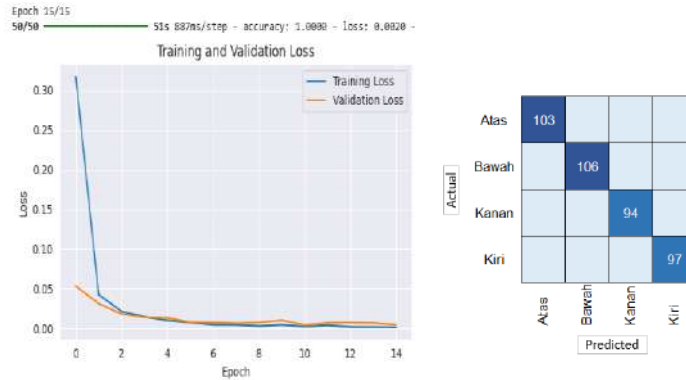
Setiap komponen diuji secara terpisah untuk memastikan tidak ada gangguan antar komponen. Pengujian komponen perangkat keras menunjukkan hasil yang sesuai dengan ekspektasi output. Webcam berhasil menangkap dan menyimpan gambar, sementara sensor ultrasonik mendeteksi jarak dengan akurasi baik (4 cm hingga 10 cm). *Push button* memberikan respon tepat untuk mengaktifkan *solenoid lock*, didukung oleh relay yang bekerja optimal. OLED menampilkan teks dengan jelas, dan LED menyala sesuai sinyal dari Raspberry Pi, yang terbukti andal dalam memproses data dan mengontrol seluruh perangkat keras. Hasil ini mengindikasikan bahwa sistem berjalan dengan stabil dan sesuai desain. Hasil pengujian dapat dilihat di Tabel 1.

Tabel 1. Pengujian Komponen

No	Komponen	Input	Ekspektasi Output	Keterangan
1	Webcam	Gambar di depan kamera	Gambar berhasil ditangkap dan tersimpan	Webcam berfungsi normal
2	Sensor ultrasonik	Jarak(cm) 4,5,6,7,8,9,10	Mendapatkan nilai jarak(cm) 4,5,6,7,8,9,10	Sensor mendeteksi jarak
3	<i>Push button</i>	Tombol ditekan	Tegangan terhubung	<i>Push button</i> berfungsi normal
4	Relay	Sinyal dari Raspberry Pi	Relay aktif	Relay berfungsi normal
5	<i>Solenoid Lock</i>	Tegangan	<i>Solenoid Open</i> dan <i>Close</i>	Tegangan di control relay
7	OLED	Teks dari Raspberry Pi	OLED menampilkan teks	Tampilan jelas dan sesuai
8	LED <i>Yellow(Y)</i> , <i>Green(G)</i> , <i>Red(R)</i>	Sinyal dari Raspberry Pi	LED menyala	LED berfungsi normal.
9	Raspberry Pi	Perintah pemrosesan data	Menjalankan fungsi kontrol seluruh perangkat keras	Seluruh proses terpenuhi

3.3. Hasil Pelatihan Model CNN

Proses pelatihan model menghasilkan performa yang baik selama proses pelatihan. Setelah melalui 15 epoch, tingkat *loss* yang dicapai adalah 0,0020, mengindikasikan bahwa kesalahan prediksi model terhadap data pelatihan sangat kecil. Grafik *loss* dapat dilihat pada Gambar 7.



Gambar 7. Grafik dan Matrix Training

Analisa hasil dari metrik evaluasi menunjukkan bahwa model memiliki kemampuan untuk memahami hubungan kompleks antar fitur dalam data pelatihan dengan sangat baik dengan tingkat akurasi pelatihan juga mencapai 100%, menunjukkan bahwa model mampu mengenali pola data pelatihan dengan sempurna tanpa adanya kesalahan klasifikasi. Hal ini terlihat dari kombinasi loss yang sangat rendah dan akurasi yang sempurna, yang mencerminkan kecocokan ideal antara prediksi model dan label sebenarnya pada data pelatihan.

3.4. Hasil Pengujian Sistem

Hasil dari pengujian ditampilkan menggunakan *Confusion Matrix*, yang merupakan metode untuk mengevaluasi kinerja model klasifikasi. *Confusion Matrix* memberikan informasi tentang jumlah deteksi untuk setiap kelas, serta apakah deteksi tersebut benar atau salah[21]. Kinerja algoritma klasifikasi dalam *confusion matrix* dapat dianalisis dengan metrik kinerja yang melibatkan istilah-istilah seperti TP, FP, TN, FN, *Precision*, *Recall*, dan *Accuracy*[22]. *Precision* dihitung menggunakan persamaan (2). Hasil dari *Recall* dapat dianalisa menggunakan persamaan (3). Presentase *Accuracy* dihitung menggunakan persamaan (4).

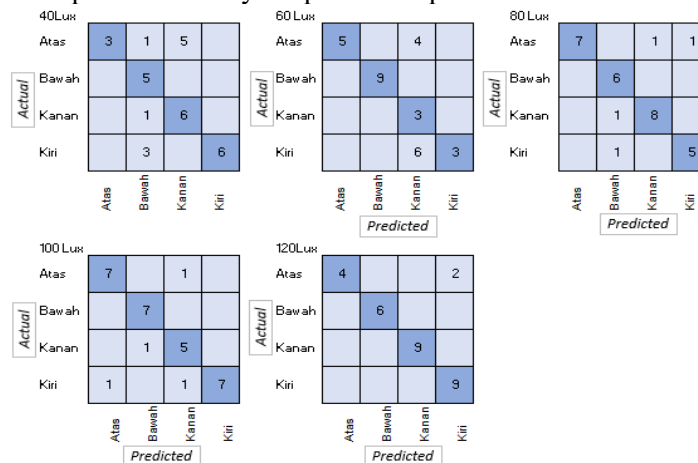
$$Precision = \frac{TP}{TP+FP} \times 100\% \tag{2}$$

$$Recall = \frac{TP}{TP+FN} \times 100\% \tag{3}$$

$$Accuracy = \frac{TN+TP}{TN+TP+FN+FP} \times 100\% \tag{4}$$

3.4.1. Pengujian Kondisi Pencahayaan

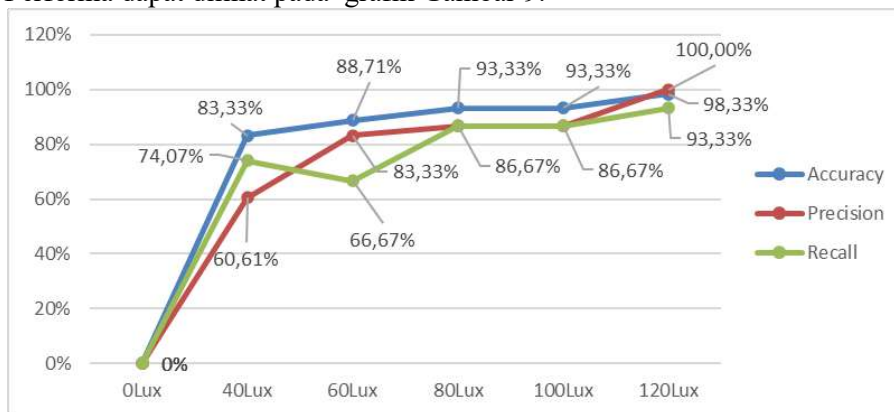
Hasil pengujian terhadap kondisi cahaya dapat dilihat pada Gambar 8.



Gambar 8. Hasil Pengujian 40Lux, 60Lux, 80Lux, 100 Lux, 120 Lux

Hasil penelitian menunjukkan bahwa tingkat pencahayaan memiliki pengaruh signifikan terhadap performa model dalam melakukan prediksi. Pada tingkat pencahayaan 40 Lux, model

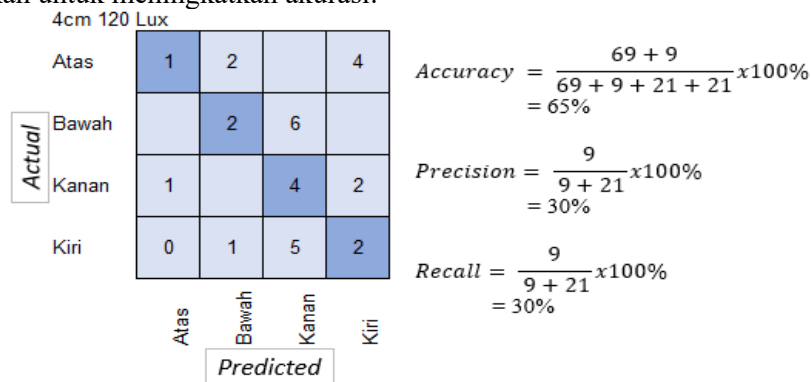
mencapai akurasi sebesar 83,33%, dengan presisi 60,61% dan *recall* 74,07%. Peningkatan pencahayaan ke 60 Lux meningkatkan akurasi menjadi 88,71%, presisi 83,33%, dan *recall* 66,67%. Selanjutnya, pada pencahayaan 80 Lux dan 100 Lux, akurasi yang dicapai adalah 93,33%, dengan presisi dan *recall* masing-masing sebesar 86,67%. Performa terbaik model dicapai pada tingkat pencahayaan 120 Lux, dengan akurasi 98,33%, presisi 100%, dan *recall* 93,33%. Hasil ini mengindikasikan bahwa tingkat pencahayaan optimal untuk performa maksimal model adalah 120 Lux. Performa dapat dilihat pada grafik Gambar 9.



Gambar 9. Grafik Accuracy, Precision, dan Recall

3.4.2. Pengujian Jarak

Pengujian pada jarak 4 cm dengan pencahayaan 120 Lux menghasilkan akurasi 65%, presisi 30%, dan *recall* 30%, analisa dapat dilihat pada Gambar 10. Meskipun pencahayaan sudah optimal, deteksi yang benar hanya 9 dari 30 percobaan, menunjukkan bahwa jarak memengaruhi kinerja sistem. Hal ini kemungkinan disebabkan oleh perbedaan jarak deteksi dengan dataset awal, yang mengurangi kualitas gambar yang diproses model. Penyesuaian jarak deteksi sesuai dengan dataset diperlukan untuk meningkatkan akurasi.



Gambar 10. Hasil Pengujian Jarak

3.4.3. Pengujian Sistem Open Lock dan Close Lock

Hasil pengujian sistem pada Tabel.2 menunjukkan bahwa sistem *Open Lock* dan *Close Lock* berhasil dijalankan. Sistem ini berhasil dalam 27 dari 30 pengujian, dengan waktu rata-rata 17-18 detik. Namun, terdapat 3 kegagalan yang terjadi (uji ke-2, ke-17, ke-24), di mana *output* tidak sesuai ekspektasi dan membuat kondisi tidak terpenuhi untuk mengaktifkan sistem *Close Lock*.

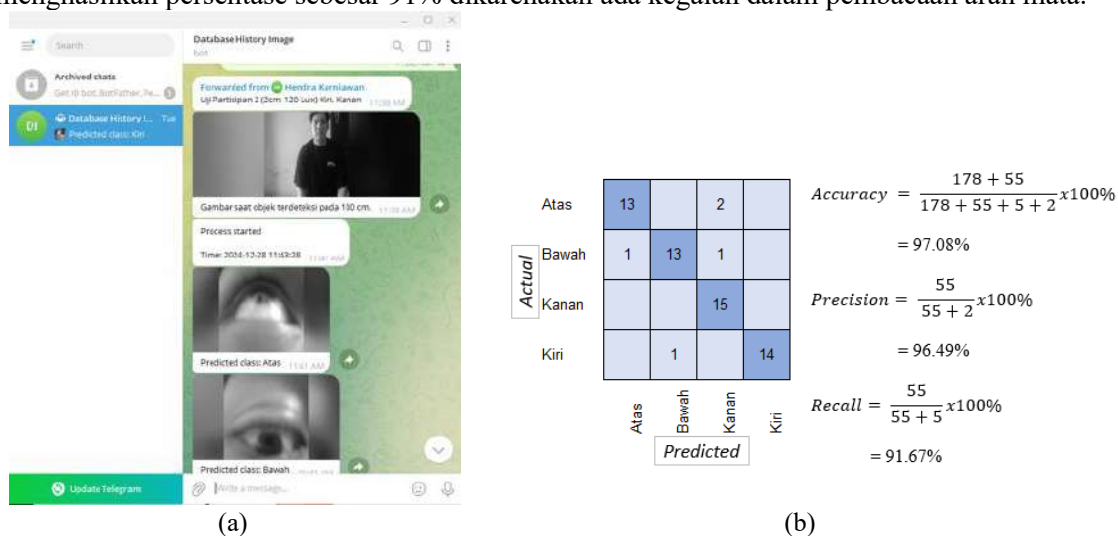
Tabel 2. Hasil Pengujian Sistem *Open Lock* dan *Close Lock*

No	Sistem yang Diuji	Input	Ekspektasi Output	Hasil	Kecepatan (Detik)	Keterangan
1	<i>Open Lock</i>	Jarak, Mata Arah Kiri & Kanan	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
	<i>Close Lock</i>	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
2	<i>Open Lock</i>	Jarak, Mata Arah Kiri & Kanan	<i>Solenoid Open</i>	Gagal	-	Salah deteksi arah mata
	<i>Close Lock</i>	Menekan <i>Push button</i>	<i>Solenoid Close</i>	-	-	Kondisi tidak terpenuhi
3	<i>Open Lock</i>	Jarak, Mata Arah Kiri & Kanan	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi

No	Sistem yang Diuji	Input	Ekspektasi Output	Hasil	Kecepatan (Detik)	Keterangan
4	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Kiri & Kanan	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
5	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Kiri & Kanan	<i>Solenoid Open</i>	Berhasil	18	Seluruh proses terpenuhi
6	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Kiri & Kanan	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
7	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Kiri & Kanan	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
8	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Kiri & Kanan	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
9	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Kiri & Kanan	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
10	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Kiri & Kanan	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
11	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Kiri & Kanan	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
12	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Kiri & Kanan	<i>Solenoid Open</i>	Berhasil	18	Seluruh proses terpenuhi
13	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Kiri & Kanan	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
14	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Kiri & Kanan	<i>Solenoid Open</i>	Berhasil	18	Seluruh proses terpenuhi
15	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Kiri & Kanan	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
16	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Atas & Bawah	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
17	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Gagal	-	Salah deteksi arah mata
	Open Lock	Jarak, Mata Arah Atas & Bawah	<i>Solenoid Open</i>	-	-	Kondisi tidak terpenuhi
18	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Atas & Bawah	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
19	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Atas & Bawah	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
20	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Atas & Bawah	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
21	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Atas & Bawah	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
22	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Atas & Bawah	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
23	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Atas & Bawah	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
24	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Gagal	-	Salah deteksi arah mata
	Open Lock	Jarak, Mata Arah Atas & Bawah	<i>Solenoid Open</i>	-	-	Kondisi tidak terpenuhi
25	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Atas & Bawah	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
26	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Atas & Bawah	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
27	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Atas & Bawah	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
28	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Atas & Bawah	<i>Solenoid Open</i>	Berhasil	18	Seluruh proses terpenuhi
29	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Atas & Bawah	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi
30	Close Lock	Menekan <i>Push button</i>	<i>Solenoid Close</i>	Berhasil	1	Seluruh proses terpenuhi
	Open Lock	Jarak, Mata Arah Atas & Bawah	<i>Solenoid Open</i>	Berhasil	17	Seluruh proses terpenuhi

Data dikirim ke telegram untuk memudahkan analisa penelitian, seperti pada Gambar 11a. Sistem *Close Lock* dengan masukan sinyal dari *pushbutton* selalu berhasil jika seluruh proses terpenuhi, dengan keluaran relay aktif dan *solenoid lock* tertutup, serta waktu eksekusi yang konsisten dalam 1 detik. Presentase keberhasilan sistem secara keseluruhan sebesar 90%. Hasil

analisa dari akurasi dan presisi pada Gambar 11b menunjukkan hasil persentase diatas 95%, *recall* menghasilkan persentase sebesar 91% dikarenakan ada kegalan dalam pembacaan arah mata.



Gambar 11. Data Telegram (a) dan Akurasi, Presisi, dan *Recall* Sistem *Open Lock* dan *Close Lock* (b)

4. Kesimpulan

Sistem yang dikembangkan memiliki beberapa keunggulan dan keterbatasan. Komponen perangkat keras pada alat berfungsi dengan baik, model CNN yang digunakan menunjukkan hasil pelatihan yang sangat bagus, dengan akurasi mencapai 100%. Namun, performa prediksi masih dipengaruhi oleh pencahayaan dan jarak. Tingkat pencahayaan terbaik untuk hasil maksimal adalah 120 Lux, dengan akurasi 98,33%, presisi 100%, dan recall 93,33%. Pengujian pada jarak 4 cm menunjukkan akurasi rendah karena jarak tersebut tidak sesuai dengan dataset awal. Selain itu, waktu kerja rata-rata sistem *Open Lock* dan *Close Lock* yang memerlukan 17–18 detik untuk membuka kunci tergolong lambat. Sistem *Open Lock* dan *Close Lock* belum sepenuhnya optimal bekerja dengan tingkat keberhasilan 90% karena terdapat beberapa kegagalan akibat kondisi yang tidak terpenuhi. Data penelitian yang hanya melibatkan lima partisipan membatasi generalisasi, sementara posisi dan jarak mata terhadap kamera memengaruhi kenyamanan dan akurasi. Sistem lebih andal mendeteksi gerakan mata ke arah kiri-kanan dibandingkan atas-bawah, meskipun gerakan mata yang relatif cepat dan posisi pengguna yang tidak ideal terhadap kamera sering kali menyebabkan kegagalan deteksi. Selain itu, sistem masih mendeteksi objek lain selain mata dan kurang optimal untuk pengguna dengan kondisi fisik tertentu atau yang memakai kacamata. Pengembangan lebih lanjut dapat difokuskan pada peningkatan jarak deteksi dan kondisi pencahayaan agar respon sistem alat lebih cepat dengan waktu eksekusi lebih efisien dan akurat diberbagai kondisi nyata, serta memberi kemudahan akses yang lebih baik tanpa memerlukan kontak fisik terutama bagi penyandang disabilitas.

Referensi

- [1] I. Karlina, L. N. Suciningtyas, M. Safry, A. Mubarrok, and A. Jefiza, "Kunci Portable Berbasis IoT Menggunakan QR Code, Android dan REST API," *Jurnal ASEECT*, vol. 2, no. 2, 2021.
- [2] I. P. Ihsan, S. Buwarda, H. Novianty, and I. A. Putra, "Voice Recognition Untuk Otomatisasi Sistem Pengakses Pintu," *JSAI: Journal Scientific and Applied Informatics*, vol. 4, no. 01, 2021, doi: 10.36085.
- [3] N. Mustakim, N. Hossain, M. Mustafizur Rahman, N. Islam, Z. Hossain Sayem, and M. Asaduz Zaman Mamun, "Face Recognition System Based on Raspberry Pi Platform," *1st International Conference on Advances in Science, Engineering and Robotics Technology 2019 (ICASERT 2019)*, 2019.

- [4] R. Mishra, A. Ransingh, M. K. Behera, and S. Chakravarty, "Convolutional Neural Network Based Smart Door Lock System," *Proceedings - 2020 IEEE India Council International Subsections Conference, INDISCON 2020*, pp. 151–156, Oct. 2020, doi: 10.1109/INDISCON50162.2020.00041.
- [5] T. Sutrisna and J. Carina, "Kurir yang Gelapkan MacBook Rp 67 Juta Pakai Gambar Wajah 3D untuk Aktifkan Akun Ojol," Kompas.com. Accessed: Jul. 14, 2024. [Online]. Available: <https://megapolitan.kompas.com/read/2021/11/24/15530661/kurir-yang-gelapkan-macbook-rp-67-juta-pakai-gambar-wajah-3d-untuk?page=all>
- [6] M. F. Naufal, "Analisi Perbandingan Algoritma SVM, KNN, Dan CNN Untuk Klasifikasi Citra Cuaca," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, vol. 8, no. 2, pp. 311–318, 2021, doi: 10.25126/jtiik.202184553.
- [7] F. Fathurrahman, M. M. Santoni, and A. Muliawati, "Penerapan Artificial Neural Network Untuk Klasifikasi Citra Teks Dalam Penerjemahan Bahasa Daerah," *Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)*, pp. 585–594, 2020.
- [8] M. Amin Nurdin, R. Cahya Wihandika, and F. Utaminingrum, "Deteksi Pergerakan Arah Mata menggunakan Convolution Neural Network berdasarkan Facial Landmark," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 4, no. 10, pp. 3338–3345, 2020, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [9] B. Sheldo, P. Laksono, and F. Utaminingrum, "Navigasi Menu Berdasarkan Arah Pandangan Mata pada Kursi Roda Pintar menggunakan Fusion-CNN berbasis Jetson TX2," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 6, no. 12, pp. 5601–5607, 2022, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [10] D. Marcella, Y. Yohannes, and S. Devella, "Klasifikasi Penyakit Mata Menggunakan Convolutional Neural Network Dengan Arsitektur VGG-19," *Jurnal Algoritme*, vol. 3, no. 1, pp. 60–70, Oct. 2022, doi: 10.35957/algoritme.v3i1.3331.
- [11] A. R. Syafeeza, M. K. Mohd Fitri Alif, Y. Nursyifaa Athirah, A. S. Jaafar, A. H. Norihan, and M. S. Saleha, "IoT based facial recognition door access control home security system using raspberry pi," *International Journal of Power Electronics and Drive Systems*, vol. 11, no. 1, pp. 417–424, Mar. 2020, doi: 10.11591/ijpeds.v11.i1.pp417-424.
- [12] E. Hartati and Verdy, "Klasifikasi Penyakit Mata Menggunakan Convolutional Neural Network Model RESNET-50," *Jurnal Rekayasa Sistem Informasi dan Teknologi*, vol. 1, no. 3, pp. 199–206, 2024.
- [13] P. Adi Nugroho, I. Fenriana, and R. Arijanto, "Implementasi Deep Learning Menggunakan Convolutional Neural Network (CNN) Pada Ekspresi Manusia," *Jurnal ALGOR*, vol. 2, no. 1, 2020, [Online]. Available: <https://jurnal.buddhidharma.ac.id/index.php/algor/index>
- [14] R. E. Wiratna, A. Sahputro, B. Danendra, and E. Y. Puspaningrum, "Pengenalan Karakter Hijaiyah Menggunakan Metode Convolutional Neural Network," *SCAN*, vol. 17, no. 1, pp. 14–18, Feb. 2022.
- [15] Rismiyati and A. Luthfiarta, "Klasifikasi Kualitas Buah Salak dengan Transfer Learning Arsitektur VGG16," *Jurnal Informatika dan Teknologi Informasi*, vol. 18, no. 1, pp. 37–48, 2021, doi: 10.31515/telematika.v18i1.4025.
- [16] R. Windiawan, A. Suharso, and S. Artikel, "Identifikasi Penyakit pada Daun Kopi Menggunakan Metode Deep Learning VGG16," *Jurnal Keilmuan Dan Aplikasi Teknik Informatika*, vol. 13, no. 2, pp. 9–16, Sep. 2021, doi: 10.35891/explorit.
- [17] N. Fadlia and R. Kosasih, "Klasifikasi Jenis Kendaraan Menggunakan Metode Convolutional Neural Network (CNN)," *Jurnal Ilmiah Teknologi dan Rekayasa*, vol. 24, no. 3, pp. 207–215, 2019, doi: 10.35760/tr.2019.v24i3.2397.
- [18] M. Bagus Baihaqi, Y. Litanianda, and A. Triyanto, "Implementasi Tensor Flow Lite Pada Teachable Untuk Identifikasi Tanaman Aglonema Berbasis Android," : *Jurnal Teknik Universitas Muhammadiyah Ponorogo*, vol. 6, no. 1, pp. 70–80, 2022, doi: 10.24269/jkt.v6i1.1143.

- [19] R. Roslidar, M. R. Syahputra, R. Muharar, and F. Arnia, “Adaptasi Model CNN Terlatih pada Aplikasi Bergerak untuk Klasifikasi Citra Termal Payudara,” *Jurnal Rekayasa Elektrika*, vol. 18, no. 3, Sep. 2022, doi: 10.17529/jre.v18i3.8754.
- [20] B. Hardika *et al.*, “Pengujian Blackbox Testing Website Garuda Farm Menggunakan Teknik Equivalence Partitioning,” *Jurnal Kridatama Sains Dan Teknologi*, vol. 06, no. 02, pp. 740–753, 2024.
- [21] U. Sri Rahmadhani and N. Lysbetti Marpaung, “Klasifikasi Jamur Berdasarkan Genus Dengan Menggunakan Metode CNN,” *Jurnal Informatika: Jurnal pengembangan IT (JPIT)*, vol. 8, no. 2, 2023.
- [22] S. Kothari, S. Chiwhane, S. Jain, and M. Baghel, “Cancerous brain tumor detection using hybrid deep learning framework,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 27, no. 1, pp. 1651–1661, Jun. 2022, doi: 10.11591/ijeecs.v26.i3.pp1651-1661.