

Analisis Face Recognition untuk Sistem Absensi Department of Informatics Universitas PGRI Madiun

Andhika Dwiky Pratama

Program Studi Teknik Informatika, Universitas PGRI Madiun
email: andhika_2005101100@unipma.ac.id

Abstract: As time progresses, attendance systems have evolved from manual paper-based methods to systems integrated with all student academic information. The attendance recording for student association meetings in the Informatics Engineering Department at PGRI Madiun University has been deemed inefficient, prompting the need for a face recognition technology that can reduce the time required compared to manual attendance using paper. Face recognition is a biometric method, similar to fingerprints and retinal scans, where the image captured by a camera is compared to photos and facial features stored in a database. This attendance system uses a confusion matrix to test the face recognition model. The results of the face recognition model are highly accurate, capable of detecting registered members' faces, as evidenced by the model's accuracy reaching 100%. The use of CNN architecture is an appropriate choice in designing an accurate face recognition model. The analysis of this model is a crucial step, as the results of the face recognition model can later be implemented on platforms such as websites or applications to address the shortcomings of conventional paper-based attendance processes. The implementation of this system is also an effective measure in preventing fraud in the attendance process.

Keywords: Face recognition, Machine learning, Python, Confussion Matrix

Abstrak: Seiring perkembangan zaman, sistem presensi mulai beralih dari manual menggunakan kertas menjadi sistem yang terintegrasi dengan semua informasi akademik mahasiswa. Pencatatan kehadiran rapat himpunan mahasiswa jurusan Teknik Informatika di Universitas PGRI Madiun dinilai kurang efektif sehingga proses pencatatan kehadiran perlu menggunakan teknologi pengenalan wajah dapat mengurangi waktu yang dibutuhkan dibandingkan dengan absensi manual yang dilakukan di atas kertas. Pengenalan wajah adalah salah satu metode biometrik, serupa dengan sidik jari dan retina mata, di mana gambar yang diambil oleh kamera dibandingkan dengan foto dan ciri-ciri wajah yang telah tersimpan dalam database. Sistem absensi ini menggunakan *confussion matrix* sebagai pengujian model pengenalan wajah. Hasil dari model pengenalan wajah menghasilkan model yang akurat, dengan dapat mendeteksi wajah anggota yang telah terdaftar, hal ini dibuktikan dengan hasil akurasi model yang mencapai 100%. Penggunaan arsitektur CNN merupakan pilihan yang tepat dalam merancang sebuah model pengenalan wajah yang akurat. Analisis model ini merupakan langkah yang tepat, yang mana nantinya hasil dari model pengenalan wajah dapat di implementasikan dalam sebuah *platform* seperti *website* atau aplikasi dalam mengatasi kekurangan yang ada pada proses absensi yang masih konvensional menggunakan kertas. Implementasi dari sistem ini juga merupakan langkah yang tepat dalam mengatasi kecurangan dalam proses absensi.

Kata kunci: Face recognition, Machine learning, Python, Confussion Matrix

Pendahuluan

Seiring perkembangan zaman, sistem presensi mulai beralih dari manual menggunakan kertas menjadi sistem yang terintegrasi dengan semua informasi akademik mahasiswa. Tren terkait sistem presensi yang sedang berkembang adalah sistem absensi berbasis *cloud* yang dapat diakses melalui internet dan data kehadiran dapat diakses dari mana saja dan kapan saja (Darmawan, 2022). Dengan adanya pencatatan kehadiran rapat himpunan mahasiswa yang telah menggunakan teknologi komputer, nantinya akan mempermudah proses pencatatan kehadiran, tentu cara seperti itu dinilai kurang efektif.

Proses pencatatan kehadiran dengan menggunakan teknologi pengenalan wajah dapat mengurangi waktu yang dibutuhkan dibandingkan dengan absensi manual yang dilakukan di atas kertas. Pengenalan wajah adalah salah satu metode biometrik, serupa dengan sidik jari dan retina mata, di mana gambar yang diambil oleh kamera dibandingkan dengan foto dan ciri-ciri wajah yang telah tersimpan dalam database (Munawir et al., 2020). Sistem ini merupakan bentuk kecerdasan buatan yang mampu

mengenali atau mengidentifikasi wajah manusia dari citra digital, baik gambar maupun video, dengan cara mendeteksi, mengenali, dan membandingkan wajah yang belum dikenal dengan data wajah yang sudah ada dalam database (Utomo et al., 2020).

Untuk melihat akurasi nantinya akan menggunakan salah satu metode *Convolutional Neural Network* (CNN). *Convolutional Neural Network* (CNN) adalah salah satu jenis *deep learning* yang berhubungan dengan pengenalan pola mulai dari pengolahan citra, pengenalan objek, hingga pengenalan wajah (Syarif et al., 2023). Pada dasarnya, algoritma *Convolutional Neural Network* (CNN) adalah arsitektur jaringan saraf tiruan yang sangat efektif untuk klasifikasi gambar. Inti dari CNN terletak pada operasi konvolusinya, di mana setiap fitur dari sebuah gambar diekstraksi untuk membentuk beberapa pola yang lebih mudah diklasifikasikan (Arsal et al., 2020).

Pada penelitian (Arsal et al., 2020), yang meneliti tentang *face recognition* untuk akses pegawai bank menggunakan *deep learning* dengan metode *Convolutional Neural Network* (CNN) dapat dikatakan berhasil dengan tingkat akurasi sebesar 95% dengan menggunakan metode perhitungan persentase keberhasilannya. Sedangkan pada penelitian (Munawir et al., 2020), yang meneliti tentang implementasi *face recognition* pada absensi kehadiran mahasiswa menggunakan metode *haar cascade classifier* dapat dikatakan berhasil namun dengan tingkat akurasi yang cukup rendah yaitu pada data tahap pelatihan model dapat menebak dengan benar 19 dari 25 kali percobaan, sedangkan pada tahap pengujian model dapat menebak 2 dari 6 kali percobaan, metode pengujian akurasi menggunakan metode perhitungan persentase keberhasilan.

Berdasarkan pada 2 penelitian tersebut penulis ingin menguji penggunaan metode *Convolutional Neural Network* (CNN) pada sistem absensi mahasiswa. Penulis ingin menguji apakah penggunaan *Convolutional Neural Network* (CNN) pada sistem absensi mahasiswa adalah metode yang lebih tepat dibandingkan dengan metode *haar cascade classifier*. Penelitian ini menggunakan data citra yang diambil dari anggota himpunan mahasiswa program studi teknik informatika.

Metode

Machine Learning

Machine Learning, merupakan proses membuat komputer untuk memodifikasi atau mengadaptasi tindakan seperti: membuat prediksi atau mengendalikan sebuah robot, sehingga tindakan yang dilakukan akan lebih akurat (Marsland, 2014).

Machine learning dibagi menjadi tiga jenis yaitu; *supervised learning*, *unsupervised learning*, dan *reinforcement learning*. *Supervised learning* merupakan jenis *machine learning* yang metode pengklasifikasiannya dilakukan dengan memberi label untuk mengklasifikasikan kelas-kelas yang belum dikenali. *Unsupervised learning* biasanya disebut dengan *clustering* karena tidak memerlukan label untuk pengklasifikasian datanya, dan hasilnya juga tidak berbentuk prediksi label. *Reinforcement learning* bekerja dalam lingkungan yang sangat dinamis dimana konsep dari *Reinforcement Learning* harus menyelesaikan sebuah tugas tanpa adanya perintah secara eksplisit mengenai tujuan yang harus dicapai (Roihan et al., 2020).

Local Binary Pattern

Local Binary Pattern atau disingkat LBP merupakan operator yang digunakan untuk mendeskripsikan tekstur dengan membandingkan nilai ukuran piksel di sekitarnya. Proses ini dilakukan dengan menghitung selisih nilai berdasarkan ambang batas tertentu, yang kemudian menghasilkan kode biner untuk setiap piksel (Utami et al., 2021).

Berikut adalah formula perhitungan dari LBP yang digunakan dalam pemrosesan citra untuk analisis tekstur menurut (Hosny et al., 2021):

$$LBP_{1,J}(P_c) = \sum_{j=0}^{J-1} G(P_j - P_c)2^j, \quad (1)$$

Dimana:

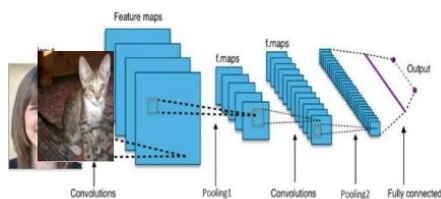
$$G(m) = \begin{cases} 0, & m < 0 \\ 1, & otherwise \end{cases} \quad (2)$$

Convolutional Neural Network

Convolutional Neural Network atau disingkat CNN adalah pengembangan dari *multi-layer neural network*. CNN paling sering digunakan untuk analisis gambar digital (Dandi Darajat et al., 2021). CNN merupakan algoritma *deep learning* berfungsi untuk memproses data dari gambar-gambar, menentukan

kepentingan pada berbagai aspek dalam gambar, dan untuk membedakan objek satu dengan objek lainnya yang terdapat dalam sebuah gambar (Azmi et al., 2023).

Pada umumnya, CNN terdiri dari beberapa layer yaitu *normalization*, *pooling*, *convolution*, *fully connected layer*. *Normalization* termasuk dalam tahap *pre-processing* yang digunakan untuk mempersiapkan data yang akan digunakan menjadi format tertentu. Tahap tersebut dilakukan untuk mempermudah proses klasifikasi (Nugroho & Husin, 2022). *Pooling layer* adalah proses pengurangan dimensi dari *feature* sehingga dapat mempercepat pemrosesan mengingat parameter yang digunakan semakin sedikit untuk diperbarui (Nurjannah et al., 2022). Arsitektur *Convolutional Neural Network* (CNN) menurut Badri et al. (2023) dapat dilihat pada Gambar 1.



Gambar 1. Arsitektur CNN (Badri et al., 2023)

Pada *convolutional layer* adalah proses untuk model dalam mengambil *input* berupa gambar yang di dapat dari pengekstrakan pada proses *pooling layer* dengan cara menghitung setiap koordinat indeks perkalian pada gambar untuk menghasilkan *array 2 dimensi* (Badri et al., 2023). Sedangkan pada proses akhir yaitu *fully connected layer* bertujuan untuk merubah dimensi data sehingga data dapat di klasifikasikan secara linear. Pada proses ini nilai matrix yang didapatkan pada proses sebelumnya akan di transformasikan menjadi vektor (Nurhadi & Pumomo, 2022).

Commented [JM1]: Format nama gambar silahkn disesuaikan dengan template. (Semua gambar pada naskah ini).

Confusion Matrix

Confusion matrix merupakan metode yang paling sering digunakan untuk menghitung akurasi pada konsep *data mining*. *Confusion matrix* digambarkan dengan label mengelompokkan pernyataan jumlah data uji yang benar dan data uji yang salah (Priyadi, 2023).

Confusion matrix biasa digunakan untuk melakukan pengukuran pada suatu *classifier* pada saat melakukan prediksi dari kelas yang berbeda. *Confusion matrix* juga dapat menilai *recall*, dan *precision* (Jamhari et al., 2020).

Secara matematis, tingkat akurasi dihitung dengan formula pada persamaan berikut:

$$Akurasi = \frac{TP+TN}{TP+FP+TN+FN} \times 100\% \quad (3)$$

Sedangkan, secara matematis *precision* dan *recall* berturut-turut dihitung menggunakan persamaan berikut:

$$precision = \frac{TP}{(TP+FP)} \quad (4)$$

$$recall = \frac{TP}{(TP+FN)} \quad (5)$$

Keterangan:

TP = True Positive

FP = False Positive

TN = True Negative

TP = True Positive

Metode Pengembangan Sistem

Pengumpulan dan pre-processing data citra

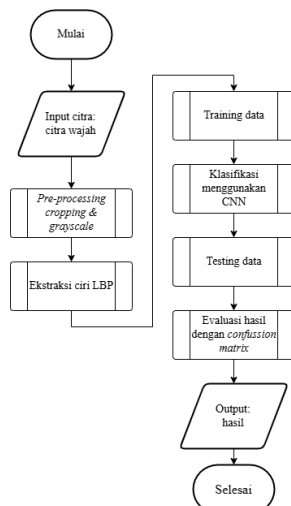
Pengumpulan *dataset* sudah menggunakan proses automasi dimana penulis sudah membuat sistem yang otomatis menangkap 50 data citra dari tiap mahasiswa. Pada sistem ini juga sudah melakukan automasi dalam proses *cropping* data sehingga semua data citra yang sudah diambil akan otomatis *cropping* yang terfokus pada bagian wajah. Tidak hanya itu, pada sistem ini juga sudah melakukan

proses *labelling* data sehingga semua data akan tersimpan secara rapi yang di grup berdasarkan nama mahasiswa.

Feature extraction dengan local binary pattern

Karena *feature extraction* yang digunakan adalah LBP maka tiap data citra yang memiliki format *Red, Green, Blue* (RGB) akan di konversi kedalam *grayscale*. Kemudian untuk mengekstraksi fitur penting dalam data citra, data akan dikonversikan menjadi array 3x3 yang kemudian nilai tengah piksel tersebut akan menjadi ambang batas. Kemudian, semua nilai array akan dibandingkan dengan nilai ambang batas, apabila melebihi ambang batas akan diberi nilai 1, sedangkan apabila tidak melebihi ambang batas akan diberi nilai 0.

Hasil akhir dari ekstraksi fitur akan berupa histogram. Dimana satu data citra memiliki ciri sebanyak 256 nilai yang merupakan frekuensi kemunculan dari nilai 0 sampai dengan 255. Tentunya hasil yang lebih akan didapatkan apabila array yang digunakan sebesar 5x5, dst. Tahapan pengembangan model dapat dilihat pada Gambar 2.



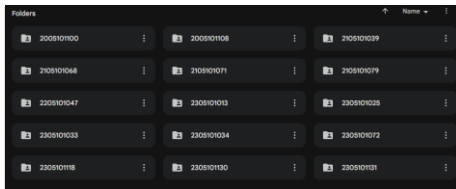
Gambar 2. Tahapan pengembangan model

Evaluasi model

Evaluasi model akan menggunakan *confusion matrix*. Apabila hasil dari perancangan model masih belum memenuhi kriteria, maka akan dilakukan pengevaluasian terhadap model CNN. Namun, apabila hasil yang dihasilkan masih belum optimal. Maka, akan dilakukan ulang tahapan pemrosesan data dengan mempertimbangkan beberapa pertimbangan, seperti pencahayaan, dan juga ekspresi wajah.

Hasil dan Pembahasan

Pengumpulan data citra merupakan langkah awal yang sangat penting dalam pengembangan model, karena kualitas dan pengaturan data yang baik akan mempengaruhi hasil akhir dari model tersebut. Setelah data citra berhasil dikumpulkan, data citra akan disusun kedalam folder-folder berdasarkan NIM mahasiswa untuk memastikan pengelolaan yang rapi dan sistematis. Penyusunan data citra dapat dilihat pada Gambar 3 sebagai berikut.



Gambar 3. Kumpulan folder data citra

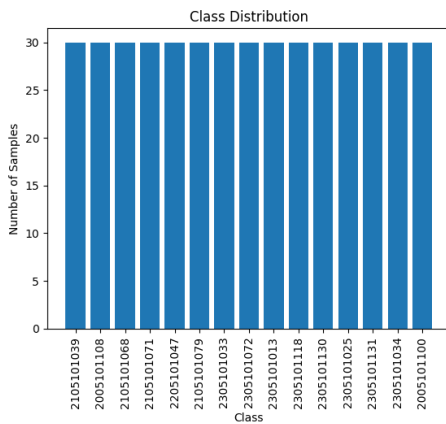
Pada Gambar 3 merupakan cara penulis dalam menyimpan data citra yang akan digunakan dalam pengembangan model. Setiap NIM akan diwakilkan oleh 1 folder dan setiap folder berisi 30 data citra yang sudah di *crop* pada bagian wajah, sehingga bagian-bagian lain seperti *background* atau bagian tubuh lainnya sudah tidak termasuk dalam data tersebut.

Commented [JM2]: Sebelum gambar silahkan diberikan narasi, tidak langsung gambar.

```
def load_images(data_dir, image_size=(128, 128)):
    images = []
    labels = []
    for label_dir in os.listdir(data_dir):
        label_path = os.path.join(data_dir, label_dir)
        if os.path.isdir(label_path):
            for image_name in os.listdir(label_path):
                image_path = os.path.join(label_path, image_name)
                image = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)
                image = cv2.resize(image, image_size)
                image = np.expand_dims(image, axis=-1)
                image = np.repeat(image, 3, axis=-1) # Convert grayscale to RGB
                images.append(image)
                labels.append(label_dir)
    return np.array(images), np.array(labels)
```

Gambar 4. Tahap pengumpulan data citra menjadi dataset

Pada Gambar 4 merupakan cara untuk melabeli tiap-tiap data. Hal ini dilakukan mengingat banyaknya data yang harus dilabeli. Total mahasiswa yang ada pada data tersebut sebanyak 15 orang yang dimana keseluruhan data citra sebanyak 450. Maka dari itu, pelabelan data akan dilakukan secara otomatis. Selain untuk memberi label pada data, pada kode tersebut data akan di *pre-process*. Seperti contoh, data tersebut akan di konversi menjadi *grayscale* dan ukuran data citra akan dibuat menjadi 120x120 piksel.



Gambar 5. Distribusi kelas data citra

Gambar 5 merupakan cara untuk mengetahui distribusi kelas pada data citra dapat dilakukan proses visualisasi untuk mengetahui, sebaran data tiap-tiap data citra dan dapat dipastikan bahwa sebaran data citra yang penulis gunakan sudah terdistribusi secara baik.

```
label_encoder = LabelEncoder()
labels_encoded = label_encoder.fit_transform(labels)
num_classes = len(label_encoder.classes_)

X_train, X_test, y_train, y_test = train_test_split(images, labels_encoded, test_size=0.2, random_state=42)
y_train = to_categorical(y_train, num_classes=num_classes)
y_test = to_categorical(y_test, num_classes=num_classes)
```

Gambar 6. Proses pemisahan data latihan dan data uji

Proses selanjutnya pada Gambar 6 yakni melakukan *split data*. Sebaran data yang digunakan untuk training sebanyak 80% dari total data keseluruhan. Jika total data yang ada sebanyak 450, maka

sebanyak 360 data citra dari 15 orang anggota akan digunakan sebagai data latih dalam proses untuk mengenali anggota-anggota. Sedangkan untuk data uji, sebanyak 20% dari total data keseluruhan, maka didapatkan sebanyak 90 data citra digunakan sebagai data uji dalam memvalidasi apakah model dapat mengetahui data ini termasuk kedalam data anggota milik siapa.

```
from keras.models import Model
from keras.layers import Input, Dense, Dropout
from keras.optimizers import Adam
from keras.utils import to_categorical

X_train_emb = facenet.embeddings(X_train)
X_test_emb = facenet.embeddings(X_test)

input_shape = X_train_emb.shape[1]
```

Gambar 7. Proses *embedding* data latih dan data uji

Langkah terakhir sebelum data akan di latih dan di uji dalam model adalah dengan melakukan konversi dari data yang semula berbentuk gambar akan di rubah menjadi kumpulan angka. Proses ini dinamakan *embedding*. Pada Gambar 7 data latih dan data uji dilakukan proses *embedding*.

```
Model: "model"
-----
Layer (type)                Output Shape         Param #
-----
input_2 (InputLayer)        [(None, 512)]        0
dense (Dense)                (None, 128)          65664
dropout (Dropout)           (None, 128)          0
dense_1 (Dense)              (None, 15)           1935
-----
Total params: 67599 (264.06 KB)
Trainable params: 67599 (264.06 KB)
Non-trainable params: 0 (0.00 Byte)
```

Gambar 8. Arsitektur model

Gambar 8 merupakan arsitektur model yang akan digunakan untuk melatih dan menguji model pengenalan wajah. Arsitektur yang digunakan merupakan arsitektur *Facenet*. *Facenet* mengambil *input* berupa data citra wajah dan akan menghasilkan *output* berupa 128 nilai vektor yang sudah dibuat pada Gambar 7.

Hasil Pengembangan Sistem

Pada penelitian ini dilakukan evaluasi terhadap model *face recognition* yang telah dikembangkan. Proses evaluasi ini dilakukan dengan menggunakan beberapa metrik yakni *accuracy*, *precision*, *recall*, dan *F1 Score* yang berguna untuk mengukur seberapa baik model dalam mengklasifikasikan data uji.

```
Accuracy: 1.0
Precision: 1.0
Recall: 1.0
F1 Score: 1.0
```

Gambar 9. Hasil *accuracy*, *precision*, *recall*, dan *F1 Score* model

Hasil evaluasi model *face recognition* pada Gambar 9 menunjukkan performa yang sangat baik dengan semua metrik evaluasi mencapai nilai maksimal. *Accuracy* yang didapatkan sebesar 1.00 atau 100% mengindikasikan bahwa model mampu mengklasifikasikan seluruh data uji dengan benar tanpa adanya kesalahan. *Precision* sebesar 1.00 atau 100% menunjukkan bahwa setiap prediksi positif yang dibuat oleh model adalah benar, yang berarti tidak terdapat *false positive*. *Recall* yang didapatkan juga sebesar 1.00 atau 100% yang berarti model berhasil mengenali semua sampel positif dengan benar, tanpa adanya *false negative*. *F1 Score* yang dihasilkan juga 1.00 atau 100% yang merupakan *harmonic*

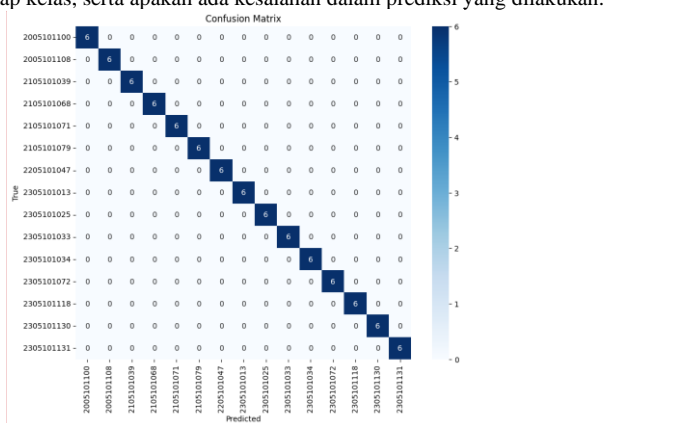
Commented [JM3]: Nomor gambar silhkn disesuaikan pada naskah pada artikel ini.

Commented [JM4]: Sebelum gambar silahkn diberikan narasi, tidak langsung gambar.

mean dari *precision* dan *recall*, yang menegaskan bahwa model memiliki keseimbangan yang sempurna antara kemampuan untuk mengidentifikasi sampel positif dan ketetapan dalam prediksi positifnya. Hasil ini mencerminkan bahwa model yang penulis kembangkan memiliki kinerja optimal dalam mengenali wajah anggota secara keseluruhan.

Hasil Pengujian Sistem

Setelah dilakukan pengujian terhadap model *face recognition* perlu melakukan analisis hasil klasifikasi dari model *face recognition* dengan menggunakan *confusion matrix* yang dapat dilihat pada Gambar 10. *Confusion matrix* ini nantinya akan memberikan gambaran yang jelas mengenai seberapa baik model mengenali setiap kelas, serta apakah ada kesalahan dalam prediksi yang dilakukan.



Gambar 10. Confusion matrix model face recognition

Pada Gambar 10 menunjukkan hasil klasifikasi model *face recognition* pada *dataset* uji yang terdiri dari beberapa kelas wajah anggota. Setiap baris diagonal dalam gambar menunjukkan jumlah prediksi yang benar untuk masing-masing kelas, sementara baris-baris diluar diagonal menunjukkan jumlah kesalahan prediksi. Dalam gambar diatas, seluruh baris diagonal memiliki nilai 6, yang berarti model berhasil mengklasifikasikan 6 gambar wajah dengan benar untuk masing-masing kelas. Tidak ada nilai di luar diagonal, yang menunjukkan bahwa tidak ada kesalahan prediksi dan *misclassification*.

Simpulan

Model yang dikembangkan menggunakan arsitektur CNN menghasilkan hasil yang sangat baik. Penggunaan LBP berperan penting dalam tahap *pre-processing*, dimana algoritma tersebut membantu dalam mengekstraksi fitur yang ada dalam data citra dengan efektif. Arsitektur CNN yang digunakan, dibantu dengan *FaceNet*, memberikan hasil yang sangat memuaskan. Hal ini dibuktikan dengan model yang berhasil mencapai akurasi *training* dan *testing* sebesar 100%.

Hasil evaluasi model pada data uji menunjukkan performa yang optimal dengan semua metrik evaluasi mencapai nilai maksimal. *Accuracy*, *precision*, *recall*, dan *F1 Score* semuanya mencapai nilai 1.00, yang menandakan bahwa model mampu mengklasifikasikan seluruh data uji dengan benar tanpa adanya kesalahan. *Confusion matrix* yang dihasilkan juga menunjukkan bahwa tidak ada data yang salah prediksi, dimana setiap gambar wajah dari setiap kelas dikenali dengan benar oleh model.

Daftar Pustaka

- Arsal, M., Agus Wardijono, B., & Anggraini, D. (2020). Face Recognition Untuk Akses Pegawai Bank Menggunakan Deep Learning Dengan Metode CNN. *Jurnal Nasional Teknologi Dan Sistem Informasi*, 6(1), 55–63. <https://doi.org/10.25077/teknosi.v6i1.2020.55-63>
- Azmi, K., Defit, S., & Sumijan. (2023). Implementasi Convolutional Neural Network (CNN) Untuk Klasifikasi Batik Tanah Liat Sumatera Barat. *Jurnal Unitek*, 16(1), 28–40.

Commented [JM5]: Sebelum gambar silahkn diberikan narasi, tidak langsung gambar.

- Badri, F., Taqijuddin Alawiy, M., & Yuniarno, E. M. (2023). Deep Learning Architecture Based on Convolutional Neural Network (CNN) on Animal Image Classification. *Jurnal Ilmiah Kursor*, 12(2).
- Dandi Darajat, M., Sari, Y. A., & Wihandika, R. C. (2021). *Convolutional Neural Network untuk Klasifikasi Citra Makanan Khas Indonesia* (Vol. 5, Issue 11). <http://j-ptiik.ub.ac.id>
- Darmawan, R. (2022). Perancangan Sistem Absensi menggunakan Face Recognition dengan Haar Cascade Classifier. *Jurnal Ilmiah Informatika Dan Komputer*, 5(2).
- Hosny, K. M., Magdy, T., Lashin, N. A., Apostolidis, K., & Papakostas, G. A. (2021). Refined Color Texture Classification Using CNN and Local Binary Pattern. *Mathematical Problems in Engineering*, 2021. <https://doi.org/10.1155/2021/5567489>
- Jamhari, A., Mukti Wibowo, F., & Andi Saputra, W. (2020). Perancangan Sistem Pengenalan Wajah Secara Real-Time pada CCTV dengan Metode Eigenface. 2(2), 20–32. <https://doi.org/10.20895/INISTA.V2I2>
- Marsland, S. (2014). *Machine Learning An Algorithmic Perspective (Second Edition)*.
- Munawir, Fitria, L., & Hermansyah, M. (2020). Implementasi Face Recognition pada Absensi Kehadiran Mahasiswa Menggunakan Metode Haar Cascade Classifier. *InfoTekJar : Jurnal Nasional Informatika Dan Teknologi Jaringan*, 4(2), 314–320. <https://doi.org/10.30743/infotekjar.v4i2.2333>
- Nugroho, A., & Husin, A. (2022). Analisis Performa Random Forest Menggunakan Normalisasi Atribut Performance Analysis of Random Forest Using Attribute Normalization. *Sistemasi: Jurnal Sistem Informasi*, 11(1), 186–196. <http://sistemasi.ftik.unisi.ac.id>
- Nurhadi, M., & Purnomo, J. (2022). Implementation of Image Classification using Convolutional Neural Network (CNN) Algorithm on Vehicles Images. *ASEAN Journal of System Engineering*, 6(1), 1–5. <http://journal.ugm.ac.id/index.php/ajse1>
- Nurjannah, A. F., Kurniasari, A. S. D., Sari, Z., & Azhar, Y. (2022). Pneumonia Image Classification Using CNN with Max Pooling and Average Pooling. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 6(2), 330–338. <https://doi.org/10.29207/resti.v6i2.4001>
- Pribadi, O. (2023). Aplikasi Pengenalan Wajah Menggunakan Algoritma HAAR CASCADE CLASSIFIER dan Local Binary Pattern Histogram. *Jurnal Times: Technology Informatics & Computer Vision*, 12(1), 40–47.
- Roihan, A., Abas Sunarya, P., & Rafika, A. S. (2020). Pemanfaatan Machine Learning dalam Berbagai Bidang: Review paper. *IJCIT (Indonesian Journal on Computer and Information Technology)*, 5(1), 75–82.
- Syarif, S., Baharuddin, M., & Muslimin. (2023). Penerapan Metode Convolutional Neural Network Pada Face Recognition Untuk Smart Loker. *Jurnal EKSITASI*, 2(2), 2829–5110.
- Utami, A. P., Stehvanie, F., & Ramadhani, K. N. (2021). Pengenalan Logo Kendaraan Menggunakan Metode Local Binary Pattern dan Random Forest. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 5(4), 639–646. <https://doi.org/10.29207/resti.v5i4.3085>
- Utomo, B. T., Fitri, I., & Mardiani, E. (2020). Penerapan Face Recognition Pada Aplikasi Akademik Online. *Jurnal Informatik*, 16(3).