

# Pengembangan Model Kecerdasan Buatan untuk Pendeteksian Wajah dalam Video Berbasis Deep Learning Menggunakan Algoritma Haar Cascade Classifier dan Convolutional Neural Network

Ananda Putra Wahyu Riyanto<sup>1</sup>, Habib Nurrohmad Sugiharto<sup>2\*</sup>

<sup>1,2\*</sup> Student of Informatics Department Universitas Pembangunan Nasional Jawa Timur

<sup>1</sup>[22081010038@student.upnjatim.ac.id](mailto:22081010038@student.upnjatim.ac.id)

<sup>2\*</sup>[22081010165@student.upnjatim.ac.id](mailto:22081010165@student.upnjatim.ac.id)

<sup>3</sup>Lecturer of Informatics Department Universitas Pembangunan Nasional Jawa Timur

\*Corresponding author email: [basukirahmat.if@upnjatim.ac.id](mailto:basukirahmat.if@upnjatim.ac.id)

**Abstrak**— Face recognition merupakan teknologi biometrik yang semakin populer di era revolusi industri 4.0, terutama dalam aplikasi seperti smart home, keamanan, dan sistem presensi. Implementasi face recognition memerlukan metode yang mampu mengenali wajah dengan cepat dan akurasi tinggi. Penelitian ini bertujuan untuk mengevaluasi tingkat akurasi dan waktu komputasi dari Algoritma Haar Cascade Classifier dan Convolutional Neural Network dalam pengenalan wajah menggunakan metode machine learning. Penentuan tingkat akurasi dilakukan dengan menghitung jumlah wajah yang berhasil dikenali dari keseluruhan data wajah. Waktu komputasi diukur berdasarkan durasi yang dibutuhkan selama proses pengenalan wajah berlangsung. Proses pengukuran akurasi dan waktu komputasi dilakukan menggunakan program komputasi Python dengan bantuan library numpy dan tensorflow. Berdasarkan analisis yang dilakukan, proses deteksi wajah menggunakan Algoritma Haar Cascade dan Convolutional Neural Network mencapai akurasi sebesar 98.84%, dengan waktu rata-rata yang dibutuhkan untuk mengenali wajah adalah 0,05 detik.

**Kata Kunci**— Face Recognition, Haar Cascade Classifier, Convolutional Neural Network, Machine Learning

## I. PENDAHULUAN

Perkembangan teknologi di era Revolusi Industri 4.0 telah mendorong inovasi di berbagai bidang, termasuk keamanan, rumah pintar, serta sistem waktu dan kehadiran. Salah satu teknologi biometrik yang akan berperan penting dalam inovasi ini adalah pengenalan wajah. Teknologi ini memungkinkan identifikasi dan verifikasi otomatis orang berdasarkan fitur wajah, memberikan solusi yang efisien dan aman untuk berbagai aplikasi praktis.

Pengenalan wajah bekerja dengan menganalisis dan mengenali pola pada gambar wajah, kemudian mencocokkannya dengan data yang disimpan sebelumnya. Agar teknologi ini dapat diterapkan secara efektif, diperlukan metode yang tidak hanya akurat tetapi juga cepat dalam melakukan proses identifikasi. Dua metode yang umum digunakan dalam pengenalan wajah adalah algoritma klasifikasi kaskade Haar dan jaringan saraf konvolusional (CNN).

Algoritma Haar Cascade Classifier adalah metode yang umum digunakan untuk pengenalan wajah karena kesederhanaan dan kecepatan komputasinya. Algoritma ini bekerja dengan menggunakan serangkaian fitur Haar untuk

mendeteksi objek pada gambar. Sedangkan convolutional neural network (CNN) merupakan metode yang lebih kompleks dan umum digunakan dalam pengenalan pola dan klasifikasi gambar. CNN mampu mengenali fitur kompleks dari gambar melalui pembelajaran mendalam.

Penelitian ini bertujuan untuk mengevaluasi kinerja dua algoritma dalam konteks pengenalan wajah. Tujuan utama dari penelitian ini adalah untuk mengetahui keakuratan dan waktu komputasi dari pengklasifikasi Haar Cascade dan algoritma CNN. Akurasi diukur berdasarkan jumlah wajah yang berhasil dikenali dari seluruh data wajah yang tersedia, sedangkan waktu komputasi diukur berdasarkan waktu yang dibutuhkan untuk melakukan pengenalan wajah.

Metode evaluasi ini dilakukan dengan menggunakan program komputer yang dikembangkan dengan bahasa pemrograman Python, menggunakan pustaka numpy dan tensorflow. Hasil penelitian ini diharapkan dapat memberikan wawasan lebih lanjut mengenai kelebihan dan kekurangan masing-masing metode, serta rekomendasi penerapan pengenalan wajah yang lebih efektif pada aplikasi dunia nyata lainnya.

## II. METODE

Penelitian ini adalah penelitian terapan yang dimulai dengan studi kepustakaan mengenai konsep sistem pengenalan wajah dan machine learning. Selanjutnya, algoritma Haar Cascade Classifier dan Convolutional Neural Network (CNN) diimplementasikan dalam proses pengenalan wajah.

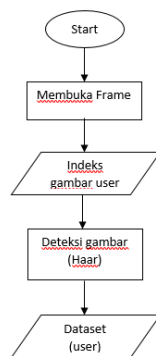
Data yang digunakan adalah data primer berupa citra wajah tampak depan dengan berbagai sudut pengambilan, yang digunakan untuk data pelatihan (training) dan pengujian (testing). Teknik pengumpulan data dilakukan dengan pengambilan citra wajah menggunakan kamera. Kamera yang digunakan adalah webcam Lenovo Legion 5 Pro 2,1 MP dan mikrofon tunggal.

Data yang dianalisis adalah citra wajah. Analisis dilakukan dengan menentukan tingkat akurasi pengenalan wajah menggunakan algoritma Haar Cascade Classifier dan CNN, serta menghitung waktu komputasi yang dibutuhkan dalam proses pengenalan wajah.

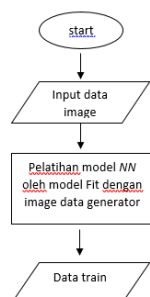
Langkah-langkah analisis data dalam penelitian ini adalah sebagai berikut:

1. Melakukan studi pustaka dari berbagai sumber seperti buku, jurnal, makalah, skripsi, dan artikel mengenai algoritma Haar Cascade Classifier, CNN, pengolahan citra, dan sistem pengenalan wajah.
2. Mengumpulkan data citra wajah menggunakan kamera laptop.
3. Melakukan pra-pemrosesan citra dengan algoritma Haar Cascade Classifier.
4. Merancang arsitektur Convolutional Neural Network untuk pelatihan dan pengujian.
5. Melakukan proses konvolusi pada citra menggunakan library Keras dan TensorFlow.
6. Melakukan proses max-pooling pada citra.
7. Membuat fully connected layer.
8. Mengimplementasikan program komputasi menggunakan bahasa Python.
9. Melakukan proses pelatihan.
10. Menentukan proses pengujian.
11. Menghitung akurasi program komputasi dalam proses pengenalan wajah.
12. Menghitung waktu komputasi program.

Proses komputasi dalam penelitian ini dilakukan menggunakan Google Colab dan IDLE Python dengan memanfaatkan library numpy, matplotlib, Keras, dan TensorFlow. Pada pengolahan citra, dilakukan proses konvolusi sebanyak tiga kali, dilanjutkan dengan max-pooling, dan kemudian fully connected layer.



Gambar 1. Flowchart pengambilan dataset



Gambar 2. Flowchart *training* dataset

Pada tahap ini, algoritma Haar Cascade Classifier dan Convolutional Neural Network disalin ke dalam format standar menggunakan bahasa pemrograman Python. Terdapat tiga program yang digunakan, yaitu:

1. Program untuk pengambilan dataset wajah dalam format grayscale yang akan digunakan sebagai data latih dan data uji menggunakan algoritma Haar.
2. Program pelatihan untuk memperoleh data training menggunakan library Keras dan TensorFlow.
3. Program pengujian untuk proses pengenalan wajah.

Setelah pembuatan program, dilakukan simulasi dan pengujian. Pada tahap ini, simulasi dan pengujian program yang telah dibuat dilakukan untuk melihat tingkat akurasi program dalam pengenalan wajah serta menentukan waktu komputasi yang dibutuhkan program dalam melakukan pengenalan wajah.

### III. HASIL DAN PEMBAHASAN

#### A. Pengumpulan Data

Langkah pertama adalah mengumpulkan data gambar. Data gambar wajah diambil dengan kamera webcam Laptop Lenovo ideaped320 dengan spesifikasi 0,3 MP kamera dengan single mic. Pengambilan citra dilakukan dengan menggunakan pustaka OpenCV dan algoritma haar cascade classification sehingga menghasilkan data citra skala abu-abu dengan ukuran piksel 140 x 140. Sebanyak citra wajah yang diperoleh dibagi menjadi dua lapisan yaitu data latih dan data uji. Data ini diekstraksi dari dataset yang telah diproses sebelumnya, kemudian dibagi menjadi dua kelas, yaitu 80% untuk data pengujian dan 20D44 untuk data pelatihan.

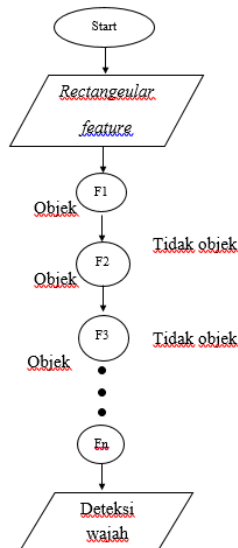
#### B. Pre-Processing Data

Tahap ini melibatkan preprocessing citra berupa cropping. Cropping dilakukan untuk fokus pada objek yang dianalisis, yaitu citra wajah, sehingga latar belakang tidak perlu diproses oleh komputer. Untuk melakukan cropping wajah pada penelitian ini, digunakan algoritma haar cascade classifier. Algoritma ini mendeteksi wajah dalam gambar dan melakukan cropping.

Cara kerja haar cascade dalam mendeteksi wajah adalah dengan menggunakan teknik sliding window berukuran 24 x 24 yang digerakkan ke seluruh bagian gambar untuk mencari pola yang menyerupai wajah. Setelah itu, dilakukan scaling agar dapat mendeteksi wajah yang ukurannya lebih besar atau lebih kecil dibandingkan dengan ukuran asli yang ada pada classifier.

Dalam preprocessing citra ini, objek diklasifikasikan menjadi wajah dan bukan wajah menggunakan algoritma haar

cascade classifier. Penelitian ini menggunakan tiga jenis fitur untuk mendeteksi objek dalam haar, yaitu fitur edge, fitur line, dan fitur tipe four-rectangle, dengan bantuan pustaka OpenCV untuk deteksi objek.



Gambar 3. Flowchart Deteksi Wajah

### C. Klasifikasi fitur

- Pada original dataset, dimisalkan terdapat dua kelas yaitu positif dan negatif, langkah awal dari adaboost adalah menentukan classifier lemah dengan melihat dimana error paling sedikit yang didefinisikan secara matematis berikut ini.
- Setelah memperoleh classifier lemah, langkah berikutnya adalah menambahkan bobot pada tahap kedua. Tujuannya adalah untuk memperbaiki klasifikasi yang sebelumnya salah, dengan mengidentifikasi klasifikasi yang memiliki nilai error paling rendah.
- Pada tahap ketiga, fokusnya adalah menemukan klasifikasi dengan nilai error yang rendah.
- Tahap keempat melibatkan penambahan bobot serupa dengan tahap kedua, diikuti dengan menentukan klasifikasi yang memiliki nilai error terendah.

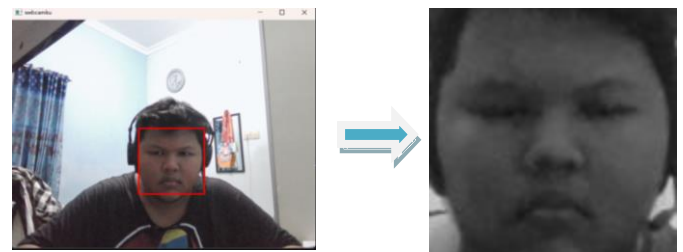
- Pada akhirnya, setelah memperoleh 3 classifier, dilakukan kombinasi untuk menghasilkan hasil akhir.

Cara mengkombinasikan dari tiga classifier menjadi strong classifier

Langkah langkahnya

1. training weak classifier
2. pilih error paling kecil

Selanjutnya melakukan klasifikasi fitur dengan cascade classifier, menentukan dibasis mana saja terdapat wajah dan tidak wajah yang dilakukan dengan teknik sliding window



Gambar 4. Hasil pendeteksi objek dan cropping

Penelitian ini menggunakan data pelatihan dan data pengujian, masing-masing mewakili 80 dan 20 kumpulan data sampel.

### D. Pengolahan Citra

Pada saat pengolahan citra, data yang diolah adalah hasil segmentasi citra pada data latih sesuai matriks, nilai matriks tersebut akan dimasukkan ke dalam model yang akan digunakan, ditambah bobot dan menentukan lapisan mana yang akan digunakan dan menambahkan fungsi aktivasi untuk mendapatkan data pelatihan, yang disebut data pelatihan. Nilai matriks diperoleh menggunakan matplotlib dan perpustakaan numpy digunakan untuk mengubah gambar skala abu-abu ke bentuk raster.

### E. Proses Komputasi

Sebelum membuat program, langkah pertama adalah melakukan instalasi bahasa Python beserta pustaka yang diperlukan untuk menjalankan proses program. Dalam

penelitian ini, terdapat tiga tahap utama yang dilakukan, yaitu pengambilan dataset, proses pelatihan (training), dan proses pengujian (testing).

#### 1. Pseudocode program untuk pengambilan dataset:

- Buka kamera menggunakan pustaka OpenCV.
- Lakukan pre-processing.
- Lakukan ekstraksi fitur.
- Gunakan Cascade Classifier yang open source untuk mengambil dataset.
- Simpan data citra wajah yang diperoleh ke dalam folder database.
- Lakukan pelabelan pada citra.
- Persiapkan data untuk digunakan.

#### 2. Pseudocode program untuk pengenalan wajah (training dan testing):

- Impor pustaka: OpenCV, NumPy, Matplotlib, Keras, dan TensorFlow.
- Buat direktori masing-masing untuk data training dan data testing.
- Buat sub-direktori untuk data wajah.
- Buat generator data citra untuk data training dan data testing.
- Bangun arsitektur CNN (Convolutional Neural Network).
- Panggil fungsi compile pada objek model.
- Tetapkan fungsi loss dan optimizer.
- Latih model. Jika proses pelatihan berhasil, lakukan prediksi menggunakan model tersebut. Jika tidak, revisi arsitektur model dan perbaiki dataset.
- Lakukan pengujian pada model dengan memasukkan dataset testing.
- Lakukan pengujian dengan input yang bukan wajah untuk menguji keandalan model.

Dengan mengikuti langkah-langkah tersebut, diharapkan program dapat dibangun dengan efektif untuk pengambilan dataset serta proses training dan testing dalam pengenalan wajah.

#### F. Hasil Training Model

Model hasil pelatihan diperoleh dengan melakukan komputasi menggunakan bahasa Python serta pustaka TensorFlow dan Keras.

Epo ch	loss	accu cy	val_loss	val accuracy
1	1.7199	0.5031	3.2873	0.2000
2	0.1947	0.9563	7.9904	0.1750
3	0.0210	0.9969	11.3161	0.2000
4	0.0030	1.0000	11.7703	0.2000
5	0.0036	1.0000	13.5160	0.2000
6	3.9992e-04	1.0000	13.6854	0.2000
7	2.9646e-04	1.0000	13.9937	0.2000
8	1.6635e-04	1.0000	14.5214	0.2000
9	9.5824e-05	1.0000	14.8098	0.2000
10	7.7488e-05	1.0000	15.0243	0.2000

TABEL I

HASIL TRAINING MODEL

Dari tabel diatas diperoleh grafik nilai loss pada traning step dan akurasi pada training step, grafik diperoleh dari hasil komputasi dengan library matplotlib.

#### IV. KESIMPULAN

Berdasarkan analisis yang telah dilakukan, dapat disimpulkan bahwa:

- Algoritma Haar Cascade Classifier dan Convolutional Neural Network berhasil mencapai tingkat akurasi sebesar 98.94% pada data uji.
- Waktu rata-rata yang diperlukan oleh kedua algoritma untuk melakukan klasifikasi citra wajah adalah 0,05 detik..

#### REFERENSI

- [1] Gangopadhyay, I. (2018). Face Detection and Recognition Using Haar Classifier and Lbp Histogram. *International Journal of Advanced Research in Computer Science*, 9(2), 592–598. <https://doi.org/10.26483/ijarcs.v9i2.5815>
- [2] Septyanto, M. W., Sofyan, H., Jayadiant, H., Simanjuntak, O. S., & Prasetyo, D. B. (2020). APLIKASI PRESENSI PENGENALAN WAJAH DENGAN MENGGUNAKAN ALGORITMA HAAR CASCADE CLASSIFIER. *Telematika Jurnal Informatika dan Teknologi Informasi*, 16(2), 87-96.
- [3] Zhao, W., Chellappa, R., Phillips, P. J., & Rosenfeld, A. (2003). Face recognition: A literature survey. *ACM computing surveys (CSUR)*, 35(4), 399-458.
- [4] Hardiyanto, D., & Sartika, D. A. (2018). Optimalisasi Metode Deteksi Wajah berbasis Pengolahan Citra untuk Aplikasi Identifikasi Wajah pada Presensi Digital. *Setrum: Sistem Kendali-Tenaga-elektronika-telekomunikasi-komputer*, 7(1), 107-116.
- [5] Maryati, R. I. S., & Tryatmojo, B. (2014). Akurasi Sistem Face Recognition OpenCV Menggunakan Raspberry Pi Dengan Metode Haar Cascade. *Universitas Paradima*.