JRIIN : Jurnal Riset Informatika dan Inovasi
Volume 2, No. 4, September Tahun 2024
ISSN 3025-0919 (media online)
Hal 565-571

Implementasi Pendeteksian Wajah Dan Mata Secara Real-Time Menggunakan Metode Haar Cascade Dengan OpenCV

Siti Khusnul Salbiah[1*], Nurma Junita[2], Cikal Putri Aprillia[3], Rizki Alpiansyah[4], Perani Rosyani[5]

[1,2,3,4,5]Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Pamulang, Kota Tangerang Selatan, Indonesia
Email : [1*]khusnul141100@gmail.com, [2]nurmaj23@gmail.com,[3]cikalputri.a@gmail.com, [4]Apinalpiansyah86@gmail.com, [5]dosen00837@unpam.ac.id
(* : coressponding author)

Abstrak - Penelitian ini mengimplementasikan pendeteksian wajah dan mata secara real-time menggunakan metode Haar Cascade di OpenCV, dengan aplikasi utama dalam keamanan, interaksi manusia-komputer, dan analisis perilaku. Dataset terdiri dari gambar wajah yang diambil secara real-time dengan variasi pencahayaan dan posisi wajah yang berbeda. Proses pendeteksian mencakup pengambilan gambar, konversi ke skala abu-abu, dan penerapan Haar Cascade. Hasil penelitian menunjukkan Haar Cascade mampu mendeteksi wajah dan mata dengan tingkat akurasi tinggi pada kondisi pencahayaan baik dan posisi wajah normal, namun performanya menurun pada kondisi pencahayaan buruk dan orientasi wajah yang tidak biasa. Meskipun efektif dan efisien, untuk aplikasi dengan persyaratan akurasi lebih tinggi, disarankan penggunaan metode deteksi berbasis deep learning.

Kata kunci: Python, Deteksi Wajah, Metode Haar Cascade, Open CV, Implementasi

Abstract *- This research implements real-time face and eye detection using the Haar Cascade method in OpenCV, with primary applications in security, human-computer interaction, and behavior analysis. The dataset consists of real-time facial images with variations in lighting conditions and facial positions. The detection process involves capturing images, converting them to grayscale, and applying the Haar Cascade method. The research findings demonstrate that Haar Cascade achieves high accuracy in detecting faces and eyes under good lighting conditions and normal facial positions, but its performance decreases in poor lighting conditions and unusual facial orientations. Despite its effectiveness and efficiency, for applications requiring higher accuracy, deep learning-based detection methods are recommended.*

Keywords: *Python, Face Detection, Haar Cascade Method, OpenCV, Implementation*

1. PENDAHULUAN

Pengenalan dan deteksi objek visual seperti wajah dan mata telah menjadi fokus utama dalam pengolahan citra dan visi komputer. Kemampuan untuk mengenali wajah dan mata sangat penting karena keduanya merupakan elemen kunci dalam interaksi manusia dengan sistem komputer. Wajah dan mata dipilih sebagai objek studi utama karena kompleksitas fitur mereka yang memainkan peran vital dalam berbagai aplikasi seperti keamanan, identifikasi individu, dan antarmuka pengguna.

Metode yang digunakan dalam penelitian ini adalah metode Haar Cascade, yang terbukti efektif dalam deteksi objek berbasis pola. Kelebihan utama metode Haar Cascade adalah kemampuannya untuk secara cepat mendeteksi pola objek seperti wajah dan mata dengan memanfaatkan fitur-fitur khas yang dikenali oleh classifier Haar. Namun, metode ini juga memiliki kelemahan, seperti sensitivitas terhadap kondisi pencahayaan yang tidak ideal dan sudut pandang yang tidak biasa, yang dapat mempengaruhi akurasi deteksi.

Meskipun menjanjikan, implementasi deteksi wajah dan mata menggunakan metode Haar Cascade menemui beberapa tantangan, termasuk perluasan akurasi deteksi di bawah kondisi pencahayaan yang bervariasi dan orientasi wajah yang tidak standar.

Untuk mengatasi tantangan ini, penelitian ini mengusulkan pendekatan implementasi menggunakan Python dengan dukungan dari OpenCV, sebuah pustaka pengolahan citra terkenal dan terpercaya. Pendekatan ini diharapkan dapat meningkatkan akurasi deteksi wajah dan mata serta meningkatkan kemampuan adaptasi terhadap berbagai kondisi lingkungan.

JRIIN : Jurnal Riset Informatika dan Inovasi
Volume 2, No. 4, September Tahun 2024
ISSN 3025-0919 (media online)
Hal 565-571

Tujuan utama dari penelitian ini adalah untuk meningkatkan akurasi deteksi wajah dan mata menggunakan metode Haar Cascade berbasis OpenCV. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi signifikan dalam pengembangan sistem pengenalan objek berbasis citra yang lebih handal dan efisien.

2. METODE PENELITIAN

Metode yang digunakan dalam penelitian ini terdiri dari beberapa langkah utama, yaiut :

a. Dimulai dari instalasi OpenCV menggunakan perintah 'pip' *pip install opencv-python* untuk memfasilitasi integrasi dengan Python.

b. Langkah selanjutnya adalah memuat classifier Haar Cascade yang sudah dilatih untuk deteksi wajah dan mata dari pustaka OpenCV. Classifier ini berisi informasi yang diperlukan untuk mengidentifikasi fitur-fitur wajah, termasuk penggunaan 'haarcascade_frontalface_default.xml' dan 'haarcascade_eye.xml', yang sudah tersedia sebagai bagian dari distribusi OpenCV.

2.1. Analisa Data

Penelitian ini dimulai dengan mengumpulkan dataset yang mencakup serangkaian gambar wajah yang diambil menggunakan kamera. Dataset dirancang untuk mencakup berbagai kondisi pencahayaan, termasuk terang, redup, dan dalam ruangan, serta variasi posisi wajah seperti sudut pandang, rotasi, dan jarak. Setiap gambar dipilih secara acak dari situasi nyata yang mungkin dihadapi dalam aplikasi deteksi visual, memastikan representasi yang menyeluruh dari keberagaman kondisi yang mungkin terjadi.

2.2. Pengolahan Data

Setiap frame gambar yang diperoleh dari kamera kemudian diolah menggunakan langkah-langkah berikut :

a. Dalam penelitian ini, setiap gambar yang telah diambil dalam format warna BGR (Blue-Green-Red) dikonversi ke skala abu-abu menggunakan fungsi cv2.cvtColor() dari OpenCV. Konversi ke skala abu-abu mengubah setiap piksel dalam gambar menjadi nilai kecerahan tunggal, yang dihitung berdasarkan rata-rata intensitas dari komponen warna (biru, hijau, merah). Langkah ini dilakukan untuk menyederhanakan data yang digunakan dalam deteksi wajah dan mata, dengan mempertahankan informasi penting tentang kecerahan gambar tanpa mempertimbangkan perbedaan warna yang mungkin tidak relevan untuk tujuan deteksi visual.

b. Setelah konversi ke skala abu-abu, setiap gambar dianalisis secara teliti untuk mengevaluasi kualitas dan representasi yang akurat dari berbagai kondisi yang ada. Gambar-gambar yang tidak memenuhi kriteria atau tidak mewakili kondisi yang diinginkan dipilih untuk dihapus atau diperbaiki kualitasnya. Langkah ini dilakukan untuk memastikan dataset yang digunakan dalam penelitian memiliki kualitas yang baik dan mampu mencerminkan variasi kondisi yang berbeda dengan tepat.

2.3. Metode Haar Cascade

Metode Haar Cascade digunakan untuk mendeteksi wajah dan mata dalam waktu nyata. Proses ini melibatkan :

a. Classifier Haar Cascade untuk deteksi wajah (haarcascade_frontalface_default.xml) dan mata (haarcascade_eye.xml) dimuat menggunakan fungsi cv2.CascadeClassifier() dari OpenCV. Teknologi ini menggunakan fitur integral dan metode pembelajaran mesin untuk mengidentifikasi pola objek dalam gambar.

b. Parameter-parameter seperti scaleFactor, minNeighbors, dan minSize disesuaikan untuk mengoptimalkan proses deteksi. scaleFactor mengontrol seberapa besar gambar diubah ukurannya pada setiap skala deteksi, minNeighbors menentukan seberapa banyak tetangga yang harus ada untuk memvalidasi wajah yang terdeteksi, dan minSize menentukan ukuran

JRIIN : Jurnal Riset Informatika dan Inovasi
Volume 2, No. 4, September Tahun 2024
ISSN 3025-0919 (media online)
Hal 565-571

minimum objek yang akan dideteksi dalam gambar.

Berikut adalah implementasi lengkap untuk deteksi wajah dan mata menggunakan metode Haar Cascade dengan OpenCV :

```
import cv2
import matplotlib.pyplot as plt
import keyboard

# Inisialisasi video capture dari kamera
cap = cv2.VideoCapture(0)

# Muat classifier wajah dan mata
faceCascade =
cv2.CascadeClassifier(cv2.data.haarcascade
s + 'haarcascade_frontalface_default.xml')
eye_cascade =
cv2.CascadeClassifier(cv2.data.haarcascade
s + 'haarcascade_eye.xml')

plt.ion()  # Mengaktifkan mode interaktif

# Membuat subplot untuk dua gambar
fig, (ax_left, ax_right) = plt.subplots(1, 2,
figsize=(10, 5))

# Inisialisasi gambar untuk subplot kiri
(gambar asli)
ax_left.set_title('Gambar Kamera - Gambar
Asli')
ret, frame = cap.read()
img_display_left =
ax_left.imshow(cv2.cvtColor(frame,
cv2.COLOR_BGR2RGB))
```

```
# Inisialisasi gambar untuk subplot kanan
(terdeteksi wajah)
ax_right.set_title('Gambar Kamera -
Terdeteksi Wajah')
img_display_right =
ax_right.imshow(cv2.cvtColor(frame,
cv2.COLOR_BGR2RGB))

while True:
    # Baca frame dari kamera
    ret, frame = cap.read()
    if not ret:
        break

    # Perbarui gambar di subplot kiri (gambar
asli)
    img_display_left.set_data(cv2.cvtColor(fr
ame, cv2.COLOR_BGR2RGB))

    # Konversi ke skala abu-abu
    gray = cv2.cvtColor(frame,
cv2.COLOR_BGR2GRAY)

    # Deteksi wajah
    faces = faceCascade.detectMultiScale(
        gray,
        scaleFactor=1.1,
        minNeighbors=5,
        minSize=(30, 30)
    )
```

JRIIN : Jurnal Riset Informatika dan Inovasi
Volume 2, No. 4, September Tahun 2024
ISSN 3025-0919 (media online)
Hal 565-571

```
# Deteksi wajah dan mata pada frame
    for (x, y, w, h) in faces:
        cv2.rectangle(frame, (x, y), (x + w, y + h),
(0, 255, 0), 2)  # Mengubah warna wajah
menjadi hijau (0, 255, 0)
        roi_gray = gray[y:y + h, x:x + w]
        roi_color = frame[y:y + h, x:x + w]
        eyes =
eye_cascade.detectMultiScale(roi_gray)
        for (ex, ey, ew, eh) in eyes:
            cv2.rectangle(roi_color, (ex, ey), (ex +
ew, ey + eh), (255, 0, 0), 2)  # Mengubah
warna mata menjadi biru (255, 0, 0)

    # Perbarui gambar di subplot kanan
(terdeteksi wajah)
    img_display_right.set_data(cv2.cvtColor(fra
me, cv2.COLOR_BGR2RGB))

    plt.draw()
    plt.pause(0.001)

    # Tambahkan pengecekan keyboard
 if keyboard.is_pressed('esc'):
        plt.close()
        break

# Bebaskan memori dan tutup semua jendela
OpenCV
cap.release()
plt.ioff()  # Menonaktifkan mode interaktif
plt.show()
```

2.4. Akurasi

Pengujian dilakukan untuk mengevaluasi keakuratan dan kinerja deteksi terhadap berbagai kondisi pencahayaan dan posisi wajah. Parameter deteksi yang diatur dievaluasi untuk memastikan respons optimal terhadap kondisi yang berbeda. Evaluasi ini mencakup analisis keakuratan deteksi dan performa deteksi dalam berbagai skenario praktis.

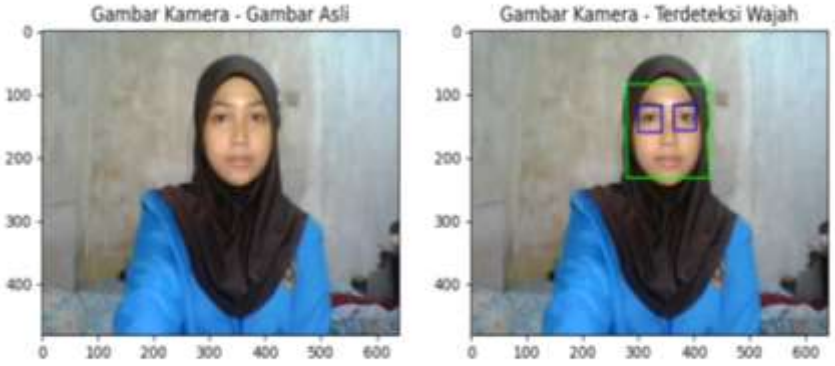
3. HASIL DAN PEMBAHASAN

Penelitian ini berhasil menerapkan metode Haar Cascade menggunakan OpenCV untuk mendeteksi wajah dan mata seseorang. Hasil dari deteksi ekspresi wajah menunjukkan tingkat akurasi yang tinggi, terutama dalam pengenalan wajah manusia, dengan syarat pencahayaan yang memadai.

Berikut adalah contoh hasil dari implementasi pengujian ini :

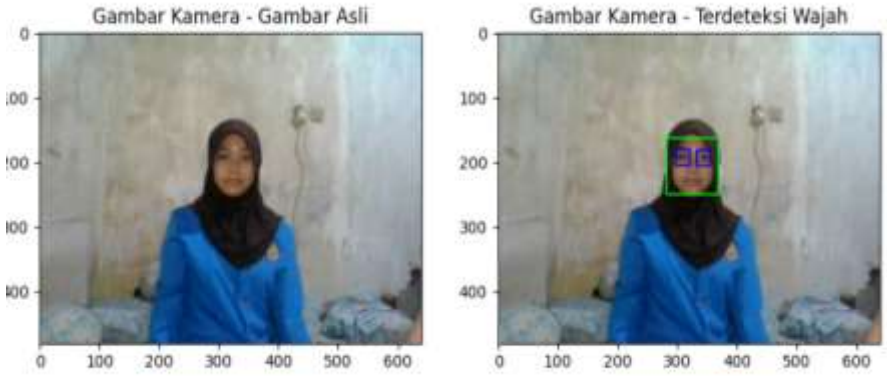
a. Dalam kondisi di mana hanya terdapat satu orang dan jaraknya 75 cm, kamera deteksi masih memberikan hasil yang sangat akurat, dengan tingkat keakuratan mencapai 100%.

JRIIN : Jurnal Riset Informatika dan Inovasi
Volume 2, No. 4, September Tahun 2024
ISSN 3025-0919 (media online)
Hal 565-571



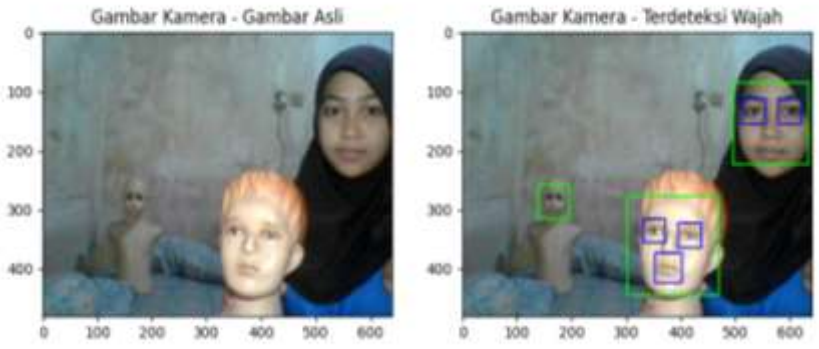
Gambar 1. Kondisi 1 orang jarak 75 cm

b. Dalam situasi di mana hanya terdapat satu orang dan jaraknya adalah 100 cm, sistem deteksi kamera tetap memberikan hasil yang sangat akurat, dengan tingkat keakuratan mencapai 100%.



Gambar 2. Kondisi 1 orang jarak 100 cm

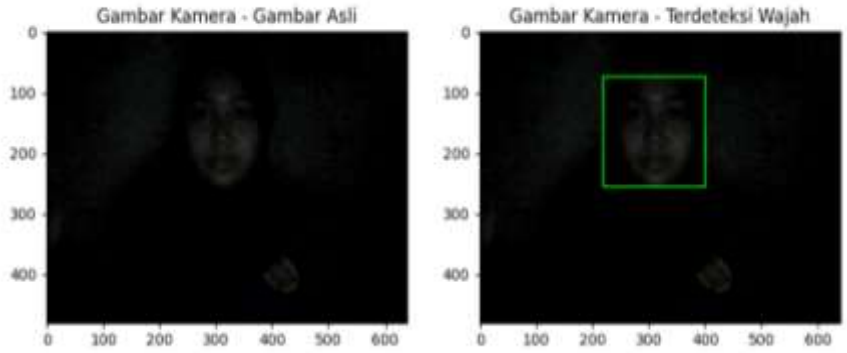
c. Dalam uji coba menggunakan tiga orang dengan jarak yang bervariasi, kamera masih mencapai tingkat akurasi sebesar 75%. Namun, karena perbedaan jarak, kamera tidak dapat mendeteksi mata orang yang berada pada jarak yang lebih jauh.



Gambar 3. Kondisi 3 orang jarak bervariasi

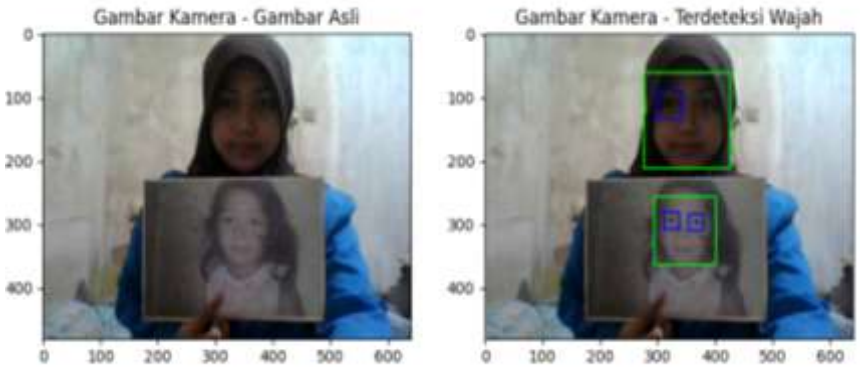
d. Dalam uji coba di kondisi pencahayaan yang kurang baik atau gelap, kamera mencapai akurasi 75% masih dapat mendeteksi wajah tetapi tidak dapat mendeteksi mata.

JRIIN : Jurnal Riset Informatika dan Inovasi
Volume 2, No. 4, September Tahun 2024
ISSN 3025-0919 (media online)
Hal 565-571



Gambar 4. Kondisi pencahayaan gelap

e. Pengujian menggunakan deteksi gambar foto menunjukkan bahwa kamera dapat mencapai akurasi hingga 100%.



Gambar 5. Menggunakan Gambar foto

4. KESIMPULAN

4.1. Kesimpulan

Penelitian ini berhasil mengimplementasikan deteksi wajah dan mata secara real-time menggunakan metode Haar Cascade dengan pustaka OpenCV. Hasil pengujian menunjukkan :

a. Akurasi Deteksi : Metode Haar Cascade mendeteksi wajah dan mata dengan akurasi tinggi, mencapai 100% dalam kondisi pencahayaan yang baik pada jarak 75 cm dan 100 cm.

b. Pengaruh Pencahayaan : Akurasi menurun menjadi sekitar 75% dalam kondisi pencahayaan buruk, dengan deteksi wajah masih memungkinkan tetapi kesulitan mendeteksi mata.

c. Variasi Jarak dan Jumlah Orang: Pada pengujian dengan tiga orang dengan jarak bervariasi, akurasi deteksi wajah adalah 75%, dengan kesulitan mendeteksi mata pada orang yang lebih jauh dari kamera.

d. Deteksi Gambar Foto : Deteksi menggunakan gambar foto mencapai akurasi hingga 100%, menunjukkan efektivitas metode ini dalam kondisi statis dengan pencahayaan baik.

4.2. Saran

Untuk penelitian dan implementasi lebih lanjut, disarankan:

JRIIN : Jurnal Riset Informatika dan Inovasi
Volume 2, No. 4, September Tahun 2024
ISSN 3025-0919 (media online)
Hal 565-571

a. Peningkatan Pencahayaan : Gunakan sistem pencahayaan tambahan atau algoritma pemrosesan gambar untuk meningkatkan kualitas gambar sebelum deteksi.

b. Metode Tambahan : Kombinasikan metode Haar Cascade dengan metode berbasis deep learning, seperti Convolutional Neural Networks (CNN), untuk meningkatkan akurasi dalam kondisi pencahayaan buruk dan posisi wajah yang tidak standar.

c. Algoritma Adaptif : Kembangkan algoritma deteksi adaptif yang dapat menyesuaikan parameter deteksi berdasarkan kondisi lingkungan.

d. Implementasi pada Platform Lain : Kembangkan implementasi pada platform lain seperti perangkat mobile atau embedded systems untuk fleksibilitas dan aplikasi praktis dalam deteksi real-time.

DAFTAR PUSTAKA

Face Detection – Python https://www.kaggle.com/discussion/general/105322

Rosyani, P., & Retnawati, R. (2023). Ekstraksi Fitur Wajah Menggunakan Metode Viola Jones dengan Tools Cascade Detector. JURIKOM (Jurnal Riset Komputer), 10(2), 633-639.

Warzuqni, A. A., Sabilla, D. P., Agustin, Z., & Rosyani, P. (2022). ANALISA SISTEM PRESENSI KELAS MENGGUNAKAN PENGENAL WAJAH DENGAN METODE HAAR CASCASE CLASSIFIER. Jurnal Manajemen, Ekonomi, Hukum, Kewirausahaan, Kesehatan, Pendidikan dan Informatika (MANEKIN), 1(1: September), 15-19.

S. Abidin, "Deteksi Wajah Menggunakan Metode Haar Cascade Classifier Berbasis Webcam Pada Matlab," J. Teknol. Elekterika, vol. 15, no. 1, p. 21, 2018, doi: 10.31963/elekterika.v15i1.2102.

OpenCV Documentation. (n.d.). Retrieved from https://docs.opencv.org/4.x/d6/d00/tutorial_py_root.html.

OpenCV Object Detection Tutorial. (n.d.). Retrieved from https://docs.opencv.org/4.x/d2/d64/tutorial_table_