

## Implementasi Sistem Deteksi Kantuk Secara Real-Time Bagi Pengemudi Menggunakan Metode Eye Aspect Ratio

Mochammad Fadiil Thoriq<sup>1\*</sup>, Muhammad Fathi Ramdhana<sup>2</sup>, Desinta Nur Rahma<sup>3</sup>,  
Najla Amelia Putri<sup>4</sup>, Rafi Hilal Zahir<sup>5</sup>, Gema Parasti Mindara<sup>6</sup>, Endang Purnama Giri<sup>7</sup>

<sup>1,2,3,4,5</sup> Departemen Teknologi Rekayasa Perangkat Lunak, IPB University, Indonesia

<sup>6</sup> Departemen Teknologi Rekayasa Komputer, IPB University, Indonesia

<sup>7</sup> Departemen Ilmu Komputer, IPB University, Indonesia

[mochfadiilthoriq@apps.ipb.ac.id](mailto:mochfadiilthoriq@apps.ipb.ac.id)<sup>1</sup>, [fathiramdhana@apps.ipb.ac.id](mailto:fathiramdhana@apps.ipb.ac.id)<sup>2</sup>, [desinta19nur@apps.ipb.ac.id](mailto:desinta19nur@apps.ipb.ac.id)<sup>3</sup>,  
[najlamelia@apps.ipb.ac.id](mailto:najlamelia@apps.ipb.ac.id)<sup>4</sup>, [pilal.hilal@apps.ipb.ac.id](mailto:pilal.hilal@apps.ipb.ac.id)<sup>5</sup>, [gemaparasti@apps.ipb.ac.id](mailto:gemaparasti@apps.ipb.ac.id)<sup>6</sup>,  
[endang\\_pg@apps.ipb.ac.id](mailto:endang_pg@apps.ipb.ac.id)<sup>7</sup>

Alamat Jl. Kumbang No.14, RT.02/RW.06, Babakan, Kecamatan Bogor Tengah, Kota Bogor

Korespondensi penulis: [mochfadiilthoriq@apps.ipb.ac.id](mailto:mochfadiilthoriq@apps.ipb.ac.id)

**Abstract.** Traffic accidents are one of the leading causes of death worldwide, where drowsiness while driving is a significant factor that reduces driver alertness. This study develops a real-time driver drowsiness detection system using the Eye Aspect Ratio (EAR) method to avoid this. EAR calculates the ratio of the upper and lower eyelid distances to detect signs of drowsiness based on changes in eye shape. This system utilizes the OpenCV and Dlib libraries to identify faces and measure EAR, with a threshold of 0.25 as a warning trigger. If the EAR value drops below the threshold in several consecutive frames, the system automatically activates an alarm to increase driver alertness. With the advantages of cost efficiency and ease of implementation without additional hardware, this system is suitable for various types of vehicles. The results show that this system is effective in providing early warnings, thus helping to reduce the risk of accidents due to drowsiness.

**Keywords:** Driver Drowsiness, Eye Aspect Ratio, Drowsiness Detection, Real-Time, Computer Vision.

**Abstrak.** Kecelakaan lalu lintas merupakan salah satu penyebab kematian tertinggi di seluruh dunia, di mana rasa kantuk saat berkendara menjadi salah satu faktor signifikan yang menurunkan kewaspadaan pengemudi. Penelitian ini mengembangkan sistem deteksi kantuk pengemudi secara real-time menggunakan metode Eye Aspect Ratio (EAR) untuk menghindari hal tersebut. EAR menghitung rasio jarak kelopak mata atas dan bawah untuk mendeteksi tanda-tanda kantuk berdasarkan perubahan bentuk mata. Sistem ini memanfaatkan pustaka OpenCV dan Dlib untuk mengidentifikasi wajah dan mengukur EAR, dengan ambang batas 0,25 sebagai pemicu peringatan. Jika nilai EAR turun di bawah ambang batas dalam beberapa frame berturut-turut, sistem secara otomatis mengaktifkan alarm untuk meningkatkan kewaspadaan pengemudi. Dengan keunggulan efisiensi biaya dan kemudahan implementasi tanpa perangkat keras tambahan, sistem ini cocok untuk berbagai jenis kendaraan. Hasil penelitian menunjukkan bahwa sistem ini efektif dalam memberikan peringatan dini, sehingga membantu mengurangi risiko kecelakaan akibat kantuk.

**Kata kunci:** Kantuk Pengemudi, Rasio Aspek Mata, Deteksi Kantuk, Waktu Nyata, Visi Komputer.

### 1. LATAR BELAKANG

Kecelakaan lalu lintas merupakan salah satu penyebab kematian terbanyak di seluruh dunia, banyak di antaranya dipicu oleh kesalahan manusia, seperti kelelahan atau kantuk, yang secara signifikan mengurangi konsentrasi dan kewaspadaan pengemudi (Huda, 2020). *National Sleep Foundation* memperkirakan bahwa sekitar 25% kecelakaan kendaraan disebabkan oleh kantuk, suatu situasi di mana pengemudi kehilangan kendali karena penurunan kesadaran dan kemampuan respon optimal (Suradi, Alam, & Rasyid, 2023). Kantuk yang berlebihan dapat menyebabkan pengemudi mengalami *microsleep*, suatu kondisi tidur singkat yang tidak

disengaja dan terjadi secara tiba-tiba. *Microsleep* ini menyebabkan jeda dalam menanggapi situasi sekitar, yang sangat berbahaya saat berkendara karena hanya dalam hitungan detik, kecelakaan dapat terjadi (Ilmadina, Apriliani, & Wibowo, 2022). Situasi ini menunjukkan betapa pentingnya mengembangkan sistem deteksi kantuk yang efektif untuk mengidentifikasi tanda-tanda awal kantuk pada pengemudi sebagai langkah pencegahan dini kecelakaan.

Seiring dengan perkembangan teknologi *computer vision*, berbagai metode telah dikembangkan untuk mendeteksi rasa kantuk pengemudi, seperti *facial landmarks*, *Histogram Oriented Gradients* (HOG), dan *Support Vector Machine* (SVM) untuk mengklasifikasikan kondisi mata dan wajah (Ramadhani, Aulia, Suhartono, & Hadiyoso, 2021). Pengolahan citra digunakan untuk memantau ekspresi wajah dan gerakan mata untuk mendeteksi tanda-tanda rasa kantuk pengemudi, sedangkan metode lainnya menggunakan pendekatan *adaptive thresholding* untuk menyesuaikan deteksi pada kondisi pencahayaan rendah yang sering terjadi saat berkendara di malam hari atau cuaca buruk (Sasono, 2023). Meskipun efektif, beberapa metode tersebut memiliki keterbatasan, terutama dalam hal akurasi pada intensitas cahaya yang bervariasi atau memerlukan perangkat keras khusus yang mungkin kurang efisien untuk diaplikasikan pada kendaraan umum (Firdaus, Utaminingrum, & Widasari, 2023).

Penelitian ini memperkenalkan pendekatan baru dengan menerapkan metode *Eye Aspect Ratio* (EAR) untuk mendeteksi rasa kantuk. EAR mengukur rasio jarak antara kelopak mata atas dan bawah sebagai indikator rasa kantuk, dengan mendeteksi perubahan bentuk mata yang biasanya dikaitkan dengan kelelahan atau peningkatan rasa kantuk (Amalia & Utaminingrum, 2021). Metode EAR terbukti efektif dalam mendeteksi kondisi mata tertutup dalam durasi tertentu, yang dianggap sebagai tanda awal rasa kantuk. Sistem yang diusulkan dalam penelitian ini menggunakan teknologi *computer vision* untuk memberikan pemantauan rasa kantuk secara *real-time*, tanpa memerlukan sensor fisik tambahan. Hal ini berbeda dengan metode yang menggunakan pengukuran fisiologis seperti denyut jantung atau elektrokardiogram yang memerlukan perangkat khusus (Kurniawan Umbu Nggiku, Rabi, & Subairi, 2023).

Penelitian ini bertujuan untuk merancang dan mengembangkan sistem pendeteksi kantuk menggunakan metode *Eye Aspect Ratio* (EAR) yang berfungsi secara *real-time* untuk memberikan peringatan dini saat pengemudi dalam kondisi mengantuk. EAR digunakan untuk mendeteksi perubahan rasio kelopak mata yang mengindikasikan kantuk berdasarkan perubahan durasi mata tertutup. Dengan sistem yang dapat bekerja secara otomatis dan terus menerus memantau kondisi pengemudi, peringatan dapat diberikan sebelum kantuk mencapai tingkat yang membahayakan. Sistem ini dirancang mudah diimplementasikan dan terjangkau,

tanpa memerlukan perangkat keras tambahan, sehingga ideal untuk digunakan di berbagai jenis kendaraan, baik konvensional maupun otonom.

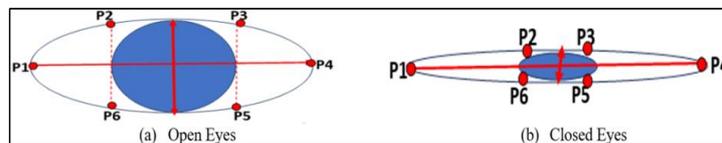
## 2. METODE PENELITIAN

### Metode *Eye Aspect Ratio* (EAR)

EAR merupakan suatu metode yang menghitung perbandingan antara jarak vertikal kelopak mata atas dan bawah terhadap jarak horizontal mata, dengan menggunakan titik-titik geometrik pada area mata yang diperoleh dari landmark wajah (Sugawara & Nikaido, 2014). Metode EAR ini menggunakan serangkaian titik pada wajah yang membentuk struktur mata untuk mengukur jarak antar kelopak mata, sehingga memungkinkan dilakukan analisis yang akurat terhadap kondisi mata saat terbuka atau tertutup. Metode ini menggunakan enam titik, yaitu P1, P2, P3, P4, P5, P6, untuk menentukan dimensi horizontal dan vertikal mata. Perhitungan EAR dirumuskan sebagai berikut:

$$EAR = \frac{\|p_2 - p_6\| + \|p_3 - p_5\|}{2\|p_1 - p_4\|}$$

**Gambar 1.** Rumus *Eye Aspect Ratio*



**Gambar 2.** *Eye Aspect Ratio* untuk Mata Terbuka dan Mata Tertutup

Berdasarkan rumus pada Gambar 1, menunjukkan rumus dasar EAR yang digunakan untuk menghitung perbandingan antara dimensi vertikal dan horizontal mata yang merupakan indikator utama dalam mendeteksi rasa kantuk. Pada perhitungan ini, titik P1, P2, P3, P4, P5, P6 berfungsi sebagai penanda lokasi pada koordinat 2D wajah, untuk mendeteksi lebar dan tinggi mata, seperti yang ditunjukkan pada Gambar 2. mengilustrasikan perbedaan nilai EAR pada kondisi mata terbuka dan tertutup yang mendasari penggunaan ambang batas untuk mendeteksi pengemudi yang mengantuk. Visualisasi ini penting untuk memudahkan pembaca memahami dasar perhitungan EAR.

Implementasi sistem deteksi kelelahan pengemudi berbasis pengukuran kedipan mata dengan metode *Eye Aspect Ratio* (EAR) difokuskan pada analisis dan deteksi mata. Pada setiap frame video yang diambil secara real-time, digunakan *library* dlib dan OpenCV untuk mendeteksi posisi mata. Langkah pertama dalam proses ini adalah deteksi wajah yang

bertujuan untuk menentukan area mata. Kemudian, dilakukan ekstraksi *landmark* wajah untuk mengidentifikasi sudut dan garis kelopak mata. Untuk menentukan kondisi tersebut, persamaan yang dapat digunakan dan saat ini sangat akurat adalah dengan menggunakan rasio panjang dan lebar *landmark* mata (EAR), dengan persamaan seperti pada Gambar 2 yang menunjukkan posisi titik-titik pada mata dalam kondisi terbuka dan tertutup:

a. P1 dan P4:

1. P1: Sudut luar mata kiri.
2. P4: Sudut luar mata kanan

Kedua titik ini membentuk garis horizontal yang menunjukkan lebar mata. Jarak horizontal antara P1 dan P4 digunakan sebagai pembanding dalam menghitung EAR. Titik ini menunjukkan seberapa lebar mata terbuka, yang merupakan salah satu komponen dalam menghitung rasio EAR.

b. P2 dan P6:

1. P2: Titik tengah atas kelopak mata kiri.
2. P6: Titik tengah bawah kelopak mata kiri.

Jarak antara P2 dan P6 digunakan untuk mengukur tinggi vertikal di sisi kiri mata. Jarak ini menunjukkan seberapa banyak kelopak mata kiri terangkat atau turun saat berkedip. Perubahan jarak ini memungkinkan sistem mendeteksi apakah mata terbuka atau tertutup.

c. P3 dan P5:

1. P3: Titik tengah atas kelopak mata kanan.
2. P5: Titik tengah bawah kelopak mata kanan.

Jarak antara P3 dan P5 digunakan untuk mengukur tinggi vertikal di sisi kanan mata. Jarak ini mencerminkan gerakan kelopak mata kanan saat berkedip. Seperti halnya sisi kiri, perubahan jarak ini digunakan untuk menghitung EAR dan mendeteksi kondisi mata.

### **Pemrograman Python untuk Metode yang Diusulkan**

```
1 import cv2
2 import dlib
3 import numpy as np
4 import os
5 from scipy.spatial import distance
6 from imutils import face_utils
7 import pygame
8 from datetime import datetime
9 import csv
10 import pandas as pd
11 import matplotlib.pyplot as plt
```

**Gambar 3.** Paket yang Terinstal

Gambar 3 menunjukkan paket-paket utama yang digunakan sebagai dasar penerapan metode deteksi kantuk dalam penelitian ini. Bahasa pemrograman Python dipilih karena kemampuannya yang fleksibel dan dukungan terhadap pustaka-pustaka penting untuk pemrosesan gambar, analisis gambar, dan peringatan waktu nyata. Masing-masing pustaka ini memainkan peran penting dalam mempercepat pengembangan dan meningkatkan akurasi hasil deteksi. Berikut ini adalah penjelasan singkat tentang pustaka-pustaka utama yang digunakan dan perannya masing-masing dalam penerapan metode ini:

a. OpenCV

*Open Computer Vision (OpenCV)* merupakan pustaka sumber terbuka yang dirancang khusus untuk pemrosesan citra digital, yang memungkinkan komputer meniru cara manusia memproses penglihatan. OpenCV menyediakan berbagai algoritma dasar di bidang visi komputer, termasuk modul untuk deteksi objek menggunakan teknik visi komputer (Zulkhaidi, Maria, & Yulianto, 2020). Dalam sistem ini, OpenCV digunakan untuk menangkap citra dari kamera secara real-time dan mendeteksi fitur wajah, seperti mata dan wajah pengemudi, yang kemudian digunakan dalam menghitung EAR (*Eye Aspect Ratio*) sebagai indikator rasa kantuk. Selain fitur-fitur tersebut, OpenCV mendukung berbagai format gambar dan video, serta memiliki integrasi yang kuat dengan pustaka pembelajaran mesin lainnya. Hal ini menjadikannya alat yang serbaguna dalam mengembangkan aplikasi berbasis visi komputer untuk berbagai kebutuhan, mulai dari keamanan, pengawasan, hingga teknologi otomotif.

b. Dlib

Dlib merupakan pustaka sumber terbuka yang menyediakan lingkungan optimal untuk mengembangkan perangkat lunak berbasis pembelajaran mesin dalam C++. Pustaka ini dibangun dengan dukungan aljabar linier melalui *Basic Linear Algebra Subprogram (BLAS)*, yang merupakan fondasi bagi banyak fungsi komputasinya. Dlib sering digunakan untuk implementasi jaringan Bayesian dan algoritma berbasis kernel, yang berperan dalam berbagai tugas seperti klasifikasi, pengelompokan, deteksi anomali, regresi, dan pemeringkatan fitur (Sharma, Shanmugasundaram, & Ramasamy, 2016). Dalam penelitian ini, Dlib digunakan untuk melacak titik-titik pada mata pengemudi yang diperlukan dalam perhitungan EAR, untuk menentukan kondisi mata terbuka atau tertutup.

c. NumPy

NumPy merupakan pustaka utama untuk pemrograman array dalam bahasa Python dan berperan penting dalam analisis data ilmiah di berbagai bidang. NumPy menyediakan array multidimensi yang efisien dan fungsi manipulasi data yang cepat, yang menjadi dasar

bagi pustaka ilmiah lainnya seperti SciPy dan Pandas (Harris et al., 2020). Dalam sistem ini, NumPy digunakan untuk mengoperasikan data gambar dan titik acuan mata, sehingga menyederhanakan perhitungan jarak antar titik mata yang diperlukan untuk menghitung *Eye Aspect Ratio* (EAR).

d. OS

Pustaka OS berfungsi untuk membersihkan layar terminal yang telah digunakan sebelumnya agar kembali kosong. Membersihkan terminal akan membuat output lebih mudah dilihat (Ilham & Saputra, 2023). OS digunakan untuk membuat direktori penyimpanan hasil deteksi, menghapus file lama, dan menyimpan hasil EAR dalam format CSV pada sistem ini. Modul `os` Python menyediakan antarmuka antara sistem operasi dan program. Penggunaan modul `os` memudahkan interaksi dengan sistem operasi dan meningkatkan portabilitas kode (Chen, 2023).

e. SciPy

SciPy memperluas fungsionalitas NumPy dengan menyediakan alat tambahan untuk komputasi ilmiah (Virtanen et al., 2020). Dalam sistem ini, SciPy digunakan untuk menghitung jarak antara titik-titik pada mata yang diperlukan dalam perhitungan EAR. SciPy mengintegrasikan fungsi matematika dan statistik tingkat lanjut ke dalam lingkungan Python, menawarkan perangkat lengkap untuk mengatasi masalah kompleks secara efisien dan intuitif (Rayhan & Kinzler, 2023).

f. Imutils

Imutils adalah pustaka yang digunakan dalam proyek untuk membaca dan memproses bingkai gambar secara bersamaan; ini mengurangi latensi pemrosesan untuk aplikasi yang memerlukan respons cepat, seperti pengenalan objek dalam sistem transportasi (Shammi, Sultana, Islam, & Chakrabarty, 2018). Sistem menggunakan Imutils untuk mengubah ukuran gambar dan memosisikannya agar sesuai dengan tampilan deteksi kantuk.

g. PyGame

Pygame adalah pustaka Python yang dirancang untuk membuat *game*. Dengan Pygame, Anda dapat dengan mudah membuat *game* 2D seperti *game platformer*, *game arcade*, *game puzzle*, dan *game* bermain peran (RPG). Pygame menyediakan berbagai fitur seperti kemampuan untuk menampilkan grafik, mengelola acara, memutar audio dan video, dan banyak lagi (Saragih, Lumbantoruan, Armando, & Junita, 2023). Ketika rasa kantuk terdeteksi (nilai EAR turun di bawah ambang batas), suara peringatan akan diputar melalui pustaka PyGame dan dihentikan ketika kondisi kembali normal. Hal ini membuat sistem lebih efektif dengan kombinasi peringatan visual dan audio.

#### h. Datetime

Dengan menggunakan *"import datetime"*, pengguna dapat mengubah tanggal dan waktu dalam kode program dengan modul *"datetime"* yang tersedia dalam Python. Modul ini sangat mudah digunakan untuk bekerja dengan tanggal dan waktu dalam kode program (Amelina & Muhammad, 2024). Dalam sistem ini, setiap kali nilai EAR dicatat, *Datetime* digunakan untuk mencatat waktu, yang memungkinkan analisis waktu untuk mendeteksi rasa kantuk.

#### i. CSV

Data tabel terstruktur dapat dengan mudah dipertukarkan dan disimpan menggunakan file CSV, yang merupakan file teks biasa. Setiap baris tabel biasanya berisi catatan data yang berbeda (Mäs et al, 2018). Dalam penggunaan ini, CSV mencatat nilai EAR dan data rasa kantuk ke dalam file CSV, yang memungkinkan analisis atau visualisasi data lebih lanjut.

#### j. Pandas

Pandas (Python Data Analysis) adalah pustaka pemrosesan data dengan domain pemrograman Python. Pustaka ini memiliki fitur data frame yang dapat membantu dalam mengelola data berbasis tabel atau informasi array (Albanna, 2018). Pandas berfungsi untuk mengolah dan menganalisis data, seperti menyimpan, memanipulasi, atau membaca data EAR yang terekam dalam format CSV.

#### k. Matplotlib

Matplotlib yang dikenal sebagai *plotting library* dalam Python digunakan untuk menghasilkan visualisasi grafis dua dimensi dan tiga dimensi yang interaktif (Hunter, 2007). Dalam sistem tersebut, matplotlib menjadi komponen penting dari sistem pendeteksi kantuk ini karena memungkinkan visualisasi data *Eye Aspect Ratio* (EAR) secara real-time, yang memungkinkan untuk memantau perubahan nilai EAR yang mengindikasikan kondisi kantuk pengemudi. Dengan menggunakan Matplotlib, data yang diperoleh dapat dianalisis lebih lanjut untuk mendapatkan pemahaman yang lebih baik tentang tanda-tanda kelelahan yang dialami oleh pengemudi.

```

1  # Calculate EAR
2  def eye_aspect_ratio(eye):
3      A = distance.euclidean(eye[1], eye[5])
4      B = distance.euclidean(eye[2], eye[4])
5      C = distance.euclidean(eye[0], eye[3])
6      return (A + B) / (2.0 * C)

```

**Gambar 4.** Fungsi Python untuk Perhitungan EAR

Gambar 4 menunjukkan kode Python yang mengimplementasikan metode *Eye Aspect Ratio* (EAR), sebuah teknik yang banyak digunakan dalam mendeteksi rasa kantuk berdasarkan posisi mata. EAR mengukur rasio aspek mata untuk mendeteksi apakah mata terbuka atau tertutup. Rasio ini dihitung berdasarkan jarak antara beberapa titik yang mewakili fitur geometris mata, yang biasanya diperoleh melalui algoritma deteksi wajah dan landmark.

```
1 # Initialize DLIB models
2 face_detector = dlib.get_frontal_face_detector()
3 landmark_predictor = dlib.shape_predictor("models/shape_predictor_68_face_landmarks.dat")
4 lStart, lEnd = 36, 42
5 rStart, rEnd = 42, 48
6 threshold = 0.25
```

**Gambar 5.** Kode Landmark Wajah

Gambar 5 menunjukkan jalur ke detektor titik wajah pra-terlatih Dlib, Dlib mendeteksi titik-titik wajah penting untuk mengidentifikasi rahang, mulut, hidung, alis, dan mata. Ini adalah prosedur yang termasuk dalam prediksi bentuk dan bertujuan untuk menentukan posisi titik-titik utama pada objek yang dianalisis. Gambar biasanya dipusatkan pada area wajah tertentu (Region of Interest atau ROI), dan prediktor bentuk menemukan titik-titik penting yang terkait dengan bentuk wajah. Kode ini menggunakan Dlib untuk mendeteksi wajah dan menggunakan model landmark pra-terlatih untuk memprediksi wajah. Aplikasi seperti pengenalan wajah, pelacakan ekspresi, dan deteksi kantuk memerlukan proses ini untuk mengidentifikasi posisi titik-titik wajah secara akurat.

```
1 if ear < threshold:
2     alert_counter += 1
3     if alert_counter >= 10:
4         drowsiness_flag = True
5         drowsiness_timer = 30
6         if not pygame.mixer.get_busy():
7             alert_sound.play()
8     else:
9         alert_counter = 0
10        drowsiness_flag = False
11        drowsiness_timer = 0
12        blink_state = False # Stop blinking immediately when no drowsiness detected
13        if pygame.mixer.get_busy():
14            alert_sound.stop()
```

**Gambar 6.** Pengkondisian Nilai EAR Berdasarkan Ambang Batas

Gambar 6 bertujuan untuk mendeteksi tanda-tanda kelelahan pengemudi atau pengguna melalui pemantauan kondisi mata (seperti berkedip atau telinga, yang dapat merujuk ke sensor). Ketika nilai EAR yang dihitung lebih kecil dari ambang batas yang telah ditentukan ( $\leq 0,25$ ), sistem akan mulai meningkatkan `alert_counter`. Penghitung ini bertambah setiap kali kondisi EAR di bawah ambang batas terdeteksi. Ketika `alert_counter`

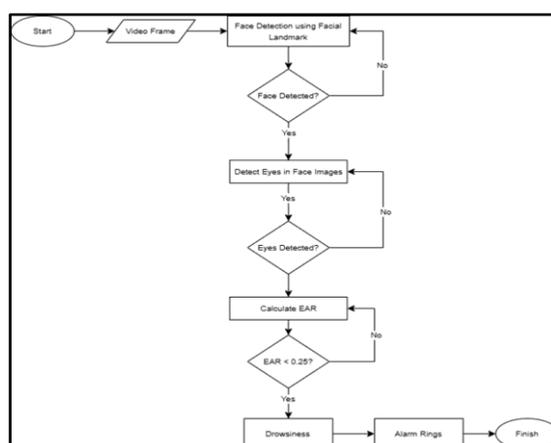
mencapai nilai 10, yang menunjukkan bahwa pengemudi mungkin mengalami kantuk parah, sistem akan mengaktifkan `sleep_flag` sebagai indikator kantuk.

Pada saat yang sama, peringatan suara akan diputar melalui `pygame.mixer` untuk mengingatkan pengemudi agar tetap terjaga. Peringatan suara ini berlanjut hingga kondisi EAR kembali normal. Jika nilai EAR kembali di atas ambang batas sebelum `alert_counter` mencapai 10, penghitung akan diatur ulang kembali ke 0 dan peringatan tidak akan diaktifkan. Selain itu, jika nilai `sleeping_flag` telah diaktifkan tetapi nilai EAR menunjukkan normalisasi (di atas ambang batas), `sleeping_flag` akan dimatikan, `alert_counter` akan diulang, dan peringatan suara akan berhenti secara otomatis. Pendekatan ini bertujuan untuk meningkatkan kewaspadaan pengemudi melalui mekanisme peringatan suara yang hanya diaktifkan ketika rasa kantuk terdeteksi secara konsisten selama jangka waktu tertentu, sehingga mengurangi potensi alarm palsu dalam kondisi normal.

### 3. HASIL DAN PEMBAHASAN

Penelitian ini bertujuan untuk mengetahui seberapa efektif sistem dalam mendeteksi rasa kantuk pengemudi menggunakan metode *Eye Aspect Ratio* (EAR) yang dilakukan secara real-time menggunakan pustaka OpenCV dan Dlib untuk mengidentifikasi mata dan menghitung nilai EAR. Selain itu, penelitian ini juga mengkaji seberapa efektif sistem ini dalam mendeteksi rasa kantuk dan memberikan peringatan kepada pengemudi saat rasa kantuk terdeteksi.

#### Proses Deteksi Kantuk



**Gambar 7.** Diagram Alur Kerja

Gambar 7 di atas menggambarkan alur kerja metode Eye Aspect Ratio (EAR) untuk mendeteksi kelelahan pengemudi. Sistem dimulai dengan merekam video wajah pengemudi secara real-time melalui kamera yang terpasang di kendaraan. Setiap frame dianalisis untuk mendeteksi wajah pengemudi. Setelah wajah teridentifikasi, sistem mendeteksi posisi mata

pengemudi di area wajah, yang menjadi dasar penghitungan nilai EAR untuk menilai kondisi mata pengemudi.

Sistem menggunakan nilai EAR untuk menentukan apakah pengemudi menunjukkan tanda-tanda kelelahan. Nilai EAR dihitung dari rasio jarak antara titik-titik di sekitar mata, dan ambang batas yang ditetapkan adalah 0,25. Jika nilai EAR di bawah ambang batas ini, sistem menganggap mata pengemudi tertutup atau hampir tertutup, yang merupakan indikasi kantuk. Jika kondisi ini terdeteksi berulang kali dalam jangka waktu tertentu, sistem secara otomatis mengaktifkan alarm sebagai peringatan dini bagi pengemudi agar lebih waspada. Dengan alarm ini, pengemudi diharapkan untuk segera mengambil tindakan, seperti beristirahat atau menyesuaikan posisi duduk untuk meningkatkan fokus. Proses ini dirancang sebagai tindakan preventif untuk mengurangi risiko kecelakaan akibat kantuk, sehingga keselamatan pengemudi dan pengguna jalan lainnya dapat lebih terjaga.

### **Hasil Pengukuran EAR dan Visualisasi Deteksi Kantuk**

Pengukuran EAR dilakukan untuk mengidentifikasi pola perubahan nilai sebagai indikator kantuk pengemudi. EAR dihitung berdasarkan rasio jarak vertikal kelopak mata atas dan bawah terhadap jarak horizontal mata, dan nilai ini dipantau secara real-time. Hasil pengukuran menunjukkan nilai EAR rata-rata sekitar 0,30 saat mata terbuka, yang menunjukkan keadaan waspada. Saat mata setengah tertutup, nilai EAR turun menjadi 0,27, yang menunjukkan kewaspadaan menurun. Saat mata tertutup sepenuhnya, nilai EAR rata-rata mencapai 0,20, yang menunjukkan pengemudi sangat mengantuk.

Dalam penelitian ini, ambang batas EAR adalah 0,25. Jika nilai EAR secara konsisten turun di bawah ambang batas ini dalam beberapa frame berturut-turut, sistem akan mendeteksi kantuk dan memicu alarm sebagai peringatan dini.

**Tabel 1.** Nilai EAR Rata-rata Berdasarkan Kondisi Mata

| Kondisi Mata          | Rata-rata Nilai EAR |
|-----------------------|---------------------|
| Mata Terbuka          | 0.30                |
| Mata Setengah Terbuka | 0.27                |
| Mata Tertutup         | 0.20                |

Dari data di atas, nilai EAR menurun seiring dengan derajat penutupan mata. Saat mata terbuka penuh, nilai EAR rata-rata berada di sekitar 0,30, yang menunjukkan kondisi waspada, saat mata setengah tertutup, nilai EAR rata-rata menurun hingga 0,27, yang menunjukkan

penurunan kewaspadaan, dan saat mata tertutup penuh, nilai EAR rata-rata mencapai 0,20, yang menunjukkan rasa kantuk yang signifikan.



**Gambar 8.** Deteksi Mata Normal

Gambar 8 menunjukkan deteksi mata dalam situasi normal, saat sistem mendeteksi mata terbuka. Akibat nilai EAR yang terukur di atas ambang batas 0,25, sistem tidak mengaktifkan alarm. Dalam kondisi pengemudi waspada, nilai EAR tidak menunjukkan tanda-tanda kantuk dan berada dalam kisaran normal, seperti yang ditunjukkan pada gambar ini.



**Gambar 9.** Deteksi Mata dalam Kondisi Tidur

Gambar 9 menunjukkan situasi saat pengemudi mengalami *microsleep* atau kondisi mengantuk sesaat, yang ditunjukkan oleh nilai EAR yang secara konsisten berada di bawah ambang batas 0,25 selama beberapa *frame* berturut-turut. Dalam situasi seperti itu, sistem mengaktifkan alarm untuk memberikan peringatan dini agar pengemudi sadar dan lebih waspada. Gambar ini menunjukkan seberapa baik sistem mendeteksi kondisi kantuk yang berbahaya.



**Gambar 10.** Tren Nilai EAR dan Deteksi Kantuk

Gambar 10 memberikan visualisasi yang jelas tentang tren nilai EAR selama kurun waktu tertentu, yang membantu dalam memantau pola kantuk pengemudi. Dengan menggunakan visualisasi ini, kita dapat mengidentifikasi saat-saat kritis ketika pengemudi berada dalam kondisi kantuk yang parah, serta mengukur efektivitas sistem dalam memberikan peringatan dini secara konsisten. Visualisasi ini memberikan pemahaman yang lebih komprehensif tentang bagaimana sistem mendeteksi dan merespons kantuk pengemudi dari waktu ke waktu. Setiap grafik pada Gambar 10 menggambarkan perubahan nilai EAR dari waktu ke waktu, dengan garis ambang batas (0,25) berfungsi sebagai indikator kantuk. Area berwarna di sekitar grafik menunjukkan saat nilai EAR berada di bawah ambang batas, yang menunjukkan kemungkinan kantuk pengemudi. Visualisasi ini penting untuk mengevaluasi efektivitas sistem dalam memberikan peringatan dini dan konsistensi deteksi kantuk dalam jangka panjang.

Pada grafik pertama, “*Raw Ear Values Over Time*”, nilai EAR terukur dari waktu ke waktu diwakili oleh garis biru, sedangkan garis merah horizontal menunjukkan ambang batas kantuk sebesar 0,25. Ketika nilai EAR turun di bawah garis merah ini, sistem menganggap bahwa pengemudi mungkin mengalami kantuk. Nilai EAR yang tetap di atas ambang batas menunjukkan keadaan pengemudi yang terjaga, sedangkan penurunan di bawah ambang batas menunjukkan tanda-tanda kantuk. Grafik ini memungkinkan analisis fluktuasi kondisi mata pengemudi secara langsung.

Grafik kedua, “*Raw Ear Values Over Time*” memberikan gambaran yang lebih rinci tentang rasa kantuk. Area yang diwarnai dengan intens (merah) menunjukkan waktu ketika nilai EAR berada di bawah ambang batas untuk jangka waktu yang lebih lama, yang menunjukkan rasa kantuk yang lebih parah atau berpotensi membahayakan bagi pengemudi. Variasi warna, seperti hijau dan kuning, menunjukkan tingkat kewaspadaan yang lebih tinggi, sedangkan merah menunjukkan kelelahan yang lebih intens. Grafik ini membantu mengidentifikasi momen kritis yang memerlukan intervensi segera.

Grafik ketiga, “*Tiredness Ear Values Over Time*” menampilkan EAR deteksi kelelahan jangka panjang. Grafik ini memberikan gambaran yang lebih jelas tentang durasi dan frekuensi pengemudi mengalami kondisi kantuk. Setiap kali nilai EAR turun di bawah ambang batas untuk jangka waktu tertentu, sistem akan memberi tahu pengemudi. Dengan demikian, grafik ini berguna untuk mengevaluasi pola kantuk secara keseluruhan dan mengidentifikasi perubahan signifikan dalam kondisi kewaspadaan pengemudi seiring waktu.

Secara keseluruhan, Gambar 10 memberikan gambaran mendalam tentang kinerja sistem dalam mendeteksi dan menanggapi rasa kantuk pengemudi. Visualisasi tren EAR juga memungkinkan analisis pengaruh faktor eksternal dan internal yang memengaruhi

kewaspadaan pengemudi. Hasil ini mendukung penilaian keakuratan dan efektivitas sistem dalam mengurangi risiko kecelakaan akibat kantuk dengan memberikan peringatan dini secara konsisten.

## **Pembahasan**

Hasil implementasi sistem deteksi kantuk menggunakan metode *Eye Aspect Ratio* (EAR) menunjukkan bahwa metode ini efektif dalam memberikan peringatan dini kepada pengemudi. Dengan menggunakan ambang batas EAR sebesar 0,25, sistem dapat mendeteksi kantuk secara akurat. Ketika nilai EAR pengemudi turun di bawah ambang batas tersebut, matanya mulai tertutup, yang merupakan tanda awal kantuk. Dengan kemampuan sistem ini dalam memberikan peringatan dengan cepat, pengemudi dapat segera mengambil tindakan pencegahan seperti beristirahat atau memperbaiki postur tubuh untuk meningkatkan kewaspadaan.

Selain itu, implementasi sistem ini dapat dilakukan dengan menggunakan kamera standar yang terpasang di dasbor kendaraan. Sistem ini dapat bekerja secara otomatis, memberikan peringatan dini kepada pengemudi yang mengantuk, dan dapat diintegrasikan dengan perangkat lunak yang ada di kendaraan tanpa memerlukan perangkat keras tambahan yang mahal. Dengan demikian, sistem ini sangat cocok untuk kendaraan komersial maupun pribadi.

Metode EAR memiliki beberapa keunggulan dibandingkan metode deteksi kantuk lainnya yang lebih kompleks, seperti mengukur aktivitas otak atau mendeteksi gerakan kepala. Beberapa di antaranya adalah biaya yang lebih rendah, proses yang mudah, dan tidak memerlukan perangkat keras tambahan. Sistem dapat mendeteksi kantuk menggunakan kamera standar dan algoritma pemrosesan gambar. Namun, metode ini memiliki beberapa keterbatasan, seperti sensitivitas yang kurang pada pengemudi yang memakai kaca mata atau pengemudi yang memiliki kebiasaan berkedip yang berbeda. Selain itu, metode ini mungkin sulit mendeteksi kelelahan pada pengemudi dengan gerakan mata yang rendah.

## **4. KESIMPULAN DAN SARAN**

Dengan ambang batas EAR sebesar 0,25, sistem deteksi kantuk berdasarkan *Eye Aspect Ratio* (EAR) secara efektif memperingatkan pengemudi saat mereka mengantuk, terutama saat nilai EAR turun di bawah ambang batas ini dan mata mulai terpejam. Peringatan aktif, yang disampaikan melalui suara, dimaksudkan untuk mendorong pengemudi agar beristirahat atau menyesuaikan postur tubuh mereka guna meningkatkan kewaspadaan.

Penelitian ini menunjukkan bahwa metode EAR dapat dengan mudah diintegrasikan ke dalam berbagai jenis kendaraan hanya dengan menggunakan kamera dan algoritma pemrosesan gambar sederhana. Dengan demikian, metode EAR merupakan pilihan yang sesuai untuk aplikasi di dunia nyata pada kendaraan, terutama bagi pengemudi yang mencari cara untuk mengurangi risiko kecelakaan yang disebabkan oleh rasa kantuk.

Penelitian di masa mendatang dapat difokuskan pada pengembangan algoritma yang lebih adaptif untuk deteksi kantuk dengan mempertimbangkan faktor-faktor individual seperti pola kedipan mata atau penggunaan kacamata. Selain itu, teknologi sensor yang lebih canggih, seperti sensor biometrik atau deteksi gerakan kepala, dapat diintegrasikan untuk meningkatkan akurasi dan keandalan deteksi kantuk. Dengan keterbatasan ini, pengembangan lebih lanjut dari sistem deteksi kantuk dapat difokuskan pada penggabungan teknologi tambahan atau penggunaan algoritma yang lebih adaptif untuk meningkatkan akurasi dan jangkauan deteksi. Temuan ini menunjukkan bahwa metode EAR dapat menjadi tindakan pencegahan yang efektif untuk menjaga keselamatan pengemudi dan pengguna jalan lainnya. Studi ini juga menunjukkan bahwa metode ini memiliki potensi signifikan untuk dikembangkan lebih lanjut dalam aplikasi kendaraan modern.

## **DAFTAR REFERENSI**

- Albanna, I. (2018). Implementasi Pandas Data frame sebagai Agregasi dan Tabulasi Penyajian Data Luaran Survei Kepuasan Pengguna Proses Pembelajaran dalam Pendidikan Tinggi. *Proceedings of AGILE*.
- Amalia, D., & Utaminingrum, F. (2021). Deteksi Kantuk pada Pengemudi melalui Jumlah Kedipan Mata Menggunakan Facial Landmark berbasis Intel NUC. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 5(12), 5529–5535.
- Amelina, M., & Muhammad, R. A. N. (2024). Pengenalan Wajah Mahasiswa Untuk Absensi Perkuliahan Menggunakan Machine Learning. *Journal Technology of Computer*, 1(1), 1–10. <http://ojs.adzkie.ac.id/index.php/jtech>
- Chen, X. (2023). Introduction and Analysis of Python Software. *Frontiers in Computing and Intelligent Systems*, 5(2), 41–43. <https://doi.org/10.54097/fcis.v5i2.12348>
- Firdaus, A., Utaminingrum, F., & Widasari, E. R. (2023). Sistem Pendeteksi Kantuk Pengemudi berbasis Eye Aspect Ratio dan Mouth Opening Ratio menggunakan Algoritme C-LSTM. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 7(2), 927–933. <http://j-ptiik.ub.ac.id>
- Harris, C. R., Millman, K. J., Van Der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, 585(7825), 357–362. <https://doi.org/10.1038/s41586-020-2649-2>

- Huda, C. (2020). DETEKSI KANTUK MEMANFAATKAN FACIAL LANDMARK BERDASARKAN ANALISIS NILAI EAR DAN BOBOT KECERAHAN PADA PENGENDARA MOBIL BERBASIS PERANGKAT BERGERAK. *Magister Thesis, Universitas Brawijaya*. <http://repository.ub.ac.id/id/eprint/178376/>
- Hunter, J. D. (2007). Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering*, 9(3), 90–95. <https://doi.org/10.1109/MCSE.2007.55>
- Ilham, M. F. N., & Saputra, A. (2023). Seleksi Penerimaan Mahasiswa Baru Dengan Metode Pemecahan Masalah Algoritma Greedy Menggunakan Python. *Jurnal Rekayasa Teknologi Informasi (JURTI)*, 7(1), 32. <https://doi.org/10.30872/jurti.v7i1.9566>
- Ilmadina, H. Z., Apriliani, D., & Wibowo, D. S. (2022). Deteksi Pengendara Mengantuk dengan Kombinasi Haar Cascade Classifier dan Support Vector Machine. *Jurnal Informatika: Jurnal Pengembangan IT*, 7(1), 1–7. <https://doi.org/10.30591/jpit.v7i1.3346>
- Kurniawan Umbu Nggiku, C., Rabi, A., & Subairi, S. (2023). Deteksi Kantuk Untuk Keamanan Berkendara Berbasis Pengolahan Citra. *Jurnal JEETech*, 4(1), 48–56. <https://doi.org/10.32492/jeetech.v4i1.4107>
- Ramadhani, N., Aulia, S., Suhartono, E., & Hadiyoso, S. (2021). Deteksi Kantuk pada Pengemudi Berdasarkan Penginderaan Wajah Menggunakan PCA dan SVM. *Jurnal Rekayasa Elektrika*, 17(2), 1–10. <https://doi.org/10.17529/jre.v17i2.19884>
- Rayhan, A., & Kinzler, R. (2023). *Advancing Scientific Computing with Python's SciPy Library*. <https://doi.org/10.13140/RG.2.2.21131.87841>
- S. Mäs, D. Henzen, L. Bernard, M. Müller, S. Jirka, dan I. Senner. (2018). "Generic Schema Descriptions for Comma-Separated Values Files of Environmental Data". Proceedings of AGILE 2018 – Lund, June 12-15.
- Saragih, Y. W., Lumbantoruan, S. K., Armando, E., & Junita, J. (2023). Penerapan library pygame dalam game RPG“the adventure.” *Jurnal Manajemen Informatika Jayakarta*, 3(1), 57–73. <https://doi.org/10.52362/jmijayakarta.v3i1.1013>
- Sasono, A. (2023). SISTEM PENDETEKSI KANTUK PADA PENGENDARA MOBIL BERBASIS IMAGE PROCESSING. *Universitas Muhamadiyah Surakarta*.
- Shammi, S. K., Sultana, S., Islam, Md. S., & Chakrabarty, A. (2018). Low Latency Image Processing of Transportation System Using Parallel Processing co-incident Multithreading (PPcM). *2018 Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2018 2nd International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*, 363–368. Kitakyushu, Japan: IEEE. <https://doi.org/10.1109/ICIEV.2018.8640957>
- Sharma, S., Shanmugasundaram, K., & Ramasamy, S. K. (2016). FAREC — CNN based efficient face recognition technique using Dlib. *2016 International Conference on Advanced Communication Control and Computing Technologies (ICACCCT)*, 192–195. Ramanathapuram, India: IEEE. <https://doi.org/10.1109/ICACCCT.2016.7831628>

- Sugawara, E., & Nikaido, H. (2014). Properties of AdeABC and AdeIJK Efflux Systems of *Acinetobacter baumannii* Compared with Those of the AcrAB-TolC System of *Escherichia coli*. *Antimicrobial Agents and Chemotherapy*, 58(12), 7250–7257. <https://doi.org/10.1128/AAC.03728-14>
- Suradi, A. A. M., Alam, S., & Rasyid, M. F. (2023). Sistem Deteksi Kantuk Pengemudi Mobil Berdasarkan Analisis Rasio Mata Menggunakan Computer Vision. *Jurnal Komputer dan Informatika*, 5(2), 222–223. <https://doi.org/10.53842/juki.v5i2.269>.
- Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Vázquez-Baeza, Y. (2020). SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods*, 17(3), 261–272. <https://doi.org/10.1038/s41592-019-0686-2>
- Zulkhaidi, T. C. A.-S., Maria, E., & Yulianto, Y. (2020). Pengenalan Pola Bentuk Wajah dengan OpenCV. *Jurnal Rekayasa Teknologi Informasi (JURTI)*, 3(2), 181. <https://doi.org/10.30872/jurti.v3i2.4033>