

RESEARCH ARTICLE

Aplikasi Pendeteksi Fokus Mahasiswa Dalam Melakukan Pembelajaran Menggunakan Mimik Muka

Wahyu Esya Nasution, Prajna Deshanta Ibnugraha* and Muhammad Rizqy Alfarisi

Fakultas Ilmu Terapan, Universitas Telkom, Bandung, 40257, Jawa Barat, Indonesia

* Corresponding author: prajna@telkomuniversity.ac.id

Received on 17 August 2023; accepted on 20 September 2023

Abstrak

Masa pandemi Covid-19 membuat metode pembelajaran yang awalnya berbasis luring harus berubah menjadi daring. Hal tersebut tentunya memiliki pengaruh terhadap tingkat fokus mahasiswa, mahasiswa yang memiliki fokus rendah dapat membuat mahasiswa tersebut mendapat nilai yang kurang baik. Dosen juga akan merasa kesusahan mengawasi satu persatu mahasiswanya pada pembelajaran berbasis daring. Berdasarkan masalah tersebut diadakan penelitian tentang sistem yang dapat mendeteksi fokus mahasiswa dalam melakukan pembelajaran. Sistem tersebut dapat mendeteksi keadaan mahasiswa yang sedang mengantuk ataupun yang sedang fokus pada saat pembelajaran. Hasil penelitian ini menunjukkan bahwa pendeteksian wajah menggunakan metode *Haar Cascade Classifier* dapat mendeteksi wajah saat kondisi wajah tidak terhalang suatu benda dan memiliki kondisi cahaya yang mencukupi. Orang yang sedang tidak fokus sering menutup mata lebih dari 14 kali dan menguap lebih dari 4 kali. Sedangkan orang yang sedang berada dalam kondisi fokus jarang menutup mata dan menguap. Didapatkan pula tingkat akurasi sebesar 80% terhadap pendeteksian yang dilakukan terhadap 10 orang dalam jangka waktu 5 menit.

Key words: *Haar Cascade Classifier*, Fokus, Pendeteksian Wajah, dan *Image Processing*

Pendahuluan

Indonesia merupakan salah satu negara yang terkena efek pandemi COVID-19. Pandemi ini memengaruhi segala aspek kehidupan manusia, salah satu aspek yang terpengaruh adalah aspek pendidikan. Pemerintah menerapkan kebijakan *Work From Home* (WFH) untuk membatasi interaksi sosial juga bertujuan untuk mengurangi dan memutus rantai pandemi COVID-19. Penerapan kebijakan tersebut menyebabkan pembelajaran yang awalnya berbasis luring harus berubah menjadi pembelajaran berbasis *internet* atau daring [1].

Metode pembelajaran yang dilakukan secara daring dapat dengan mudah membuat mahasiswa memasuki masa kejenuhan atau mengantuk. Salah satu hal yang membuat mengantuk adalah kekurangan tidur; kekurangan tidur dapat menurunkan kinerja, perhatian berkelanjutan atau fokus, dan mengakibatkan waktu respon yang lambat [2]. Menurut teori *microsleep*, salah satu tanda mengantuk adalah pada saat seseorang sering menutup mata [3]. Hal tersebut tentunya dapat membuat nilai mahasiswa menurun dan mempersulit dosen untuk mengawasi mahasiswa secara menyeluruh [4]. Oleh karena itu, penelitian aplikasi pendeteksi fokus mahasiswa ini dapat memberikan kemudahan bagi dosen untuk mengawasi mahasiswanya. Metode yang digunakan pada penelitian ini adalah metode *Haar Cascade Classifier* dimana metode

ini bergantung pada jumlah piksel sehingga memiliki komputasi yang cepat [5]. Dengan metode ini dan bantuan dari library *dlib* Python diharapkan dapat mendeteksi kondisi fokus mahasiswa dalam melakukan pembelajaran. Tujuan penelitian ini adalah membuat aplikasi pendeteksi fokus mahasiswa melalui mimik muka yang dapat mendeteksi gerakan mata dan mulut. Mengukur nilai rasio mata tertutup, mulut menguap, tingkat akurasi fokus mahasiswa berdasarkan gerakan kelopak mata dan mulut, serta jarak antara wajah dan kamera pada sistem juga merupakan tujuan penelitian ini [6].

Tinjauan Pustaka

A. Tinjauan Pustaka

Penelitian ini mengambil berbagai informasi dari penelitian sebelumnya sebagai bahan perbandingan dan sumber referensi. Pada penelitian berjudul "Deteksi Wajah Menggunakan Metode *Haar Cascade Classifier* Berbasis Webcam Pada Matlab" yang ditulis oleh Suhepy Abidin bertujuan untuk mengimplementasikan metode *Haar Cascade Classifier* kepada sebuah aplikasi pendeteksi wajah. Dengan dilakukannya berbagai uji coba berdasarkan variasi sudut pengambilan citra wajah,

intensitas cahaya, dan pengujian jarak *webcam* dengan wajah didapatkan kesimpulan bahwa metode *Haar Cascade Classifier* sangat ideal untuk pendeteksian wajah secara *realtime* [1].

Penelitian selanjutnya yang berjudul "Pendeteksian Kantuk Secara Real Time Menggunakan Pustaka OPENCV dan DLIB PYTHON" yang ditulis oleh Afrizal Zein bertujuan melakukan pendeteksian hanya berdasarkan daerah sekitar mata dengan menggunakan OpenCV dan Dlib Python. Penelitian tersebut menggunakan Dlib Python untuk menentukan landmark mata dan menggunakan Eye Aspect Ratio (EAR) untuk menentukan rasio kondisi mata saat sedang terbuka atau tertutup. Dengan dua aspek tersebut dapat diketahui kondisi pengguna saat sedang mengantuk atau tidak [2].

Pada penelitian yang berjudul "Deteksi Kantuk Pada Pengendara Roda Empat Melalui Citra Wajah Menggunakan Metode Facial Landmark" yang ditulis oleh Muhammad Akbar Maulana bertujuan melakukan pendeteksian kantuk menggunakan *Eye Aspect Ratio* (EAR) dan *Mouth Aspect Ratio* (MAR) untuk menentukan kondisi kantuk pada pengguna [3]. Pendeteksi kantuk juga dijelaskan pada penelitian yang ditulis Andrea Hartoko Aji Putra Perdana, dimana penelitian tersebut menggunakan facial landmark Dlib Python dan Eye Aspect Ratio sebagai penentu kondisi mata mengantuk [4]. Adapun penelitian mengenai Eye Aspect Ratio (EAR) untuk mendeteksi kedipan mata yang berjudul "Real-Time Eye Blink Detection using Facial Landmarks" yang ditulis oleh Tereza Soukupova and Jan Cech [5].

B. Dasar Teori

Open Source Computer Vision Library atau sering disingkat *OpenCV* merupakan software pengolahan citra dinamis secara *realtime* yang dibuat oleh Intel. *OpenCV* juga mendukung berbagai bahasa pemrograman seperti Python, Java, C, dan C++. *OpenCV* difokuskan pada aplikasi *realtime* dan efisiensi dalam komputasi [6].

Dlib merupakan sebuah pustaka detektor koordinat wajah yang sudah dilatih sebelumnya untuk memetakan struktur wajah. Dlib memiliki total sebanyak 68 titik koordinat wajah [6].

Eye Aspect Ratio atau disingkat EAR merupakan salah satu metode menghitung rasio jarak antara kelopak mata atas dengan kelopak mata bawah berdasarkan titik koordinat wajah untuk mencari nilai lebar mata. Nilai EAR didapatkan dengan menghitung jumlah jarak dua titik koordinat kelopak mata atas ditambah dengan jumlah jarak dua titik koordinat kelopak mata bawah, lalu dibagi dua kali jarak horizontal mata [7].

Mouth Aspect Ratio atau disingkat MAR merupakan salah satu metode menghitung rasio jarak antara bibir atas dengan bibir bawah untuk mencari nilai lebar mulut [7].

Microsleep adalah suatu kejadian hilangnya kesadaran karena mengantuk atau kelelahan. Durasi *microsleep* berkisar 3-5 detik, bahkan ada yang sampai 10 detik [3].

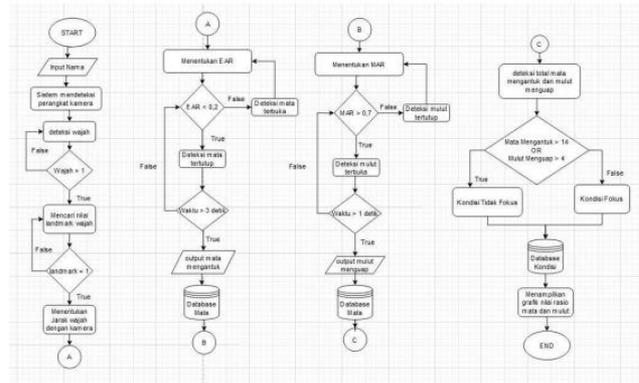
Matplotlib merupakan sebuah pustaka Python untuk membuat grafik 2 dimensi seperti plots, histograms, spectra, dll dengan tampilan yang dapat dikustomisasi dan memiliki kualitas yang tinggi [8]. Seaborn merupakan sebuah pustaka untuk melakukan visualisasi data yang indah tanpa kostumisasi rumit pada bahasa Python dan dibangun di bawah pustaka *matplotlib* [9].

Firebase merupakan NoSQL *database* yang ditampilkan menggunakan *Javascript Object Notation* (JSON). Setiap ada perubahan data, *Firebase* akan melakukan sinkronisasi dengan sangat cepat [10].

Metodologi Penelitian

Perancangan Sistem

Sistem pendeteksi fokus mahasiswa dimulai dengan memasukkan nama pengguna, kemudian sistem melakukan pendeteksian wajah menggunakan metode *Haar Cascade Classifier*. Ketika wajah sudah



Gambar 1. Flowchart sistem

terdeteksi, langkah selanjutnya adalah menentukan *landmark* menggunakan *library* Dlib Python. Kemudian sistem akan menentukan jarak sebenarnya antara wajah dengan kamera. Setelah itu, sistem akan melakukan perhitungan rasio mata dan rasio mulut menggunakan *Eye Aspect Ratio* (EAR) dan *Mouth Aspect Ratio* (MAR). Saat nilai rasio mata kurang dari 0,2, maka akan terdeteksi mata tertutup dan saat nilai rasio mata lebih dari 0,2 akan terdeteksi mata terbuka. Saat mata tertutup lebih dari 3 detik, maka akan keluar sebuah output kondisi mata mengantuk, dan setelah itu akan dikirim ke database. Setelah itu melakukan hal yang sama terhadap nilai rasio mulut, saat nilai rasio mulut lebih dari 0,7, maka akan terdeteksi mulut terbuka dan saat nilai rasio mulut kurang dari 0,7 akan terdeteksi mulut tertutup. Saat mulut terbuka lebih dari 1 detik, maka akan keluar sebuah output kondisi mulut menguap dan setelah itu akan dikirim ke database. Setelah diketahui total kondisi mata mengantuk dan mulut menguap, maka akan dihitung jumlah kondisi tersebut. Saat kondisi mata mengantuk lebih dari 14 kali atau mulut menguap lebih dari 4, maka mahasiswa berada pada kondisi tidak fokus. Sedangkan jika mata mengantuk kurang dari 14 kali dan mulut menguap kurang dari 4, maka mahasiswa berada pada kondisi fokus. Selanjutnya data kondisi tersebut akan dikirim ke database dan sistem akan menampilkan grafik nilai rasio mata dan mulut. Flowchart dari sistem dapat dilihat pada Gambar 1.

Kebutuhan Perangkat Keras dan Perangkat Lunak

Perangkat keras yang dibutuhkan pada Proyek Akhir ini adalah sebuah webcam Fantech Luminous C30 yang memiliki kamera 4MP dan frame rate 25 fps. Sedangkan perangkat lunak yang digunakan pada Proyek Akhir ini adalah Visual Studio Code, Python *language*, dan *Firebase*.

Hasil dan Pembahasan

Pengujian Intensitas Cahaya

Pengujian intensitas cahaya dilakukan dengan menggunakan alat luxmeter untuk mengukur besar intensitas cahaya.

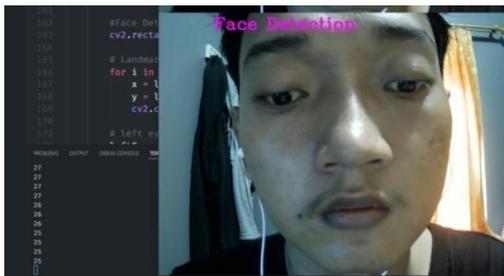
Dari pengujian pada Tabel 1, didapatkan hasil bahwa jika nilai cahaya yang diukur menggunakan luxmeter adalah 76 lux dan wajah membekali arah datangnya cahaya, maka sistem tidak dapat melakukan deteksi. Sedangkan pada saat nilai cahaya 118,3 lux, 382 lux, dan 671 lux, sistem dapat melakukan deteksi dengan baik.

Pengujian Jarak

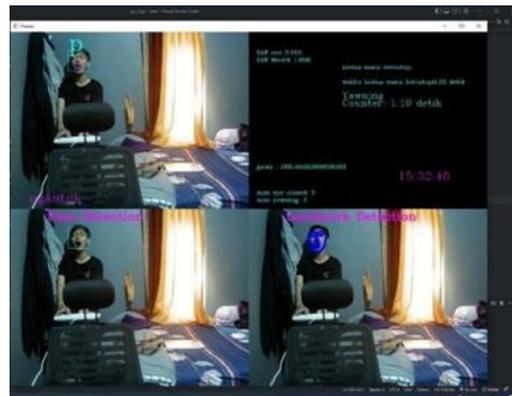
Pengujian dilakukan dengan melakukan pendeteksian wajah dengan berbagai jarak yang beragam. Pada pengujian dengan jarak 25 cm

Table 1. Pengujian intensitas cahaya

| Nilai intensitas cahaya (lux) | Keterangan Gambar | Gambar | Hasil Sistem |
|-------------------------------|---|---|------------------|
| 76 | Kondisi ruangan gelap dan wajah membekali arah datangnya cahaya |  | Tidak Terdeteksi |
| 118,3 | Kondisi cahaya normal |  | Terdeteksi |
| 119,6 | Kondisi cahaya normal, namun muka membekali arah datangnya cahaya |  | Tidak Terdeteksi |
| 382 | Kondisi Cahaya Normal di dalam ruangan |  | Terdeteksi |
| 671 | Kondisi cahaya normal di luar ruangan |  | Terdeteksi |



Gambar 2. Jarak 25 cm



Gambar 4. Jarak 268 cm



Gambar 3. Jarak 191 cm



Gambar 5. Jarak 446 cm - Terdeteksi

atau pada saat wajah tidak sepenuhnya terlihat, sistem tidak dapat melakukan deteksi seperti pada gambar 2.

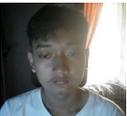
Gambar 3 adalah percobaan selanjutnya melakukan pengujian pada jarak 191 cm, sistem masih dapat mendeteksi.

Selanjutnya, percobaan dilakukan pada jarak 268 cm dan sistem masih dapat mendeteksi wajah pada gambar 4.

Gambar ?? dan gambar ?? adalah percobaan pada jarak 446 cm, sistem deteksi mulai kesulitan dalam mendeteksi wajah.

Berdasarkan pengujian di atas, didapatkan hasil beberapa yaitu, jarak minimal yang dapat dideteksi sistem adalah 25 cm atau pada saat wajah tidak sepenuhnya terlihat, dan pada jarak di atas 400 cm, wajah masih terdeteksi tetapi tidak seoptimal pada saat jarak di bawah 200 cm.

Table 2. Pengujian deteksi wajah

| Gambar | Kondisi | Status Wajah |
|--|---|------------------|
|  | Wajah menghadap ke depan | Terdeteksi |
|  | Menggunakan kacamata | Terdeteksi |
|  | Wajah tertutup tetapi mata terlihat | Terdeteksi |
|  | Wajah tertutup hanya satu mata terlihat | Tidak Terdeteksi |
|  | Wajah menunduk | Tidak Terdeteksi |
|  | Sedang menghadap kesamping | Tidak Terdeteksi |
|  | Sedang membelakangi arah datang cahaya | Tidak Terdeteksi |



Gambar 6. Jarak 446 cm - Tidak Terdeteksi

Pengujian Deteksi Wajah

Pengujian akan dilakukan dengan mendeteksi wajah menggunakan metode haar cascade classifier terhadap berbagai variabel.

Berdasarkan pengujian pada Tabel 2, diperoleh hasil bahwa dengan menggunakan metode haar cascade classifier tidak mengalami kendala pada saat kondisi normal. Akan tetapi, pada saat kondisi kepala menghadap samping maupun kondisi ruangan sedikit gelap, metode haar cascade classifier mengalami kendala pada saat melakukan pendeteksian wajah.

Pengujian landmark mata dan mulut

Pengujian akan dilakukan dengan membuat titik lingkaran pada semua landmark wajah. Berdasarkan pengujian table 3 dapat disimpulkan



Gambar 7. Rasio mata 0.311



Gambar 8. Rasio mata 0.136

jika landmark dapat terdeteksi bahkan saat sedang menggunakan kacamata.

Pengujian Eye Aspect Ratio (EAR) dan Mouth Aspect Ratio (MAR)

Pengujian lebar mata dilakukan dengan menentukan Eye Aspect Ratio (EAR) yang dapat ditemukan dengan mencari jumlah 2 jarak vertikal mata dibagi 2 kali jumlah jarak horizontal mata seperti pada rumus berikut.

$$EAR = \frac{\|P2 - P6\| + \|P3 - P5\|}{2\|P1 - P4\|} \tag{1}$$

Pengujian pertama dilakukan pada kondisi mata terbuka lebar dan mendapatkan nilai rasio mata sebesar 0.311, seperti terlihat pada gambar 7.

Pengujian selanjutnya dilakukan dengan kondisi mata yang sedang tertutup dan mendapatkan nilai rasio sebesar 0.136, 0.150, 0.161, 0.193.

Table 3. Pengujian landmark

| Gambar | Kondisi | Status Wajah |
|---|--|---------------------|
|  | Wajah menghadap ke depan | Landmark terdeteksi |
|  | Sedang membelakangi arah datang cahaya | Landmark terdeteksi |



Gambar 9. Rasio mata 0.150



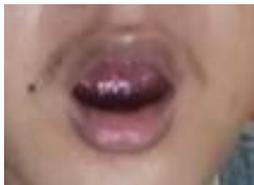
Gambar 10. Rasio mata 0.161



Gambar 11. Rasio mata 0.193



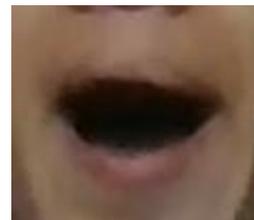
Gambar 12. Rasio mulut 0.066



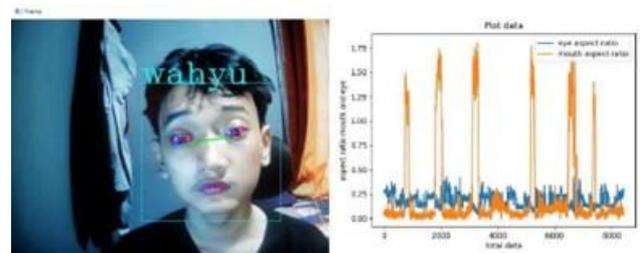
Gambar 13. Rasio mulut 0.735



Gambar 14. Rasio mulut 0.743



Gambar 15. Rasio mulut 0.787



Gambar 16. Pengujian Fokus

Pengujian selanjutnya adalah pengujian lebar mulut dengan menentukan *Mouth Aspect Ratio* (MAR) yang dapat ditemukan dengan mencari jumlah 3 jarak vertikal mulut dibagi 2 kali jumlah jarak *horizontal* mulut seperti pada rumus berikut.

$$MAR = \frac{\|P2 - P8\| + \|P3 - P7\| + \|P4 - P6\|}{2\|P1 - P5\|} \quad (2)$$

Berikutnya pada gambar 12 melakukan pengujian pada kondisi mulut yang sedang tertutup dan mendapatkan nilai rasio sebesar 0.066.

Terakhir pada gambar 13, gambar 14, gambar 15 dilakukan pengujian pada saat kondisi mulut yang sedang menguap dan mendapatkan nilai rasio sebesar 0.735, 0.743, 0.787. Dari pengujian tersebut, diperoleh hasil bahwa mata tertutup saat nilai rasio kedua mata.

Pengujian Fokus

Pengujian akan dilakukan dengan menghitung variabel total kedua mata tertutup dan total waktu mulut terbuka dalam jangka waktu 5 menit

terhadap 10 orang. Pengujian dilakukan terhadap beberapa orang yang sedang berada dalam kondisi tidak fokus dan kondisi fokus.

Setelah pengujian dilakukan akan didapatkan tabel grafik garis dengan data nilai rasio dan jumlah keseluruhan data. Data bagian kiri adalah tingkat nilai rasio sedangkan data bagian bawah adalah total keseluruhan data. Data yang berwarna oranye merupakan data nilai rasio gerakan mulut dan data yang berwarna biru merupakan data nilai rasio gerakan mata.

Setelah dilakukan pengujian, pada tabel 5 didapatkan hasil bahwa saat 5 orang berada dalam kondisi tidak fokus nilai rata-rata mata tertutup adalah 15,2 dan nilai rata-rata mulut terbuka adalah 3,8. Sedangkan saat 5 orang dengan kondisi fokus nilai rata-rata mata tertutup adalah 0,6 dan nilai rata-rata mulut terbuka adalah 0,6. Setelah didapatkan nilai rata-rata di atas, didapatkan hasil bahwa saat kondisi mahasiswa tidak

Table 4. Data Gerakan mata dan mulut

| Nama | Mata Tertutup > 3detik (kali) | Mulut Terbuka > 1 detik (kali) |
|-----------|-------------------------------|--------------------------------|
| Wahyu | 14 | 6 |
| Panggah | 19 | 6 |
| Ivan | 16 | 1 |
| Daffahanz | 23 | 0 |
| Jufer | 4 | 6 |
| Budi | 1 | 3 |
| Wafiq | 0 | 0 |
| Olan | 0 | 0 |
| Umar | 2 | 0 |
| Daffa | 0 | 0 |

Table 5. Data kondisi mahasiswa

| Nama | Kondisi |
|-----------|-------------|
| Wahyu | Tidak Fokus |
| Panggah | Tidak Fokus |
| Ivan | Tidak Fokus |
| Daffahanz | Tidak Fokus |
| Jufer | Tidak Fokus |
| Budi | Fokus |
| Wafiq | Fokus |
| Olan | Fokus |
| Umar | Fokus |

fokus mahasiswa sering menutup mata lebih dari 14 kali atau menguap lebih dari 4 kali.

Setelah dilakukan pengujian sebanyak 2 kali terhadap 10 orang yang berbeda, seperti pada tabel 6 dan tabel 7 didapatkan sebuah grafik garis yang menunjukkan total data yang diperoleh, gerakan mata, dan gerakan mulut yang dihitung menggunakan nilai rasio. Menampilkan data menggunakan grafik garis untuk menunjukkan peningkatan data nilai rasio mata dan mulut yang selalu bertambah[11]. Pada percobaan pertama dan kedua didapatkan 8000-10000 data dalam jangka waktu 5 menit. Akan tetapi pada saat wajah mahasiswa tidak menghadap tegak terhadap kamera, data pengujian akan berkurang. Total data gerakan mata mengantuk dan mulut menguap akan disimpan di database. Dari data dan pengujian tersebut didapatkan sebuah tingkat akurasi hasil yaitu 8 orang dinyatakan benar dan 2 orang dinyatakan tidak berada dalam kondisi yang benar. 2 orang yang tidak berada dalam kondisi yang benar adalah budi dan umar. Karena pada saat rekaman diambil kondisi sebenarnya budi dan umar adalah tidak fokus. Budi tidak sering menatap kamera sehingga pendeteksian tidak berjalan lancar, sedangkan umar melakukan rekaman saat kondisi badan tidak sehat dan memaksakan diri untuk fokus. Jadi pada pengujian fokus didapatkan hasil 80% benar memiliki kondisi fokus maupun tidak fokus dari 10 orang yang diuji dalam waktu 5 menit.

Kesimpulan

Dalam melakukan pendeteksian wajah menggunakan metode haar cascade classifier membutuhkan suatu kondisi yaitu wajah tidak terhalang sesuatu benda dan kondisi ruangan memiliki pencahayaan yang cukup. Dilib python juga dapat melakukan pendeteksian saat mahasiswa menggunakan kacamata maupun tidak.

Mata yang sedang tertutup memiliki nilai rasio dibawah 0,2 sedangkan mulut yang sedang terbuka memiliki nilai rasio diatas 0,7. Saat kondisi sedang tidak fokus nilai rata-rata mata tertutup selama 3 detik yaitu 15,2 kali dan nilai rata-rata mulut terbuka selama 1 detik yaitu 3,8 kali. Sedangkan pada saat kondisi sedang fokus nilai rata-rata mata tertutup yaitu 0,6 kali dan nilai rata-rata mulut terbuka yaitu 0,6 kali.

Jarak minimal agar wajah terdeteksi sistem adalah 25 cm sedangkan pada jarak diatas 400 cm sistem sudah tidak optimal dalam mendeteksi wajah. Sistem memiliki akurasi deteksi tingkat fokus sebesar 80% benar dari 10 orang yang telah diuji selama 5 menit dengan kondisi fokus dan tidak fokus.

Daftar Pustaka

1. Siahaan M. Dampak Pandemi Covid-19 Terhadap Dunia Pendidikan; 2020. Available from: <http://ejurnal.ubharajaya.ac.id/index.php/JKI>.
2. Massar SAA, Lim J, Sasmita K, Chee MWL. Sleep deprivation increases the costs of attentional effort: Performance, preference and pupil size. *Neuropsychologia*. 2019 Feb;123:169–177.
3. Saputra CA, Erwanto D, Rahayu PN. DETEKSI KANTUK PENGEN-DARA RODA EMPAT MENGGUNAKAN HAAR CASCADE CLASSIFIER dan CONVOLUTIONAL NEURAL NETWORK. *JEECOM*. 2021;3(1).
4. Pawicara R, Conilie M. ANALISIS PEMBELAJARAN DARING TERHADAP KEJENUHAN BELAJAR MAHASISWA TADRIS BIOLOGI IAIN JEMBER DI TENGAH PANDEMI COVID-19. 2020.
5. Abidin S. Deteksi Wajah Menggunakan Metode Haar Cascade Classifier Berbasis Webcam Pada Matlab.
6. Zein A. Pendeteksian Kantuk Secara Real Time Menggunakan Pustaka OPENCV dan DLIB PYTHON Real Time Sleepiness Detection Using OPENCV Library and PYTHON DLIB. 2018.
7. Maulana MA. Deteksi Kantuk Pada Pengendara Roda Empat Melalui Citra Wajah Menggunakan Facial Landmark. 2022.
8. David F. VISUALISASI DATA DALAM BENTUK 3 DIMENSI DENGAN MENGGUNAKAN BAHASA PEMROGRAMAN PYTHON. 2019.
9. Herho SHS. SERI KOMPUTASI TUTORIAL VISUALISASI DATA MENGGUNAKAN SEABORN; 2019. Available from: <https://sandyherho.github.io/>.
10. Fitri NA, Emba RZ, Mufid MR, Fiyanto A, Wajib W, Shofyan A. Kediri City Tourism Object Application Using Firebase Realtime Database Technology. 2022.
11. Valcheva S. Types of Graphs and Their Uses;. Available from: <http://intellspot.com/>.
12. Perdana AHAP. IMPLEMENTASI SISTEM DETEKSI MATA KANTUK BERDASARKAN FACIAL LANDMARKS DETECTION MENGGUNAKAN METODE REGRESSION TRESS. 2019.
13. Soukupova T, Cech JA. Real-Time Eye Blink Detection using Facial Landmarks. 2016.

Table 6. Pengujian Pertama

| Nama | Gambar | Grafik Pengujian Pertama |
|-----------|--------|---|
| Wahyu | | <p>Mata tertutup selama 3 detik sebanyak 14 kali Mulut terbuka selama 1 detik sebanyak 6 kali</p> |
| Panggih | | <p>Mata tertutup selama 3 detik sebanyak 19 kali Mulut terbuka selama 1 detik sebanyak 6 kali</p> |
| Ivan | | <p>Mata tertutup selama 3 detik sebanyak 16 kali Mulut terbuka selama 1 detik sebanyak 1 kali</p> |
| Daffahanz | | <p>Mata tertutup selama 3 detik sebanyak 14 kali Mulut terbuka selama 1 detik sebanyak 6 kali</p> |
| Jufer | | <p>Mata tertutup selama 3 detik sebanyak 4 kali Mulut terbuka selama 1 detik sebanyak 6 kali</p> |
| Budi | | <p>Mata tertutup selama 3 detik sebanyak 1 kali Mulut terbuka selama 1 detik sebanyak 3 kali</p> |
| Wafiq | | <p>Mata tertutup selama 3 detik sebanyak 0 kali Mulut terbuka selama 1 detik sebanyak 0 kali</p> |
| Olan | | <p>Mata tertutup selama 3 detik sebanyak 0 kali Mulut terbuka selama 1 detik sebanyak 0 kali</p> |
| Umar | | <p>Mata tertutup selama 3 detik sebanyak 0 kali Mulut terbuka selama 1 detik sebanyak 0 kali</p> |

Table 7. Pengujian Kedua

| Nama | Gambar | Grafik Pengujian Pertama |
|-----------|--------|---|
| Wahyu | | <p>Mata tertutup selama 3 detik sebanyak 14 kali Mulut terbuka selama 1 detik sebanyak 7 kali</p> |
| Panggha | | <p>Mata tertutup selama 3 detik sebanyak 20 kali Mulut terbuka selama 1 detik sebanyak 5 kali</p> |
| Ivan | | <p>Mata tertutup selama 3 detik sebanyak 15 kali Mulut terbuka selama 1 detik sebanyak 2 kali</p> |
| Daffahanz | | <p>Mata tertutup selama 3 detik sebanyak 14 kali Mulut terbuka selama 1 detik sebanyak 6 kali</p> |
| Jufer | | <p>Mata tertutup selama 3 detik sebanyak 3 kali Mulut terbuka selama 1 detik sebanyak 7 kali</p> |
| Budi | | <p>Mata tertutup selama 3 detik sebanyak 1 kali Mulut terbuka selama 1 detik sebanyak 3 kali</p> |
| Wafiq | | <p>Mata tertutup selama 3 detik sebanyak 0 kali Mulut terbuka selama 1 detik sebanyak 0 kali</p> |
| Olan | | <p>Mata tertutup selama 3 detik sebanyak 0 kali Mulut terbuka selama 1 detik sebanyak 0 kali</p> |
| Umar | | <p>Mata tertutup selama 3 detik sebanyak 0 kali Mulut terbuka selama 1 detik sebanyak 0 kali</p> |