

# Deteksi Kedipan dengan Metode CNN dan *Percentage of Eyelid Closure (PERCLOS)*

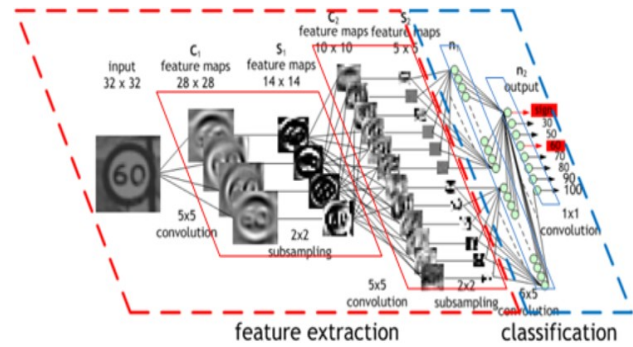
Lutfi Ananditya Septiandi, Eko Mulyanto Yuniarno dan Ahmad Zaini  
Departemen Teknik Komputer, Institut Teknologi Sepuluh Nopember (ITS)  
e-mail: ekomulyanto@ee.its.ac.id; zaini@ee.its.ac.id.

**Abstrak**— Pengembangan teknologi mengenai *face detection* dan *eyes detection* melaju sangat pesat, sehingga peneliti berlomba-lomba meneliti metode dan algoritma yang optimal untuk pengaplikasian di kehidupan sehari-hari, mulai dari pengamanan *biometrics* sampai identifikasi wajah secara otomatis. Di tugas akhir ini diusulkan penggunaan metode *Convolutional Neural Network (CNN)* dan *Percentage of Eyelid Closure (PERCLOS)* pada pendeteksi kedipan mata. Sistem dibangun menggunakan *webcam personal computer* sebagai kamera dan mendeteksi secara *real-time*. Sistem dapat mengenali kondisi ketika mata tertutup atau mata terbuka dan menentukan lebar bukaan mata dengan menggunakan *Eye Aspect Ratio (EAR)* serta dapat mengestimasi skor tatapan dengan menggunakan *Percentage of Eyelid Closure (PERCLOS)*. Sistem dapat mengenali wajah dari objek bukan wajah dengan jarak pendeteksian optimal antara 40-70 cm. Model hasil *training* dapat mengklasifikasikan kondisi mata terbuka dan mata tertutup dengan menggunakan *Convolutional Neural Network* dengan arsitektur yang memiliki 3 layer mendapatkan hasil *accuracy* 98% dan *loss* 2.05%.

**Kata Kunci**—*Face detection*, *Eyes detection*, *Convolutional Neural Network (CNN)*, *Eye Aspect Ratio (EAR)*, *Percentage of Eyelid Closure (PERCLOS)*.

## I. PENDAHULUAN

PERKEMBANGAN teknologi pada era masa kini sangatlah pesat apalagi dibidang komputer, hal ini ditandai dengan hampir semua pengelola data dan informasi telah dilakukan dengan komputer. Banyak pekerjaan manusia yang tergantikan dengan komputer dikarenakan sistem yang ada di komputer lebih optimal dan lebih efisien serta menghemat waktu oleh karena itu peran komputer didunia ini sangatlah penting sekarang. Deteksi wajah merupakan salah satu teknologi yang sekarang ini banyak dikembangkan seiring berkembangnya teknologi komputer. Dengan berkembangnya teknologi deteksi wajah, penelitian mengenai *eyes detection* juga ikut berkembang pesat. Beberapa teknologi yang menggunakan deteksi mata antara lain pendeteksian kedipan mata. Ada banyak metode yang digunakan untuk mendeteksi kedipan mata, salah satunya adalah menggunakan *Percentage of Eyelid Closure (PERCLOS)*. Metode *Percentage of Eyelid Closure (PERCLOS)* digunakan diberbagai penelitian untuk menangkap kondisi mata pengemudi menggunakan sistem visi komputer. Selain itu, metode ini juga digunakan sebagai pengukuran durasi penutupan mata dan tingkat kedip [1]. Dengan adanya penelitian ini, diharapkan sistem dapat mendeteksi kondisi mata dan juga dapat mengestimasi skor tatapan pengemudi.



Gambar 1. Ilustrasi Arsitektur *Convolutional Neural Network (CNN)* [1].

## II. DASAR TEORI

### A. CNN

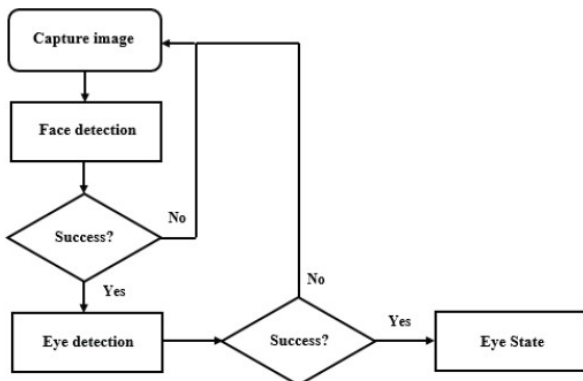
*Convolutional Neural Network* merupakan *trainable multistage architectures* yang terdiri dari beberapa tahapan *input* dan *output*. *Convolutional Neural Network* termasuk dalam jenis *Deep Neural Network* karena dalamnya tingkat jaringan dan banyak diimplementasikan dalam data citra. Cara kerja *Convolutional Neural Network* memiliki kesamaan pada *Multi Layer Preceptron*, namun dalam *Convolutional Neural Network* setiap neuron dipresentasikan dalam bentuk dua dimensi, tidak seperti *Multi Layer Preceptron* yang setiap neuron hanya berukuran satu dimensi. Ada 3 lapisan didalam arsitektur CNN, yaitu *convolutional layer*, *fully connected layer*, dan *subsampling layer*. *Convolutional layer* dilengkapi dengan activation *rectified linear unit* atau aktivasi *ReLU*. *Subsampling layer* menggunakan fungsi *max pooling* dimana fungsi ini digunakan untuk mengurangi jumlah parameter dan perhitungan jaringan, dan untuk mengendalikan *overfitting*. *Fully connected layer* menggunakan fungsi aktivasi *softmax*. Gambar 1 merupakan ilustrasi dari arsitektur CNN.

### B. Kinerja Klasifikasi

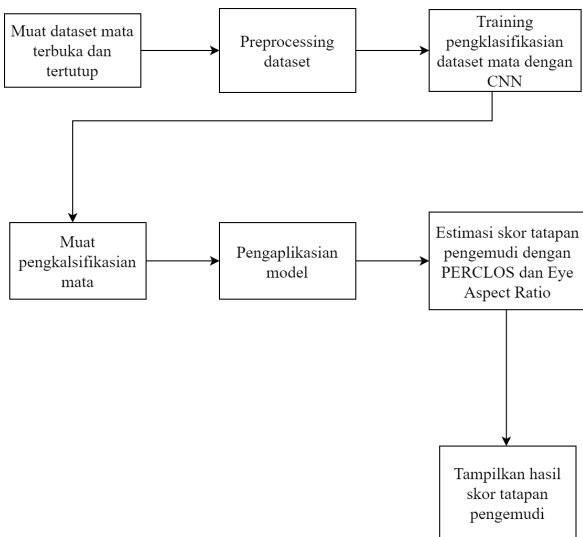
Dalam pengukuran kinerja suatu sistem klasifikasi menggambarkan kinerja sistem tersebut untuk mengklasifikasi data. Salah satu metode yang dapat digunakan untuk mengukur kinerja suatu sistem klasifikasi adalah *confusion matrix*. *Confusion matrix* berisi perbandingan hasil klasifikasi dari sistem dengan hasil yang seharusnya diklasifikasikan. *Confusion matrix* merupakan tabel dua dimensi yang terdiri dari data aktual dan data prediksi serta memiliki kelas masing-masing, dimana data aktual terletak pada bagian kolom sedangkan data prediksi terletak pada bagian baris.

Table 1. Confusion Matrix.

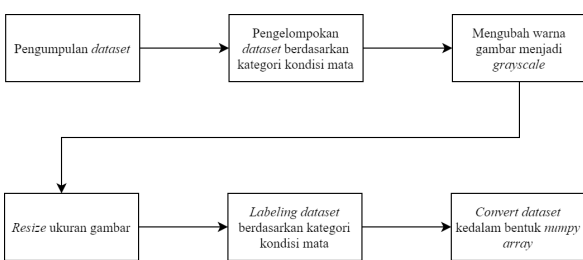
Positive	True Positive (TP)	False Negative (FN)
Negative	False Positive (FP)	True Negative (TN)
	Positive	Negative



Gambar 2. Ilustrasi algoritma Haar Cascade Classifier dalam sistem pendeteksian mata [2].



Gambar 3. Alur Kerja Sistem.



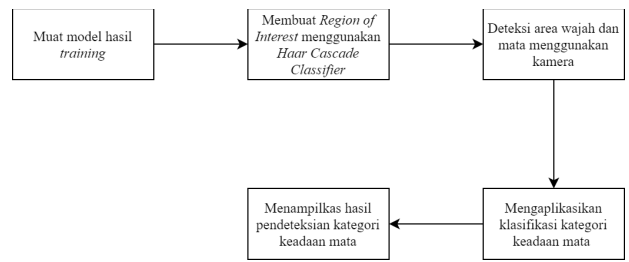
Gambar 4. Alur Preprocessing

Tabel 1 merupakan tabel *confusion matrix*. Hasil dari tabel *confusion matrix* dapat digunakan untuk berbagai pengukuran performa yang lebih detail.

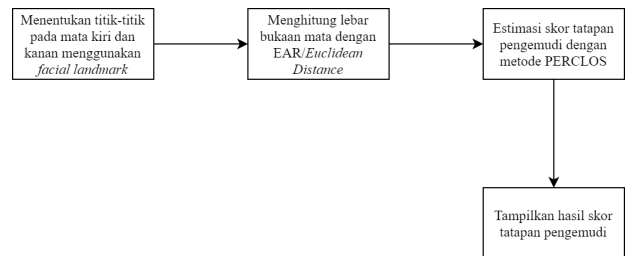
1) *Accuracy*: *Accuracy* digunakan untuk mencari keakuratan nilai dari sebuah model yang dapat mengklasifikasi data yang benar. Oleh karena itu, *accuracy* adalah jumlah klasifikasi data yang benar dari jumlah data keseluruhan. Nilai *accuracy* dapat diperoleh dari persamaan 1.

$$Accuracy = \frac{TP+TN}{TP+FN+FP+TN} \times 100\% \quad (1)$$

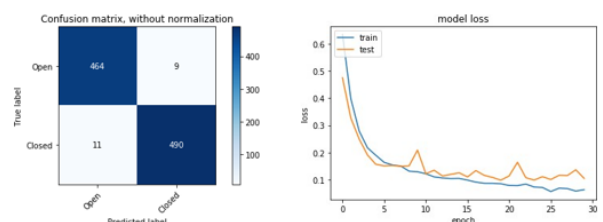
2) *Error*: *Error* merupakan jumlah klasifikasi data yang salah dari jumlah data keseluruhan. Ada dua macam *errors* dalam *confusion matrix* yaitu: Tipe I dan Tipe II.



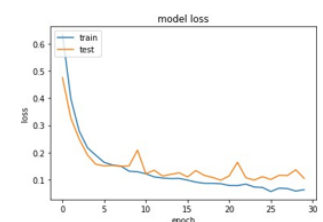
Gambar 5. Alur Pengaplikasian Model.



Gambar 6. Alur Estimasi Skor Tatapan.



(a) Confusion matrix



(b) Loss chart

	precision	recall	f1-score	support
0	0.98	0.98	0.98	473
1	0.98	0.98	0.98	501
accuracy			0.98	974
macro avg	0.98	0.98	0.98	974
weighted avg	0.98	0.98	0.98	974

(c) Classification report

Gambar 7. Confusion Matrix, Loss Chart, Dan Classification Report Dari Hasil Pengujian Kedipan Mata.

*False Positive* berisi satu kata *negative (False)* sehingga *False Positive* merupakan *error* tipe I. *False Negative* memiliki dua kata *negative (False + Negative)* sehingga *False Negative* merupakan *error* tipe II. Nilai *error* dapat diperoleh dari persamaan 2.

$$Error = \frac{FP+FN}{TP+FN+FP+TN} \times 100\% \quad (2)$$

3) *Precision*: *Precision* digunakan untuk mencari tingkat keakuratan antara data yang diminta dan hasil prediksi. Oleh karena itu, *precision* merupakan rasio prediksi *True Positive* terhadap prediksi *positive* secara keseluruhan. Nilai *precision* dapat diperoleh dari persamaan 3.

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

4) *Recall*: *Recall* digunakan untuk mencari keberhasilan model dalam memulihkan kembali informasi. Oleh karena itu, *recall* merupakan rasio antara prediksi *true positive* dengan data *true positive* secara keseluruhan. Nilai *recall* dapat diperoleh dari persamaan 4.

$$Recall = \frac{TP}{TP+FN} \times 100\% \quad (4)$$

5) *F1 Score*: *F1 Score* merupakan nilai antara 0 dan 1 yang dimana diperoleh dari rata-rata harmonik antara *precision*

Table 2.  
Pengujian Video

Video	Terdeteksi	Tidak terdeteksi
1-Jan	19,661	4,026
2-Jan	3,612	5,887
3-Jan	1,732	7,141
1-Feb	2,058	15,885
2-Feb	46	9,356
3-Feb	942	7,224
1-Mar	5,685	12,533
2-Mar	1,011	7,848
3-Mar	518	8,360
1-Apr	12,538	5,303
2-Apr	4,509	4,284
3-Apr	8,058	3,218
1-May	12,112	6,155
2-May	8,318	1,484
3-May	12,835	511
1-Jun	4,406	13,507
2-Jun	298	8,556
3-Jun	41	8,917
2-Jul	1,260	6,688
3-Jul	155	8,724
1-Aug	9,326	8,794
2-Aug	2,249	6,671
3-Aug	1,803	7,321
2-Sep	7,275	10,952
3-Sep	5,247	12,912
1-Oct	70	17,797
3-Oct	1,072	16,827
1-Nov	6,506	11,652
2-Nov	1,009	16,877
3-Nov	349	17,528
1-Dec	9,987	7,913
13-1	2,078	15,856
13-2	6,664	11,968
14-1	2,614	15,271
14-2	1,149	16,713
14-3	1,102	16,868

dan recall. Nilai *F1 Score* dapat diperoleh dari persamaan 5.

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (5)$$

### C. Face Detection

Wajah merupakan salah satu bagian dari manusia yang memiliki ciri berbeda dari setiap manusia. Bagian dari wajah manusia meliputi mata, hidung, pipi, mulut, dahi, alis, dagu dan juga rambut [2]. *Face detection* merupakan sebuah teknologi untuk mendeteksi wajah yang dibangun menggunakan algoritma tertentu dan berfokus untuk mendeteksi wajah manusia dalam bentuk *digital images*. Algoritma pendeteksian wajah biasanya dimulai dari mata manusia, dikarenakan mata merupakan salah satu fitur termudah untuk dideteksi. Kemudian pencarian lainnya bisa saja mendeteksi alis, mulut, hidung, lubang hidung, dan iris.

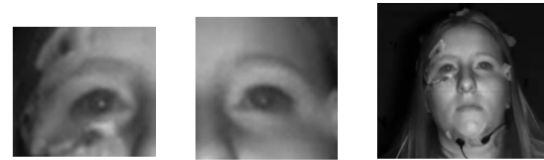
### D. Pendeteksian Mata

Salah satu pengembangan dari *face detection* adalah *eyes detection*. *Eyes detection* merupakan pengembangan lanjut dari deteksi wajah, dimana citra wajah manusia yang berhasil dideteksi akan diproses kembali dengan mendeteksi kedua mata pada wajah [3]. Salah satu metode pendeteksian mata

adalah dengan menggunakan *Haar Cascade Classifier*. *Haar*



(a) Kondisi mata kanan. (b) Kondisi mata kiri. (c) Wajah.  
Gambar 9. Contoh *Data Testing* 1-1 - 1-3.



(a) Kondisi mata kanan. (b) Kondisi mata kiri. (c) Wajah.  
Gambar 10. Contoh *Data Testing* 2-1 - 2-3.



(a) Kondisi mata kanan. (b) Kondisi mata kiri. (c) Wajah.  
Gambar 11. Contoh *Data Testing* 3-1 - 3-3.

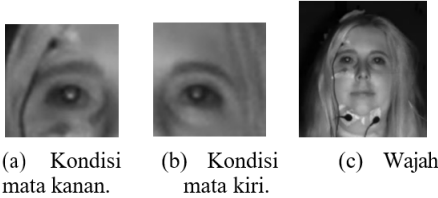
*Cascade Classifier* merupakan salah satu *library* dari *OpenCV* yang dapat mengenali posisi dari objek yang ingin dideteksi dengan cara memberikan *Region of Interest (RoI)* kepada objek tersebut. *Cascade Classifier* merupakan langkah untuk mendapatkan hasil yang lebih akurat dengan menghitung nilai *Haar Feature* secara banyak dan berulang. Gambar 2 merupakan ilustrasi dari algoritma *Haar Cascade Classifier* dalam mendeteksi mata. Pada tahap pertama, penangkapan dan pendeteksian citra wajah. Pada tahap kedua, ketika wajah dapat di deteksi proses selanjutnya adalah pendeteksian citra mata, bila tidak terdeteksi maka proses selanjutnya menangkap kembali citra wajah. Pada tahap ketiga, ketika citra mata dapat dideteksi proses selanjutnya menentukan kondisi mata, bila tidak terdeteksi maka proses selanjutnya kembali ke tahap pertama [4].

### E. Related Works

- 1) *Related Works* tentang CNN: Dalam penelitian "Convolutional Networks and Applications in Vision" membahas tentang *vision science* dan *convolutional network*, dimana permasalahan mengenai *vision science* dapat diselesaikan menggunakan *convolutional network* dan *gradient-based supervised learning* [5].
- 2) *Related Works* tentang *face detection*: Dalam penelitian "Deteksi Wajah Menggunakan Metode Haar Cascade Classifier Berbasis Webcam Pada Matlab" membahas tentang penggunaan algoritma *Haar Cascade Classifier* dalam mendeteksi wajah secara real-time [2].
- 3) *Related Works* tentang *eyes detection*: Dalam penelitian "Real-Time Eye State Detection System Using Haar



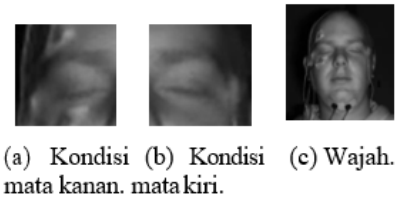
Gambar 12. Contoh *Data Testing* 4-1 - 4-3.



Gambar 13. Contoh *Data Testing* 5-1 - 5-3.



Gambar 14. Contoh *Data Testing* 6-1 - 6-3.



Gambar 15. Contoh *Data Testing* 7-2 dan 7-3.

*Cascade Classifier and Circular Hough Transform*” membahas tentang penggunaan algoritma *Haar Cascade Classifier* dan *Circular Hough Transform* dalam mendeteksi kondisi mata. Penelitian ini dapat mendeteksi keadaan mata terbuka dan tertutup dan mendapatkan akurasi sebesar 98.56% [4].

### III. DESAIN SISTEM

Penelitian ini bertujuan untuk mendeteksi kedipan dari pengemudi dan dapat mengestimasi skor tatapan dari pengemudi beserta kondisi mata pengemudi. Kemudian dari kumpulan citra tersebut di-*training* menggunakan CNN dan mendapatkan pengklasifikasian mata tertutup dan terbuka. Terdapat 4 bab yang membangun tujuan dari penelitian ini, yaitu pengumpulan *dataset* mengenai pengelompokan kondisi mata, kemudian melakukan *training dataset* dengan menggunakan CNN, selanjutnya mengaplikasikan model *training* secara *real-time* menggunakan kamera agar mendapatkan kondisi mata, yang terakhir adalah mengestimasi skor tatapan pengemudi dengan menggunakan metode *PERcentage of eyelid CLOSure* dan menentukan lebar bukaan mata dengan *Euclidean Distance*. Alur kerja sistem dijelaskan pada Gambar 3.

#### A. Preprocessing Dataset

Pada tahapan ini, dilakukan pengumpulan dan pemrosesan awal agar *dataset* siap digunakan untuk proses selanjutnya. *Dataset* yang digunakan berupa potongan bagian mata kanan dan kiri secara terpisah yang berasal dari dokumentasi penelitian sebelumnya [6] dan dokumentasi pribadi. Kemudian *dataset* diubah dalam format *grayscale* dan dikelompokkan berdasarkan keadaan mata, yaitu: keadaan mata terbuka (*open*) dan keadaan mata tertutup (*close*). Setelah gambar mata di ubah dalam bentuk *grayscale*, selanjutnya gambar di ubah atau di-*convert* dalam bentuk *numpy array*. Gambar 4 adalah alur kerja dari *preprocessing*.

#### B. Training Dataset

*Training dataset* bertujuan untuk melatih komputer agar mencapai sebuah keputusan atau prediksi dengan cara mengolah citra sehingga terbentuk pola atau karakteristik. Total *dataset* yang di-*training* sebanyak 4,868 citra dengan rincian 2,476 citra mata terbuka dan 2,392 citra mata tertutup.

*Dataset* yang dikumpulkan akan diproses dengan melakukan *training* yang menggunakan *Convolutional Neural Network* dengan 3 *layer* dan ada sebanyak 30 *epochs*. Ketika *epochs* sudah mencapai 30, maka secara otomatis proses *training* akan berhenti. *Output* dari *training* berupa model yang nantinya akan digunakan untuk proses pendeteksian.

#### C. Pengaplikasian Model

Pada proses ini dilakukan pendeteksian pada wajah dan mata dengan menggunakan *Haar Cascade Classifier* agar mendapatkan *Region of Interest (RoI)* pada bagian wajah dan mata. Ketika wajah dan mata sudah dapat dideteksi, maka langkah selanjutnya adalah model yang sudah di-*training* akan di muat untuk proses pendeteksian. Selanjutnya melakukan *labeling* pada pendeteksi mata dan menentukan kondisi mata terbuka (*open*) atau mata tertutup (*close*). Pendeteksian secara *real-time* dijelaskan pada gambar 5.

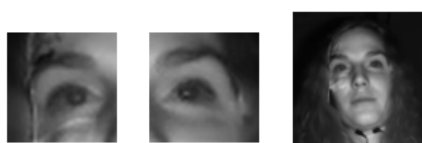
#### D. Estimasi Skor Tatapan dan Eye Aspect Ratio (EAR)

Ketika kamera sudah dapat mengenali kondisi mata terbuka atau tertutup, selanjutnya sistem akan menganalisa lebar bukaan mata. Cara menentukan lebar bukaan mata dengan menggunakan *Eye Aspect Ratio (EAR)*. Untuk dapat menggunakan *Eye Aspect Ratio*, yang dibutuhkan adalah menentukan titik-titik pada mata. Untuk menentukan titik-titik pada mata, bisa menggunakan *facial recognition landmark*. Setelah mendapatkan titik-titik dari mata dan nilai *Eye Aspect Ratio*, proses selanjutnya adalah mengestimasi skor pengemudi dengan menggunakan metode *Percentage of Eyelid Closure (PERCLOS)*. Gambar 6 adalah alur kerja dari penggunaan *Eye Aspect Ratio (EAR)* dan *PERcentage of eyelid CLOSure (PERCLOS)*.

- 1) *Menghitung Lebar Bukaan Mata*: Untuk menghitung lebar bukaan mata menggunakan *Eye Aspect Ratio (EAR)*, dimana *Eye Aspect Ratio (EAR)* merupakan proporsi antara lebar dan ketinggian bukaan kelopak mata atas dan kelopak mata bawah. *Eye Aspect Ratio (EAR)* dirumuskan pada persamaan 6 [7].

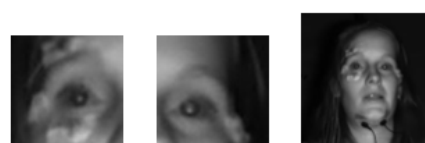
$$EAR = \frac{\|P_2 - P_6\| + \|P_3 - P_5\|}{2\|P_1 - P_4\|} \tag{6}$$

- 2) *Estimasi Skor Tatapan*: Untuk mengestimasi skor tatapan menggunakan *PERcentage of eyelid CLOSure (PERCLOS)*. *PERcentage of eyelid CLOSure (PERCLOS)* digunakan untuk menunjukkan durasi penutupan mata



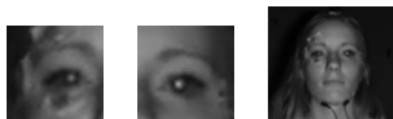
(a) Kondisi mata kanan. (b) Kondisi mata kiri. (c) Wajah.

Gambar 16. Contoh Data Testing 8-1 - 8-3.



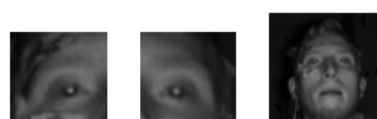
(a) Kondisi mata kanan. (b) Kondisi mata kiri. (c) Wajah.

Gambar 18. Contoh Data Testing 10-1 dan 10-3.



(a) Kondisi mata kanan. (b) Kondisi mata kiri. (c) Wajah.

Gambar 17. Contoh Data Testing 9-2 dan 9-3.



(a) Kondisi mata kanan. (b) Kondisi mata kiri. (c) Wajah.

Gambar 19. Contoh Data Testing 11-1 - 11-3.

dan tingkat kedipan mata. *PERcentage of eyelid CLOSure (PERCLOS)* dirumuskan pada persamaan 7 [8].

$$PERCLOS = \frac{Nm - Na}{Nm} \times 100\% \quad (7)$$

#### IV. PENGUJIAN DAN ANALISA

Langkah pengujian dalam penelitian ini dilakukan seperti pada tahap berikut ini.

##### A. Pengujian Model Terhadap Klasifikasi Kedipan Mata

Pada klasifikasi kondisi mata menggunakan epochs sebesar 30 mendapatkan akurasi sebesar 98%, *Train loss* sebesar 0.6%, *Validation loss* sebesar 1.05%, dan *error* sebesar 2.05%. Gambar 7 merupakan *confusion matrix*, *loss chart*, dan *classification report* dari hasil evaluasi deteksi kedipan. Adapun lima kriteria yang menentukan berhasil atau tidaknya pendeteksian adalah sebagai berikut:

1. Nilai dari *Eye Aspect Ratio (EAR)*.
2. Estimasi skor tatapan pengemudi dengan menggunakan *Percentage of Eyelid Closure (PERCLOS)*.
3. *Region of Interest (RoI)* dari wajah.
4. *Region of Interest (RoI)* dari mata.
5. Kondisi mata ketika terbuka (*open*) atau tertutup (*close*) serta nilai *confidence* dari prediksi *training*.

Gambar 8 merupakan hasil dari pengujian model yang dilakukan secara *real-time* menggunakan CNN dan PERCLOS untuk mendapatkan lima kriteria dan menentukan berhasil atau tidaknya sistem pendeteksian.

##### B. Pengujian Terhadap Video Testing

Pengujian menggunakan video *testing* terbagi menjadi 36 video dan setiap video memiliki durasi yang berbeda-beda untuk mengetahui kondisi mata terbuka atau tertutup, nilai lebar bukaan mata, dan skor tatapan. Dalam pengujian ini menggunakan dataset yang berbeda dari pengujian sebelumnya. *Dataset* yang digunakan dalam pengujian kali ini berupa potongan citra mata kanan dan kiri dari video *testing*. Total *dataset* yang digunakan berjumlah 4,096 dengan rincian 2,048 mata terbuka (*open*) dan 2,048 mata tertutup (*close*). Hasil dari kelima kriteria disimpan kedalam bentuk *log.csv* dan menghitung berapa jumlah *frames* yang dapat terdeteksi dan tidak terdeteksi. Tabel 2 merupakan hasil dari jumlah *frames* yang dapat terdeteksi dan tidak terdeteksi.

- 1) *Video Testing 1-1*: Pengujian pertama menggunakan video *testing* 1-1. Dalam video *testing* 1-1 mempunyai 23,687 *frames*, dengan rincian 19,661 *frames* yang terdeteksi dan 4,026 *frames* yang tidak dapat terdeteksi. Adapun beberapa *frames* tambahan pada *frames* tertentu berjumlah 5,822 *frames*. Gambar 9 merupakan *data testing* 1-1.
- 2) *Video Testing 1-2*: Pengujian kedua menggunakan video *testing* 1-2. Dalam video *testing* 1-2 mempunyai 9,499 *frames*, dengan rincian 3,612 *frames* yang terdeteksi dan 5,887 *frames* yang tidak dapat terdeteksi. Adapun beberapa *frames* tambahan pada *frames* tertentu berjumlah 2 *frames*. Gambar 9 merupakan *data testing* 1-2.
- 3) *Video Testing 1-3*: Pengujian ketiga menggunakan video *testing* 1-3. Dalam video *testing* 1-3 mempunyai 8,873 *frames*, dengan rincian 1,732 *frames* yang terdeteksi dan 7,141 *frames* yang tidak dapat terdeteksi. Adapun beberapa *frames* tambahan pada *frames* tertentu berjumlah 7 *frames*. Gambar 9 merupakan *data testing* 1-3.
- 4) *Video Testing 2-1*: Pengujian keempat menggunakan video *testing* 2-1. Dalam video *testing* 2-1 mempunyai 17,943 *frames*, dengan rincian 2,058 *frames* yang terdeteksi dan 15,885 *frames* yang tidak dapat terdeteksi. Adapun beberapa *frames* tambahan pada *frames* tertentu berjumlah 44 *frames*. Gambar 10 merupakan *data testing* 2-1.
- 5) *Video Testing 2-2*: Pengujian kelima menggunakan video *testing* 2-2. Dalam video *testing* 2-2 mempunyai 9,402 *frames*, dengan rincian 46 *frames* yang terdeteksi dan 9,356 *frames* yang tidak dapat terdeteksi. Adapun beberapa *frames* tambahan pada *frames* tertentu berjumlah 1 *frames*. Gambar 10 merupakan *data testing* 2-2.
- 6) *Video Testing 2-3*: Pengujian keenam menggunakan video *testing* 2-3. Dalam video *testing* 2-3 mempunyai 8,166 *frames*, dengan rincian 942 *frames* yang terdeteksi dan 7,224 *frames* yang tidak dapat terdeteksi. Adapun beberapa *frames* tambahan pada *frames* tertentu berjumlah 20 *frames*. Gambar 10 merupakan *data testing* 2-3.
- 7) *Video Testing 3-1*: Pengujian ketujuh menggunakan video *testing* 3-1. Dalam video *testing* 3-1 mempunyai



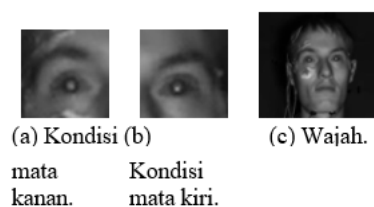
Gambar 20. Data Testing 12-1.



Gambar 21. Contoh Data Testing 13-1 dan 13-2.

18,218 *frames*, dengan rincian 5,685 *frames* yang terdeteksi dan 12,533 *frames* yang tidak dapat terdeteksi. Adapun beberapa *frames* tambahan pada *frames* tertentu berjumlah 336 *frames*. Gambar 11 merupakan *data testing* 3-1.

- 8) *Video Testing* 3-2: Pengujian kedelapan menggunakan *video testing* 3-2. Dalam *video testing* 3-2 mempunyai 8,859 *frames*, dengan rincian 1,011 *frames* yang terdeteksi dan 7,848 *frames* yang tidak dapat terdeteksi. Adapun beberapa *frames* tambahan pada *frames* tertentu berjumlah 11 *frames*. Gambar 11 merupakan *data testing* 3-2.
- 9) *Video Testing* 3-3: Pengujian kesembilan menggunakan *video testing* 3-3. Dalam *video testing* 3-3 mempunyai 8,878 *frames*, dengan rincian 518 *frames* yang terdeteksi dan 8,360 *frames* yang tidak dapat terdeteksi. Adapun beberapa *frames* tambahan pada *frames* tertentu berjumlah 4 *frames*. Gambar 11 merupakan *data testing* 3-3.
- 10) *Video Testing* 4-1: Pengujian kesepuluh menggunakan *video testing* 4-1. Dalam *video testing* 4-1 mempunyai 17,841 *frames*, dengan rincian 12,538 *frames* yang terdeteksi dan 5,303 *frames* yang tidak dapat terdeteksi. Adapun beberapa *frames* tambahan pada *frames* tertentu berjumlah 52 *frames*. Gambar 12 merupakan *data testing* 4-1.
- 11) *Video Testing* 4-2: Pengujian kesebelas menggunakan *video testing* 4-2. Dalam *video testing* 4-2 mempunyai 8,793 *frames*, dengan rincian 4,509 *frames* yang terdeteksi dan 4,284 *frames* yang tidak dapat terdeteksi. Adapun beberapa *frames* tambahan pada *frames* tertentu berjumlah 330 *frames*. Gambar 12 merupakan *data testing* 4-2.
- 12) *Video Testing* 4-3: Pengujian keduabelas menggunakan *video testing* 4-3. Dalam *video testing* 4-3 mempunyai 11,276 *frames*, dengan rincian 8,058 *frames* yang terdeteksi dan 3,218 *frames* yang tidak dapat terdeteksi. Adapun beberapa *frames* tambahan pada *frames* tertentu berjumlah 2,387 *frames*. Gambar 12 merupakan *data testing* 4-3.
- 13) *Video Testing* 5-1: Pengujian ketiga belas menggunakan *video testing* 5-1. Dalam *video testing* 5-1 mempunyai 18,267 *frames*, dengan rincian 12,112 *frames* yang terdeteksi dan 6,155 *frames* yang tidak dapat terdeteksi. Adapun beberapa *frames* tambahan pada *frames* tertentu berjumlah 369 *frames*. Gambar 13 merupakan *data testing* 5-1.
- 14) *Video Testing* 5-2: Pengujian keempat belas menggunakan *video testing* 5-2. Dalam *video testing* 5-2 mempunyai 9,802 *frames*, dengan rincian 8,318 *frames* yang terdeteksi dan 1,484 *frames* yang tidak dapat terdeteksi. Adapun beberapa *frames* tambahan pada *frames* tertentu berjumlah 1,627 *frames*. Gambar 13 merupakan *data testing* 5-2.
- 15) *Video Testing* 5-3: Pengujian kelima belas menggunakan *video testing* 5-3. Dalam *video testing* 5-3 mempunyai 13,346 *frames*, dengan rincian 12,835 *frames* yang terdeteksi dan 511 *frames* yang tidak dapat terdeteksi. Adapun beberapa *frames* tambahan pada *frames* tertentu berjumlah 5,116 *frames*. Gambar 13 merupakan *data testing* 5-3.
- 16) *Video Testing* 6-1: Pengujian keenam belas menggunakan *video testing* 6-1. Dalam *video testing* 6-1 mempunyai 17,913 *frames*, dengan rincian 4,406 *frames* yang terdeteksi dan 13,507 *frames* yang tidak dapat terdeteksi. Adapun beberapa *frames* tambahan pada *frames* tertentu berjumlah 35 *frames*. Gambar 14 merupakan *data testing* 6-1.
- 17) *Video Testing* 6-2: Pengujian ketujuh belas menggunakan *video testing* 6-2. Dalam *video testing* 6-2 mempunyai 8,854 *frames*, dengan rincian 298 *frames* yang terdeteksi dan 8,556 *frames* yang tidak dapat terdeteksi. Adapun beberapa *frames* tambahan pada *frames* tertentu berjumlah 2 *frames*. Gambar 14 merupakan *data testing* 6-2.
- 18) *Video Testing* 6-3: Pengujian kedelapan belas menggunakan *video testing* 6-3. Dalam *video testing* 6-3 mempunyai 8,958 *frames*, dengan rincian 41 *frames* yang terdeteksi dan 8,917 *frames* yang tidak dapat terdeteksi. Adapun beberapa *frames* tambahan pada *frames* tertentu berjumlah 1 *frames*. Gambar 14 merupakan *data testing* 6-3.
- 19) *Video Testing* 7-2: Pengujian kesembilan belas menggunakan *video testing* 7-2. Dalam *video testing* 7-2 mempunyai 7,948 *frames*, dengan rincian 1,260 *frames* yang terdeteksi dan 6,688 *frames* yang tidak dapat terdeteksi. Adapun beberapa *frames* tambahan pada *frames* tertentu berjumlah 36 *frames*. Gambar 15 merupakan *data testing* 7-2.
- 20) *Video Testing* 7-3: Pengujian kedua puluh menggunakan *video testing* 7-3. Dalam *video testing* 7-3 mempunyai 8,879 *frames*, dengan rincian 155 *frames* yang terdeteksi dan 8,724 *frames* yang tidak dapat terdeteksi. Adapun beberapa *frames* tambahan pada *frames* tertentu berjumlah 1 *frames*. Gambar 15 merupakan *data testing* 7-3.



Gambar 22. Contoh Data Testing 14-1 - 14-3.



- 21) *Video Testing 8-1*: Pengujian kedua puluh satu menggunakan video testing 8-1. Dalam video testing 8-1 mempunyai 18,120 frames, dengan rincian 9,326 frames yang terdeteksi dan 8,794 frames yang tidak dapat terdeteksi. Adapun beberapa frames tambahan pada frames tertentu berjumlah 257 frames. Gambar 16 merupakan data testing 8-1.
- 22) *Video Testing 8-2*: Pengujian kedua puluh dua menggunakan video testing 8-2. Dalam video testing 8-2 mempunyai 8,920 frames, dengan rincian 2,249 frames yang terdeteksi dan 6,671 frames yang tidak dapat terdeteksi. Adapun beberapa frames tambahan pada frames tertentu berjumlah 46 frames. Gambar 16 merupakan data testing 8-2.
- 23) *Video Testing 8-3*: Pengujian kedua puluh tiga menggunakan video testing 8-3. Dalam video testing 8-3 mempunyai 9,124 frames, dengan rincian 1,803 frames yang terdeteksi dan 7,321 frames yang tidak dapat terdeteksi. Adapun beberapa frames tambahan pada frames tertentu berjumlah 99 frames. Gambar 16 merupakan data testing 8-3.
- 24) *Video Testing 9-2*: Pengujian kedua puluh empat menggunakan video testing 9-2. Dalam video testing 9-2 mempunyai 18,227 frames, dengan rincian 7,275 frames yang terdeteksi dan 10,952 frames yang tidak dapat terdeteksi. Adapun beberapa frames tambahan pada frames tertentu berjumlah 319 frames. Gambar 17 merupakan data testing 9-2.
- 25) *Video Testing 9-3*: Pengujian kedua puluh lima menggunakan video testing 9-3. Dalam video testing 9-3 mempunyai 18,159 frames, dengan rincian 5,247 frames yang terdeteksi dan 12,912 frames yang tidak dapat terdeteksi. Adapun beberapa frames tambahan pada frames tertentu berjumlah 237 frames. Gambar 17 merupakan data testing 9-3.
- 26) *Video Testing 10-1*: Pengujian kedua puluh enam menggunakan video testing 10-1. Dalam video testing 10-1 mempunyai 17,867 frames, dengan rincian 70 frames yang terdeteksi dan 17,797 frames yang tidak dapat terdeteksi. Adapun beberapa frames tambahan pada frames tertentu berjumlah 1 frames. Gambar 18 merupakan data testing 10-1.
- 27) *Video Testing 10-3*: Pengujian kedua puluh tujuh menggunakan video testing 10-3. Dalam video testing 10-3 mempunyai 17,899 frames, dengan rincian 1,072 frames yang terdeteksi dan 16,827 frames yang tidak dapat terdeteksi. Adapun beberapa frames tambahan pada frames tertentu berjumlah 10 frames. Gambar 18 merupakan data testing 10-3.
- 28) *Video Testing 11-1*: Pengujian kedua puluh delapan menggunakan video testing 11-1. Dalam video testing 11-1 mempunyai 18,158 frames, dengan rincian 6,506 frames yang terdeteksi dan 11,652 frames yang tidak dapat terdeteksi. Adapun beberapa frames tambahan pada frames tertentu berjumlah 244 frames. Gambar 19 merupakan data testing 11-1.
- 29) *Video Testing 11-2*: Pengujian kedua puluh sembilan menggunakan video testing 11-2. Dalam video testing 11-2 mempunyai 17,886 frames, dengan rincian 1,009 frames yang terdeteksi dan 16,877 frames yang tidak dapat terdeteksi. Adapun beberapa frames tambahan pada frames tertentu berjumlah 3 frames. Gambar 19 merupakan data testing 11-2.
- 30) *Video Testing 11-3*: Pengujian ketiga puluh menggunakan video testing 11-3. Dalam video testing 11-3 mempunyai 17,877 frames, dengan rincian 349 frames yang terdeteksi dan 17,528 frames yang tidak dapat terdeteksi. Adapun beberapa frames tambahan pada frames tertentu berjumlah 2 frames. Gambar 19 merupakan data testing 11-3.
- 31) *Video Testing 12-1*: Pengujian tiga puluh satu menggunakan video testing 12-1. Dalam video testing 12-1 mempunyai 17,900 frames, dengan rincian 9,987 frames yang terdeteksi dan 7,913 frames yang tidak dapat terdeteksi. Adapun beberapa frames tambahan pada frames tertentu berjumlah 1 frames. Gambar 20 merupakan data testing 12-1.
- 32) *Video Testing 13-1*: Pengujian tiga puluh dua menggunakan video testing 13-1. Dalam video testing 13-1 mempunyai 17,934 frames, dengan rincian 2,078 frames yang terdeteksi dan 15,856 frames yang tidak dapat terdeteksi. Adapun beberapa frames tambahan pada frames tertentu berjumlah 32 frames. Gambar 21 merupakan data testing 13-1.
- 33) *Video Testing 13-2*: Pengujian tiga puluh tiga menggunakan video testing 13-2. Dalam video testing 13-2 mempunyai 18,632 frames, dengan rincian 6,664 frames yang terdeteksi dan 11,968 frames yang tidak dapat terdeteksi. Adapun beberapa frames tambahan pada frames tertentu berjumlah 732 frames. Gambar 21 merupakan data testing 13-2.
- 34) *Video Testing 14-1*: Pengujian tiga puluh empat menggunakan video testing 14-1. Dalam video testing 14-1 mempunyai 17,885 frames, dengan rincian 2,614 frames yang terdeteksi dan 15,271 frames yang tidak dapat terdeteksi. Adapun beberapa frames tambahan pada frames tertentu berjumlah 2 frames. Gambar 22 merupakan data testing 14-1.
- 35) *Video Testing 14-2*: Pengujian tiga puluh lima menggunakan video testing 14-2. Dalam video testing 14-2 mempunyai 17,862 frames, dengan rincian 1,149 frames yang terdeteksi dan 16,713 frames yang tidak dapat terdeteksi. Adapun beberapa frames tambahan pada frames tertentu berjumlah 1 frames. Gambar 22 merupakan data testing 14-2.
- 36) *Video Testing 14-3*: Pengujian tiga puluh enam menggunakan video testing 14-3. Dalam video testing 14-3 mempunyai 17,970 frames, dengan rincian 1,102 frames yang terdeteksi dan 16,868 frames yang tidak dapat terdeteksi. Adapun beberapa frames tambahan pada frames tertentu berjumlah 52 frames. Gambar 22 merupakan data testing 14-3.

## V. KESIMPULAN

Pada penelitian ini dapat ditarik beberapa kesimpulan antara lain:

1. Model dapat mengklasifikasikan kondisi mata terbuka dan mata tertutup dengan menggunakan *Convolutional Neural Network* dengan arsitektur yang memiliki 3 layer mendapatkan hasil *accuracy* 98% dan *error* 2.05%.

2. Model dari hasil *training* dapat berjalan secara *real-time*.
3. Pendeteksian secara *real-time* memiliki kendala bila wajah terhalangi oleh objek lain.

#### DAFTAR PUSTAKA

- [1] W. Zhang, B. Cheng, and Y. Lin, "Driver drowsiness recognition based on computer vision technology," *Tsinghua Sci. Technol.*, vol. 17, no. 3, pp. 354–362, 2012, doi: 10.1109/TST.2012.6216768.
- [2] S. Abidin, "Deteksi wajah menggunakan metode Haar Cascade Classifier berbasis webcam pada Matlab," *J. Teknol. Elekterika*, vol. 15, no. 1, pp. 21–27, 2018, doi: 10.31963/ELEKTERIKA.V15I1.2102.
- [3] N. S. Ibrahim, "Aplikasi Android Deteksi Mata Menggunakan Metode Viola-Jones," Universitas Dian Nuswantoro, Semarang, Indonesia, 2015.
- [4] N. L. Fitriyani, C. K. Yang, and M. Syafrudin, "Real-time eye state detection system using haar cascade classifier and circular hough transform," 2016, doi: 10.1109/GCCE.2016.7800424.
- [5] Y. LeCun, K. Kavukcuoglu, and C. Farabet, "Convolutional networks and applications in vision," in *ISCAS 2010 - 2010 IEEE International Symposium on Circuits and Systems: Nano-Bio Circuit Fabrics and Systems*, 2010, pp. 253–256, doi: 10.1109/ISCAS.2010.5537907.
- [6] Y. Yoshidan, "GitHub - yoshidan/pytorch-eyeblick-detection: Simple Eye Blink Detection with CNN." [Online]. Available: <https://github.com/yoshidan/pytorch-eyeblick-detection>.
- [7] C. B. S. Maior, M. J. das C. Moura, J. M. M. Santana, and I. D. Lins, "Real-time classification for autonomous drowsiness detection using eye aspect ratio," *Expert Syst. Appl.*, vol. 158, p. 113505, 2020, doi: 10.1016/J.ESWA.2020.113505.
- [8] T. Pradhan, A. N. Bagaria, and A. Routray, "Measurement of PERCLOS using eigen-eyes," 2012, doi: 10.1109/IHCI.2012.6481864.