

Deteksi Pengendara Mengantuk dengan Kombinasi *Haar Cascade Classifier* dan *Support Vector Machine*

Hepatika Zidny Ilmadina¹, Dyah Apriliani², Dega Surono Wibowo³

^{1,2,3} Program Studi Teknik Informatika, Politeknik Harapan Bersama,

^{1,2,3} Jalan Mataran No 9, Kota Tegal, 50272, Indonesia

email: hepatica.zidny@poltektegal.ac.id

Abstract – Microsleep often occurs in our activities without realizing it, especially when driving. Microsleep is one of the factors that cause accidents caused by human errors such as drowsiness, unfocused, and fatigue, contributing 61% of the share, 9% due to vehicle factors, and 30% due to infrastructure environmental factors. Several microsleeps detection techniques through the blink of an eye have been developed, including using eye aspect ratio, namely manual thresholding by setting the minimum second of eye closing/blinking. In this study, the detection of drowsy drivers was carried out using Haar Cascade, which classifies the rider's face, and the Support Vector Machine method to determine closed and open eyes—implementing the drowsy driver detection system in real-time reaches 99% accuracy. This research is expected to help reduce and even prevent accidents caused by microsleep which can be detected early through the blink of the eye of the driver.

Abstrak – Microsleep sering terjadi pada aktivitas kita tanpa disadari, terutama pada saat berkendara. Hal tersebut menjadi salah satu faktor penyebab kecelakaan yang diakibatkan kesalahan manusia seperti mengantuk, tidak fokus, dan kelelahan menyumbangkan porsi 61%, selain itu 9% dikarenakan faktor kendaraan, serta 30% disebabkan oleh faktor prasarana dan lingkungan. Beberapa teknik deteksi *microsleep* melalui kedipan mata telah dikembangkan antara lain menggunakan *eye aspect ratio*, yaitu *thresholding* manual dengan menetapkan detik minimal mata menutup/berkedip. Pada penelitian ini dilakukan deteksi pengendara mengantuk dengan kombinasi *Haar Cascade* yang mengklasifikasi wajah pengendara, serta metode *Support Vector Machine* yang mampu menentukan mata menutup dan membuka. Implementasi pada sistem deteksi pengendara mata mengantuk secara *real time* akurasi mencapai 99%. Penelitian ini diharapkan dapat membantu dalam mengurangi bahkan mencegah terjadinya kecelakaan yang disebabkan oleh *microsleep* yang dapat dideteksi secara dini melalui kedipan mata pengendara.

Kata Kunci – *microsleep*, *eye aspect ratio*, *support vector machine*, *haar cascade classifier*

I. PENDAHULUAN

Dalam keadaan mengantuk mungkin kita sering tidak sadar bahwa kita sering tidak sengaja tertidur saat sedang beraktivitas. Menutupnya mata tanpa sadar tersebut merupakan

tanggapan terhadap stimulus, dimana terjadi penyimpangan perhatian yang dapat mengganggu kemampuan untuk mendeteksi dan menanggapi rangsangan pada suatu peristiwa penting [1]. Menurut Rivera dan Salas (2013), pengemudi yang mengantuk sendiri memiliki kontribusi sebelas kali lipat dalam kecelakaan serius dan kerusakan pada jangka waktu terlelap yang singkat berkisar 1-30 detik [2].

Mengacu pada informasi diatas dapat disimpulkan bahwa mengemudi dalam kondisi mengantuk menjadi suatu fenomena yang berbahaya dalam berkendara. Berdasarkan data yang diambil dari Direktorat Lalu Lintas Polda Metro Jaya (Ditlantas) Polda Metro Jaya, dilaporkan adanya 7.565 kasus kecelakaan selama tahun 2020 dengan korban luka berat berjumlah 1.565 orang dan korban meninggal dunia pun mencapai 559 orang. Diperoleh informasi bahwa beberapa faktor penyebab kecelakaan antara lain 61% disebabkan kesalahan manusia seperti mengantuk, tidak fokus, dan kelelahan, 9% dikarenakan faktor kendaraan itu sendiri, serta 30% disebabkan oleh faktor prasarana dan lingkungan. Tiga faktor kecelakaan tersebut paling besar berkontribusi oleh kesalahan manusia salah satunya mengendarai kendaraan dalam kondisi mengantuk.

Masih banyak penelitian yang dikembangkan untuk mengurangi kecelakaan lalu lintas dengan beberapa metode dalam mengidentifikasi pengendara yang mengantuk. Hal tersebut telah terimplementasi melalui metode berbasis fisiologis, berbasis kendaraan itu sendiri, serta berbasis perilaku. Metode berbasis fisiologis dilakukan dengan mengukur dan memonitor irama detak jantung, denyut nyadi, serta electrocardiogram yang juga berhubungan dengan tingkat kelelahan [3] [4]. Metode deteksi pengendara mengantuk juga dapat diimplementasikan langsung pada kendaraan dengan mengukur akselerasi dan pergerakan atau pola kendaraan berjalan. Basis perilaku dilakukan dengan melihat kondisi perilaku seperti menguap, menutup mata sesaat, mengedipkan mata, dll [5].

Pada pendekatan berbasis kendaraan sudah terimplementasi pada kendaraan autonomous yang bisa berkendara sesuai dengan garis jalan raya karena didukung dengan *Engine Control Unit* (ECU) sebagai pengatur kecepatan mobil secara otomatis [6] [7]. Akan tetapi hal tersebut bukan menjadi solusi yang tepat untuk mengurangi kecelakaan yang diakibatkan oleh pengendara mengantuk dikarenakan harga kendaraan yang mahal serta tidak semua

*) penulis korespondensi: Hepatika Zidny Ilmadina

Email: hepatica.zidny@poltektegal.ac.id

episode singkat dari perilaku pola tidur dan penghentian

perusahaan kendaraan ternama mengimplementasikan sistem tersebut.

Beberapa teknik deteksi pengendara mengantuk dewasa ini telah dikembangkan oleh beberapa peneliti. Deteksi pengendara mengantuk telah dikembangkan melalui kedipan mata dengan memperhitungkan *eye aspect ratio*, yaitu sebuah metode untuk melakukan *thresholding* secara manual dengan menetapkan detik minimal mata menutup/berkedip [7] [8] [9] [10]. Penelitian dilakukan dengan mengukur rasio (jarak) antar kelopak mata bagian atas dengan kelopak mata bagian bawah. Jika nilai jarak antar kelopak mata kurang dari *threshold* yang ditentukan, maka akan memberikan notifikasi mata pengendara tersebut terpejam sesaat atau mengantuk.

Dalam menghitung rasio jarak antar kelopak mata tersebut diperlukan metode pendeteksian yang tepat agar sistem mampu menghitung dengan baik. Beberapa peneliti menggunakan metode Histogram of Gradients (HoG) dan Haar Cascade Classifier. Dalam praktiknya kedua metode tersebut masih sering melakukan kesalahan deteksi terutama pada objek lain yang di sekitar wajah yang menyerupai mata. Oleh karena itu, peneliti akan melakukan kombinasi pendeteksian mata dengan algoritma *machine learning* agar lebih tepat dalam mendeteksi mata sebelum mengukur *eye aspect ratio*-nya.

Adanya penelitian ini diharapkan dapat membantu dalam mengurangi bahkan mencegah terjadinya kecelakaan yang disebabkan oleh pengendara mengantuk yang dapat dideteksi secara dini melalui tempo pengendara dalam memejamkan kedua mata.

II. PENELITIAN YANG TERKAIT

Eye aspect ratio digunakan dalam mengukur rasio atau jarak yang dihasilkan antar kelopak mata (bagian atas dan bagian bawah). Rasio yang dihitung dideteksi berdasarkan kedipan yang dilakukan oleh pengendara. Rasio itu akan diperoleh dari rekaman kedipan mata yang terimplementasi dalam sistem berdasarkan koordinat mata yang terdeteksi [7] [11]. Novie Theresia Br. Pasaribu, dkk mengembangkan deteksi pengendara mengantuk dengan memodifikasi jumlah kedipan serta perhitungan *eye aspect ratio* [10]. Perkembangan penelitian deteksi mata mengantuk dengan mengukur *eye aspect ratio* masih dilakukan oleh beberapa peneliti. Perhitungan rasio tersebut dilakukan ketika sistem mampu mengenali mata dari pengendara. Sistem deteksi pengendara mengantuk pernah dikembangkan dengan melakukan pengenalan wajah pengendara terlebih dahulu yang kemudian dilanjutkan dengan pendeteksian kedua mata [7]. Setelah kedua mata terdeteksi dengan baik, perhitungan *eye aspect ratio* dilakukan dengan tujuan memberikan batasan kapan diberikan peringatan bahwa pengemudi tersebut mengantuk atau tidak. Metode ini kemudian dikembangkan oleh S. Ajay (2020) untuk memberikan peringatan lewat pengiriman *short message service* (SMS) selain dilakukan peringatan alarm. Pengiriman SMS dan alarm akan berbunyi bersamaan ketika nilai *eye aspect ratio* mencapai batasan yang telah ditentukan [12] [13]. Kelemahan yang ditemukan dari penelitian serupa tersebut adalah kurang berhasilnya metode dalam mendeteksi mata sebagai syarat perhitungan *eye aspect ratio*. Hal ini disebabkan teknik deteksi terkendala dengan objek lain yang ada di wajah yang menyerupai mata. Pengembangan selanjutnya dilakukan untuk memastikan area mata sebagai deteksi utama pengendara mengantuk dengan baik adalah

mengkombinasikannya dengan sebuah teknik *feature extraction* yaitu dengan *Uniform Local Binary Pattern* (ULBP). Teknik ekstraksi fitur yang digunakan berhasil mengenali mata dengan baik. Akan tetapi teknik ini memiliki kelemahan pada akurasi yang kecil ketika pengendara menggunakan kacamata [14]. Implementasi dengan komponen perangkat keras dan dikombinasikan dengan teknik deteksi juga pernah diteliti oleh C. Ryan (2021). Pada penelitian tersebut menggunakan *event camera* yang kemudian melakukan deteksi wajah secara *real time*. Proses selanjutnya sama dengan penelitian sebelumnya, yakni dengan melakukan pendeteksian mata dan menghitung *eye aspect ratio* yang dihasilkan dari kedipan mata pengemudi [9]. Kekurangan penelitian-penelitian terdahulu yang pernah dikembangkan adalah mencakup kurang akuratnya deteksi mata sebagai komponen utama perhitungan *eye aspect ratio* dan beberapa terkendala dari objek yang memiliki *noise* berupa kacamata. Oleh karena itu diperlukan suatu kombinasi model klasifikasi sehingga deteksi mata dapat dilakukan dengan baik sehingga perhitungan kedipan maupun *eye aspect ratio* dapat dilakukan dengan baik.

III. METODE PENELITIAN

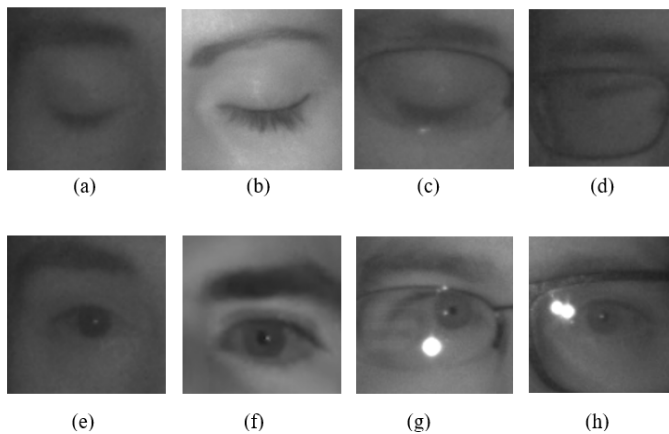
Penelitian ini bertujuan untuk mendeteksi pengendara yang mengantuk secara dini melalui kedipan mata dan *eye aspect ratio* yang dikombinasikan dengan klasifikasi algoritma *machine learning*. Tahapan awal dilakukan dengan data acquisition berupa pengumpulan citra pengenalan mata yang memejam atau tidak memejam melalui berbagai sumber. Tahapan selanjutnya adalah melakukan proses pelatihan dataset tersebut pada beberapa model *machine learning* yang dibangun. Model *machine learning* akan dilatih dengan menggunakan parameter terbaik yang diperoleh dengan menggunakan *grid search*. Evaluasi model kemudian dilakukan untuk menentukan model terbaik. Langkah selanjutnya adalah menguji secara *real time* pada kondisi memejam dan membuka menggunakan model terbaik. Secara umum tahapan penelitian diilustrasikan pada gambar di bawah ini.



Gbr. 1 Ilustrasi metode penelitian yang digunakan

A. Data Acquisition

Penelitian ini menggunakan beberapa sumber dataset seperti mata menutup dari UMass Amherst (13233 gambar), mata membuka dari Nanjing University (2423 gambar), serta beberapa dataset mata membuka dan menutup dari Kaggle (2900 gambar) [15]. Masing-masing dataset tersebut terdiri atas komponen mata sebelah kanan, mata sebelah kiri laki-laki dan perempuan, tanpa kacamata serta menggunakan kacamata.

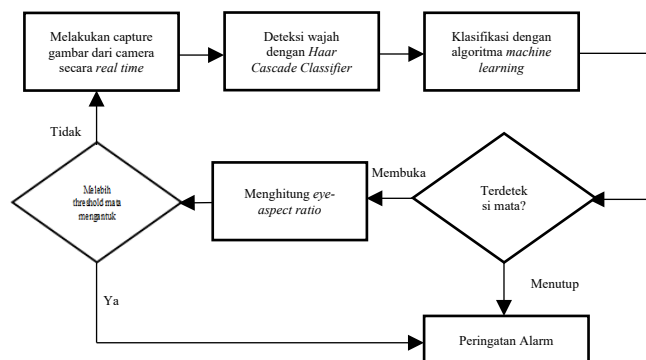


Gbr. 2 Dari kiri ke kanan dataset yang digunakan; (a)-(d) Variasi mata menutup dengan dan tanpa kacamata (e)-(h) Variasi mata membuka dengan dan tanpa kacamata

Dataset yang digunakan memiliki variasi citra mata dari subjek perempuan dan laki-laki dengan kondisi mata membuka dan mata menutup. Selain itu juga disediakan citra subjek dengan menggunakan kacamata dan tidak menggunakan kacamata.

B. Implementasi Model

Model dalam penelitian ini bertujuan untuk mengklasifikasikan mata dalam kondisi membuka maupun kondisi menutup secara lebih akurat dibandingkan dengan teknik deteksi biasa. Model ini diharapkan dapat mengoptimalkan kemampuan *haar cascade classifier* dalam mendeteksi objek mata. Jika mata terdeteksi dengan baik, maka perhitungan *eye aspect ratio* sebagai tanda kondisi mengantuknya pengemudi juga akan semakin akurat. Ketika validasi menutupnya mata bukan merupakan mata mengedip, maka akan ada peringatan yang ditunjukkan dengan bunyinya alarm. Secara umum digambarkan pada bagan alir berikut ini.

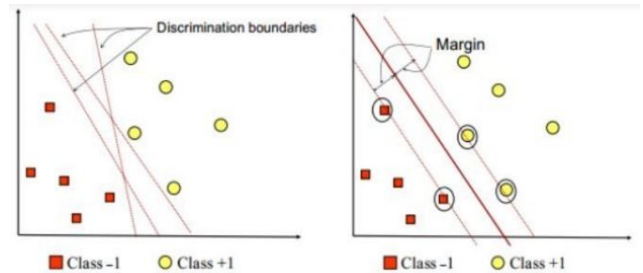


Gbr. 3 Alur metodologi pendeteksian mata mengantuk secara *real time*

Model klasifikasi mata menutup sebagai tanda pengemudi yang mengantuk akan dilatih dengan dataset dengan model klasifikasi *Support Vector Machine (SVM)* dan *k-Nearest Neighbors (KNN)*.

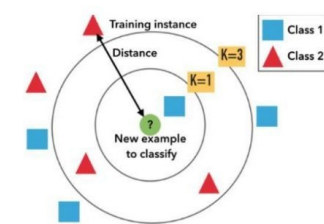
SVM merupakan metode *supervised* pada *machine learning*. Proses pelatihan/pengenalan yang dilakukan pada algoritma SVM bertujuan untuk memperoleh hipotesis yaitu *hyperplane* terbaik yang mampu mengklasifikasi data dengan baik pada setiap kelasnya. SVM mampu bekerja

pada data linear dan nonlinear separable. Pada Gambar dibawah ini, terlihat bahwa berbagai *alternative hyperplane* yang dapat memisahkan semua data ke dalam kelas yang sesuai. *Hyperplane* terbaik tidak hanya mampu untuk memisahkan data, tetapi juga memiliki margin paling besar.



Gbr. 4 Ilustrasi SVM

Sedangkan *k-nearest neighbor* (k-NN atau kNN) adalah sebuah metode untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. Proses klasifikasi dilakukan dengan mencari titik c terdekat dari c-baru (*nearest neighbor*). Teknik pencarian tetangga terdekat yang umum dilakukan dengan menggunakan formula jarak *Euclidean* antar data satu dengan data lainnya.



Gbr. 5 Ilustrasi kNN

Kedua metode akan diimplementasikan dengan parameter terbaik yang dicari dengan menggunakan *grid search*, sehingga akan mendapatkan akurasi klasifikasi yang maksimal.

C. Haar Cascade Classifier

Kesuksesan aplikasi *computer vision* pertama kali dimulai pada tahun 2001, ketika Paul Viola dan Michael Jones mengajukan framework pertama Object Detection dalam deteksi video secara *real time* serta tertuang dalam paper berjudul "*Rapid Object Detection using a Boosted Cascade of Simple Features*" (2001) [16]

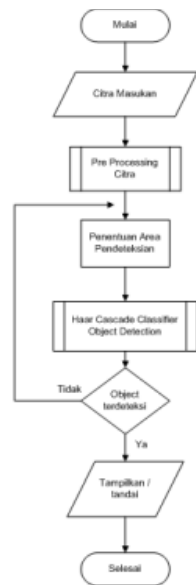
1) Sistem kerja dari *cascade classifier* itu sendiri, antara lain:

- Algoritma ini memerlukan gambar positif (dengan muka) dan gambar negatif (tanpa muka) untuk melatih classifier
- Haar cascade mampu mendeteksi suatu bagian dalam suatu gambar, tetapi dapat dilatih untuk mengidentifikasi hampir semua objek
- Haar feature extraction dilakukan untuk menggali informasi objek, salah satu metode edge detection

- Setiap cascade merupakan nilai tunggal yang diperoleh atas selisih proporsi jumlah fitur pada kotak putih dari jumlah proporsi fitur pada kotak hitam atau sebaliknya

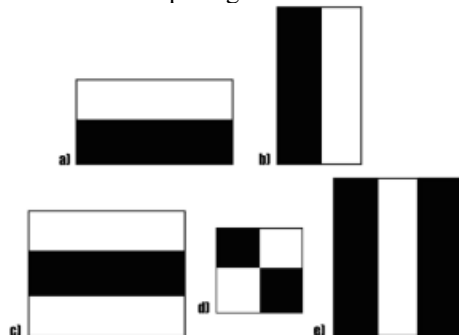
2) Cara kerja cascade classifier secara umum
Haar Cascade menerapkan cascade function untuk melatih gambar melalui 4 tahapan utama, yaitu:

- Haar features selection,
- Membuat gambar integral,
- Adaboost training
- Melakukan klasifikasi dengan cascading classifier.



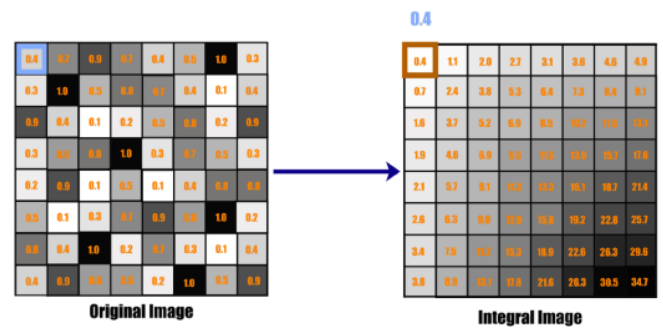
Gbr. 6 . Alur *cascade classifier*

Area yang lebih gelap pada haar feature adalah bernilai pixel 1, dan yang lebih terang memiliki nilai pixel 0, ditunjukkan lebih rinci pada gambar di bawah ini.



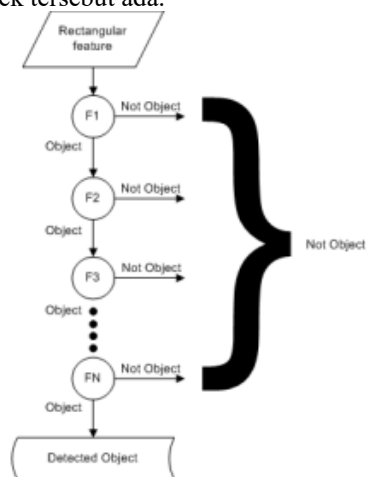
Gbr. 7 . Kotak Fitur *Haar*

Gambar Integral kemudian dihitung dari Gambar Asli sedemikian rupa sehingga setiap piksel dalam hal ini adalah jumlah dari semua piksel yang terletak di sebelah kiri dan di atasnya pada Gambar Asli menyesuaikan ukuran kotak fitur Haar yang terbentuk.



Gbr. 8 . Proses Gambar Integral

Selanjutnya dilakukan *Adaboost training* yang merupakan salah satu algoritma feature selection dengan konsep kerja memilih subset fitur yang paling representatif dari jumlah fitur yang banyak, serta melakukan eliminasi untuk fitur yang tidak relevan. Proses tersebut kemudian dilakukan klasifikasi dengan *cascade* yang juga disebut *The Attentional Cascade*. Idennya adalah tidak semua fitur harus berjalan di setiap jendela/kotak fitur haar. Jika suatu fitur gagal pada jendela tertentu, maka kita dapat mengatakan bahwa fitur objek tidak ada di sana. Oleh karena itu, kita dapat pindah ke jendela berikutnya di mana ada fitur objek tersebut ada.



Gbr. 9 . Proses Klasifikasi pada *Cascade*

D. Evaluasi Performa

Performa dari klasifikasi yang dilakukan dengan kedua metode yang akan dibandingkan diukur dengan menghitung nilai *accuracy*, *precision*, dan *recall*. Nilai akurasi menunjukkan ukuran rasio model dalam mengklasifikasi dengan benar dibandingkan dengan jumlah klasifikasi keseluruhan. *Precision* merupakan rasio terklasifikasinya nilai positif terhadap seluruh hasil klasifikasi positif, sedangkan *Recall* atau *sensitivity* adalah rasio terhadap kelas yang seharusnya positif.

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

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

Keterangan:

TP: hasil prediksi positif dan nilai sebenarnya juga bernilai positif (*true positive*)

TN: hasil prediksi negatif dan nilai sebenarnya juga bernilai negatif (*true negative*)

FN: hasil prediksi negatif sedangkan nilai sebenarnya positif (*false negative*)

FP: hasil prediksi positif sedangkan nilai sebenarnya negatif (*false positive*)

IV. HASIL DAN PEMBAHASAN

A. Pelatihan dan Evaluasi Model

Penelitian ini akan diawali dengan melakukan pelatihan pada dua metode yang diusulkan, yaitu SVM dan KNN yang menggunakan parameter terbaik hasil dari *grid search CV*. Hasil model dengan evaluasi terbaik kemudian digunakan dalam implementasi sistem deteksi pengendara mengantuk. Sistem ini akan mendeteksi wajah pengendara dengan metode *Haar Cascade Classifier*. Wajah yang terdeteksi kemudian akan diklasifikasikan mata kanan dan kiri dengan model implementasi terbaik. Ketika mata berhasil terklasifikasi, kedepan mata nantinya akan memiliki nilai yang menjadi tanda bahwa pengendara tersebut mengantuk. Semakin lama pengendara memejamkan mata pada saat berkedip maka akan mengaktifkan alarm peringatan.

Sebelum diimplementasikan secara *real time*, model dilatih dengan menggunakan dataset yang telah disiapkan. Model dengan evaluasi performa terbaik nantinya yang akan dikombinasikan dengan *Haar Cascade Classifier* untuk sistem deteksi pengendara mengantuk. Parameter dari kedua model ditetapkan dengan *Grid Search CV* untuk mendapatkan hasil pelatihan yang optimal. Pada model SVM dan KNN dilakukan proses pelatihan dengan parameter yang dijelaskan pada Tabel 1 di bawah ini.

TABEL 1 PARAMETER EKSPERIMEN

Model	Parameter	Nilai
SVM	C	0.1
	gamma	1
	kernel	linear
KNN	number of neighbours	11
	metric	Euclidean
	weights	uniform

Pelatihan dilakukan dengan melakukan pelatihan untuk masing-masing model yang kemudian dilakukan evaluasi untuk menentukan model terbaik dalam mengklasifikasikan mata menutup yang merupakan tanda pengendara mengantuk. Model yang memiliki evaluasi lebih baik akan disimpan untuk diimplementasikan bersama *Haar Cascade Classifier* dalam implementasi secara *real time*.

B. Hasil Perbandingan Metode SVM dan KNN

Implementasi penelitian ini dilakukan dengan menggunakan *Python* dan memanfaatkan *Keras* dengan *tensorflow backend*. Pelatihan dilakukan pada seluruh

dataset untuk mendapatkan model terbaik dalam melakukan klasifikasi mata memejam yang merupakan tanda pengendara mengantuk. Hasil evaluasi pelatihan klasifikasi model SVM dan KNN dijelaskan pada Tabel 2.

TABEL 2 PERBANDINGAN HASIL AKURASI TRAINING DENGAN DATASET MATA MEMBUKA DAN MENUTUP

Metode	Accuracy	Precision	Recall
SVM	0.99	0.99	0.99
KNN	0.97	0.89	0.99

Berdasarkan data tersebut dapat disimpulkan bahwa performa yang unggul dicapai oleh model SVM dibandingkan dengan KNN pada saat diimplementasikan untuk mengklasifikasikan mata membuka dan menutup. Selain itu evaluasi performa *accuracy*, *precision*, dan *recall* untuk setiap kelas yang terklasifikasi juga dijelaskan pada Tabel 3.

TABEL 3 PERBANDINGAN HASIL EVALUASI PADA MATA YANG MEMBUKA DAN MENUTUP PADA PERCOBAAN

Metode	SVM		KNN	
	Closed	Open	Closed	Open
Precision	0.89	1.00	0.89	0.97
Recall	1.00	0.99	1.00	0.98
f1-score	1.00	1.00	0.99	0.98

Melihat evaluasi performa yang disajikan pada tabel 3 dan Tabel 4, maka pada penelitian ini akan dilakukan dengan menggunakan metode klasifikasi SVM yang dikombinasikan dengan deteksi wajah *Haar Cascade Classifier*. SVM yang digunakan telah dilatih dengan dataset sesuai dengan parameter pada Tabel 1. SVM adalah algoritma yang dikenalkan oleh Vladimir N. Vapnik yang kemudian dikembangkan oleh Cortes dan Vapnik di tahun 1933. SVM merupakan algoritma *supervised learning* yang sering digunakan dalam beberapa ranah klasifikasi citra, seperti *face recognition*, *medical image*, dan *text categorization* [17].

Model SVM kemudian diimplementasikan secara *real time* pada sistem deteksi yang dikombinasikan dengan *haar cascade classifier*. Pada sistem ini kamera akan menangkap gambar yang kemudian *haar cascade* akan mendeteksi wajah pengemudi. Wajah pengemudi yang terdeteksi kemudian akan diklasifikasikan bagian mata kiri dan kanan dengan menggunakan SVM. Hasil klasifikasi tersebut kemudian dilanjutkan dengan perhitungan nilai tunggu mata yang terklasifikasi menutup. Waktu tunggu yang melebihi dari batas ambang yang ditentukan akan mengaktifkan alarm peringatan yang menunjukkan adanya pengendara yang mengantuk.



Gbr. 10 Dari kiri ke kanan hasil implementasi sistem *real time*; (a) Mata dalam keadaan membuka; (b) Mata melirik ke bawah; (c) Mata berkedip; (d) Mata menutup selama beberapa saat

Dari gambar 10 dapat kita lihat dengan kondisi mata membuka, sistem akan memberikan notifikasi mata membuka dengan *score* 0 yang artinya tidak ada sistem alarm yang diaktifkan. Begitu juga ketika mata dalam kondisi melirik ke bawah seolah-olah menutup dan berkedip, sistem tidak mendeteksi adanya pengemudi yang mengantuk. Ketika mata menutup seperti pada gambar (d) maka sistem akan memberikan keterangan "Closed" dengan jumlah *score* yang mengaktifkan alarm peringatan.

Dibandingkan dengan penelitian terdahulu yang melakukan deteksi pengemudi mengantuk dengan perhitungan *eye aspect ratio* tanpa adanya klasifikasi dengan algoritma *machine learning* [7] [10] [11] [12] [13], metode usulan pada penelitian ini mampu mendeteksi mata dengan baik dengan waktu deteksi yang relatif singkat. Hal ini dikarenakan, model SVM yang mampu mengklasifikasikan mata pada hasil deteksi wajah dengan baik. Klasifikasi dilakukan setiap mata melakukan pergerakan membuka maupun berkedip. Sistem akan otomatis melakukan klasifikasi mata pengemudi tersebut membuka atau menutup.

V. KESIMPULAN

Pada penelitian ini diimplementasikan sebuah sistem deteksi pengemudi mengantuk berdasarkan klasifikasi kedipan mata yang menjadi tanda *microsleep*. Deteksi kedipan mata dapat dilakukan dengan baik menggunakan kombinasi *Haar Cascade Classifier* dengan SVM dan mendapatkan akurasi pelatihan model sebesar 99%. Hal tersebut menunjukkan model SVM mampu mengklasifikasi mata berkedip dan membuka dengan baik yang telah dideteksi terlebih dahulu dengan *Haar Cascade Classifier*.

UCAPAN TERIMA KASIH

Ucapan terima kasih penulis kepada Politeknik Harapan Bersama yang membantu ataupun memberikan dukungan

terkait dengan penelitian yang dilakukan.

DAFTAR PUSTAKA

- [1] L. N. Boyle, J. Tiffin, A. Paul dan M. Rizzo, "Driver performance in the moments surrounding a microsleep," *Transportation Research Part F: Traffic Psychology and Behaviour*, vol. 11, no. 2, pp. 126-136, 2008.
- [2] M. Rivera dan L. Salas, "Monitoring of Micro-sleep and Sleepiness for the Drivers Using EEG Signal," School of Innovation, Design and Engineering (IDT), Mälardalen University, Västerås, Sweden, 2013.
- [3] T. Hwang, M. Kim, S. Hong dan K. S. Park, "Driver drowsiness detection using the in-ear EEG," dalam *Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Orlando, FL, USA, 2016.
- [4] S. Junawane, P. Jagtap, L. Deshpande, K. Soni dan R. Jab, "Real-time Driver Drowsiness Detection for Android Application Using Deep Neural Networks Techniques," *Procedia Computer Science*, vol. 6, no. 11, pp. 400 - 407, 2017.
- [5] S. Sangle, B. Rathore dan A. R. Rathod, "Real Time Drowsiness Detection System," *IOSR Journal of Computer Engineering (IOSR-JCE)*, pp. 87-92, 2018.
- [6] K. Kaida, M. Takahashi, T. Akerste, A. Nakata, Y. Otsuka, T. Haratani dan K. Fukusawa, "Validation of the Karolinska sleepiness scale against performance and EEG variables," *Clin Neurophysiol*, vol. 117, no. 7, pp. 1574-81, 2006.
- [7] S. Saravanaraj, M. Abd Kadir, S. Sharifah, S. Azmi, S. Mohamad Md dan A. Hussein Ali, "Drowsiness Detection System using Eye Aspect Ratio Technique," dalam *2020 IEEE Student Conference on Research and Development, SCORED 2020*, Michigan, 2020.
- [8] C. B. S. Maior, M. J. d. C. Moura, J. M. M. Santana dan I. D. Lins, "Real-time classification for autonomous drowsiness detection using eye aspect ratio," *Expert Systems with Applications*, p. 113505, 2020.
- [9] C. Ryan, B. O'Sullivan, A. Elrasad, A. Cahill, J. Lemley dan E. Perot, "Real-time face & eye tracking and blink detection using event cameras," *Neural Networks*, vol. 141, pp. 87-97, 2021.
- [10] N. Theresia Br. Pasaribu, A. Prijono, R. Ratnadewi, R. Pramono Adhie dan J. Felix, "Drowsiness Detection According to the Number of Blinking Eyes Specified From Eye Aspect Ratio Value Modification," dalam *Proceedings of the 1st International Conference on Life, Innovation, Change and Knowledge (ICLICK 2018)*, Bandung, 2019.
- [11] J. Cech dan T. Soukupov, "Eye Blink Detection Using Facial Landmarks," dalam *21st Computer Vision Winter Workshop*, Slovenia, 2016.
- [12] A. S. A. J. K. R. Subhashini dan J. Thomas, "Drowsiness Detection Using Eye Blink and Facial Features Image Analysis," *Medico-Legal Update*, vol. 20, no. 4, pp. 27-30, 2020.

- [13] M. S. Satyanarayana, T. M. Aruna dan Y. K. Guruprasad, "Continuous monitoring and identification of driver drowsiness alert system," *Global Transitions Proceedings*, vol. 2, no. 1, pp. 123-127, 2021.
- [14] D. A. Navastara, W. Y. M. Putra dan C. Fatichah, "Drowsiness Detection Based on Facial Landmark and Uniform Local Binary Pattern," *Journal of Physics: Conference Series*, vol. 1529, no. 5, 2020.
- [15] P. V. Patil, "Kaggle," [Online]. Available: <https://www.kaggle.com/prasadvpatil/mrl-dataset>. [Diakses 17 October 2021].
- [16] V. Paul dan J. Michael, "Rapid object detection using a boosted cascade of simple features," dalam *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, Kauai, HI, USA, 2001.
- [17] W. Ayadi, W. Elhamzi, I. Charfi dan M. Atri, "A hybrid feature extraction approach for brain MRI classificationbased on Bag-of-words," *Biomedical Signal Processing and Control*, vol. 48, pp. 144-152, 2019.