

DETEKSI KANTUK PADA PENGENDARA MOBIL MENGGUNAKAN ALGORITMA CONVOLUTIONAL NEURAL NETWORK (CNN)

Dimsyar M Al Hafiz¹, Finandra Nusantara², Khoirul Amaly³
Dimsyar417@gmail.com¹, Finandra.nusantara@gmail.com², khoirulamaly@gmail.com³

^{1,2,3}Program Studi Teknik Elektro, Fakultas Teknik, Universitas Sriwijaya

Abstrak: Rasa kantuk pengemudi adalah penyebab utama dari kecelakaan lalu lintas, yang mengakibatkan hilangnya nyawa yang berharga. Sistem meKantuk pengemudi yang andal dan tepat diperlukan untuk mencegah kecelakaan lalu lintas dan meningkatkan keselamatan lalu lintas. Berbagai sistem deteksi kantuk pengemudi telah dirancang dengan teknologi yang berbeda. *Paper* ini mengusulkan untuk mendeteksi kantuk pengemudi menggunakan *Convolution Neural Networks (CNN)* diikuti oleh analisis emosi. Analisis emosi, dalam model yang diusulkan ini, menganalisis kerangka berpikir pengemudi yang mengidentifikasi faktor motivasi untuk mengemudi yang berbeda pola. Pola mengemudi ini dianalisis berdasarkan pengenalan wajah pengemudi. Pola wajah pengemudi diperlakukan dengan 2D Convolution Neural Network (CNN) untuk mendeteksi perilaku dan emosi pengemudi. Model yang diusulkan diimplementasikan menggunakan Tensorflow. *Training* yang dilakukan pada penelitian ini dilakukan terhadap 4 model *Pure CNN*. Dalam pelatihan model dibuat menjadi dua jenis yaitu menggunakan *image augmentation* dan juga satunya tidak menggunakan *image augmentation*.

Kata Kunci—kantuk, *Convolutional Neural Network*, pengemudi, dan mobil, keselamatan, *Tensorflow*.

I. PENDAHULUAN

Perkembangan teknologi memungkinkan memperkenalkan solusi yang lebih maju dalam kehidupan sehari-hari. Ini membuat pekerjaan tidak terlalu melelahkan bagi manusia. Sistem yang dapat digunakan untuk mendeteksi kelelahan operator kendaraan menggunakan berbasis visi solusi. Kelelahan adalah kondisi psikofisik manusia, yang tidak memungkinkan untuk konsentrasi penuh. Ini mempengaruhi waktu respon manusia, karena orang yang lelah bereaksi jauh lebih lambat, dibandingkan dengan yang beristirahat. Munculnya tanda-tanda pertama kelelahan bisa menjadi sangat berbahaya, terutama untuk profesi seperti pengemudi. Saat ini, semakin banyak profesi membutuhkan

konsentrasi jangka panjang. Rakyat, yang bekerja untuk bisnis transportasi (pengemudi mobil dan truk, pengemudi, pilot pesawat), harus mengawasi jalan dengan cermat, sehingga mereka dapat bereaksi terhadap kejadian mendadak (misal kecelakaan, binatang di jalan, dll).

Di Indonesia, angka kecelakaan lalu lintas tiap tahunnya meningkat. Banyaknya kecelakaan disebabkan oleh berbagai faktor diantaranya faktor manusia, faktor kendaraan, dan faktor lingkungan [1]. Faktor kecelakaan yang disebabkan oleh manusia, Rasa kantuk saat berkendara adalah salah satu kondisi yang sering diabaikan oleh para pengendara mobil dan menjadi salah satu penyebab kecelakaan, terutama saat berkendara pada jarak yang cukup jauh [2]. Beberapa detik tertidur saat mengemudi nyatanya dapat mengakibatkan kecelakaan. Faktanya, hampir 100.000 kecelakaan lalu lintas pada 2017 disebabkan oleh pengemudi yang kantuk [3]. Berdasarkan faktor di atas, maka kecelakaan kendaraan mobil dapat diminimalisir dengan adanya pendeteksi pengendara meKantuk atau tertidur yang dipasang pada mobil [4].

Kelelahan pengemudi dapat menyebabkan *microsleep* (misalnya kehilangan konsentrasi, tidur singkat yang berlangsung dari 1 hingga 30 detik), dan tertidur di belakang kemudi. Oleh karena itu, perlu dikembangkan sebuah sistem yang akan mendeteksi dan memberi tahu pengemudi tentang kondisi psikofisiknya yang buruk, yang secara signifikan dapat mengurangi jumlah kecelakaan mobil yang berhubungan dengan kelelahan.

Proyek ini bertujuan mendeteksi kantuk saat mengemudi untuk memperingatkan pengemudi pada waktu yang tepat untuk mencegah kecelakaan. Proyek ini menggunakan model CNN untuk memprediksi apakah seseorang merasa meKantuk atau tidak berdasarkan apakah mata tertutup atau terbuka dan mulut menguap atau tidak. Tujuan utama proyek ini adalah membatasi jumlah parameter model CNN yang dapat dilatih hingga di bawah 250 ribu sehingga sistem dapat digunakan di perangkat edge atau komputasi yang kurang efisien. Proyek

ini bisa menjadi aplikasi langsung dalam industri otomotif, membuat berkendara lebih aman, dan mengurangi jumlah kematian akibat mengemudi dalam keadaan meKantuk. Dengan adanya deteksi wajah kantuk maka diharapkan akan mampu mengurangi tingkat kecelakaan yang sering terjadi akibat faktor kelalaian manusia dan dapat membantu mengawasi pengemudi mobil agar lebih waspada dengan kondisi yang ada [5]. Berdasarkan permasalahan tersebut, maka penulis membuat suatu sistem "Deteksi MeKantuk Pada Pengemudi Mobil Menggunakan *Algoritma Convolutional Neural Network (CNN)*".

II. STUDI LITERATUR

A. Related Works

Banyak sekali metode deteksi kantuk yang dapat dilakukan berdasarkan ekstraksi fitur mata seperti PERCLOS, penelitian yang dilakukan oleh Junaedi dan Akbar di tahun 2018. Selain mata dan mulut, fitur wajah lain seperti alis dan raut wajah juga dapat digunakan untuk mendeteksi kantuk pada pengemudi kendaraan. Namun dengan kondisi background pengemudi yang sangat variatif maka fitur yang dibutuhkan juga akan berbeda-beda. Namun saat ini metode ekstraksi fitur manual sudah mulai ditinggalkan dan beralih ke metode Convolutional Neural Network (CNN) yang otomatis. Artinya fitur dapat dilatih secara otomatis didalam lapisan konvolusi deep learning. Pada penelitian (Jabbar et al., 2020) telah mengembangkan CNN berbasis Facial Landmark Detection yang dirancang untuk perangkat Android. Akurasi yang dihasilkan dapat mencapai 83%. Namun model ini akan mengalami kesulitan jika pencahayaan lingkungan kurang baik dimana fitur wajah sulit didapatkan. Pada penelitian ini kami tertarik untuk melihat bagaimana performa CNN untuk citra mentah tanpa perlu memodelkan karakteristik-karakteristik tersebut secara khusus. Artinya bagaimana caranya mendeteksi pengemudi yang sedang meKantuk menggunakan fitur yang dapat ditemukan secara otomatis menggunakan CNN.

B. Convolutional Neural Network (CNN)

CNN merupakan salah satu jenis model *deep learning* untuk mengolah data yang memiliki pola grid, dan dirancang untuk secara otomatis dan adaptif mempelajari hierarki spasial fitur, dari rendah - ke pola tingkat tinggi [5]. CNN adalah konstruksi matematis yang biasanya terdiri dari tiga jenis lapisan: konvolusi, penyatuan, dan lapisan yang sepenuhnya terhubung. Dua yang pertama, lapisan konvolusi dan penyatuan, melakukan ekstraksi fitur, sedangkan yang ketiga, lapisan yang sepenuhnya terhubung, memetakan fitur yang diekstraksi ke dalam hasil akhir, seperti klasifikasi. Lapisan konvolusi memainkan peran kunci dalam CNN, yang terdiri dari tumpukan operasi matematika, seperti konvolusi, jenis khusus dari operasi linier. Dalam gambar digital, nilai piksel disimpan dalam bentuk dua dimensi (2D), dan kisi kecil parameter yang disebut kernel, ekstraktor fitur yang dapat dioptimalkan, diterapkan pada setiap posisi gambar, yang membuat CNN sangat efisien untuk pemrosesan gambar, karena fitur dapat muncul di mana saja dalam gambar. Saat satu lapisan memasukkan keluarannya ke lapisan berikutnya,

fitur yang diekstraksi dapat secara hierarkis dan progresif menjadi lebih kompleks. Proses pengoptimalan parameter seperti kernel disebut pelatihan, yang dilakukan untuk meminimalkan perbedaan antara keluaran dan label kebenaran dasar melalui algoritma pengoptimalan yang disebut backpropagation dan penurunan gradien.

III. METODOLOGI PENELITIAN

A. Dataset

Pada penelitian ini kami menggunakan data yang bersumber dari Kaggle [6]. Data terdiri dari 2320 data *training* dan dibagi menjadi 4 kelas masing – masing. Data *training* mata terbuka sebanyak 580, mata tertutup 580, Kantuk 580, dan tidak Kantuk 580. Untuk dataset pengujian atau data *testing* kami menggunakan 30 data citra dan wajah kami sendiri dengan mengambil foto dari salah satu anggota kami serta kami menambahkan validation data sebanyak 580 dan dibagi masing-masing 145 per kelas. Dengan keterangan data yang kami gunakan seperti ditunjukkan pada tabel berikut ini.

Dataset	Jumlah Gambar Mata Tertutup	Jumlah Gambar Mata Terbuka	Jumlah Gambar Mengantuk	Jumlah Gambar Tidak Mengantuk
Training Dataset	580	718	580	580
Validation Data	145	145	145	145
Testing Dataset	8	8	6	8

Tabel 1. Rincian Jumlah Dataset

Dari tabel tersebut dapat dilihat pada bagian *Training* dataset maupun *Testing* dataset yang kami buat hampir seimbang agar mendapatkan tingkat akurasi yang tidak jauh berbeda antar kelasnya.

B. Augmentasi Data

Pada penelitian ini kami melakukan dua kali latihan data. Yang pertama kami menggunakan dataset yang mengubah ukuran gambar menjadi 200x200 dan yang kedua kami melakukan pelatihan model menggunakan dataset yang diaugmentasikan. Augmentasi dilakukan dengan harapan untuk menghindari *overfitting* dan untuk meningkatkan kemampuan dari arsitektur CNN yang kami buat. Teknik augmentasi data yang digunakan seperti rescaling, flipping, dan zooming. Penggunaan jenis augmentasi data ini memastikan bahwa CNN kami melihat variasi data baru di setiap epoch selama proses pelatihan (*Training*). Berbagai teknik augmentasi data yang digunakan dalam pekerjaan ini tercantum dalam tabel berikut.

Teknik pada Augmentasi Data	Nilai (Value)
Rescale	200x200
Random Flip (Horizontal)	True
Random Zoom	0.1

Tabel 2. Teknik Augmentasi Data

Berikut adalah contoh tampilan salah satu gambar dari dataset *training* yang kami gunakan untuk augmentasi.



Gambar 1. Dataset Setelah Augmentasi Random Flip Horizontal dan Zoom.

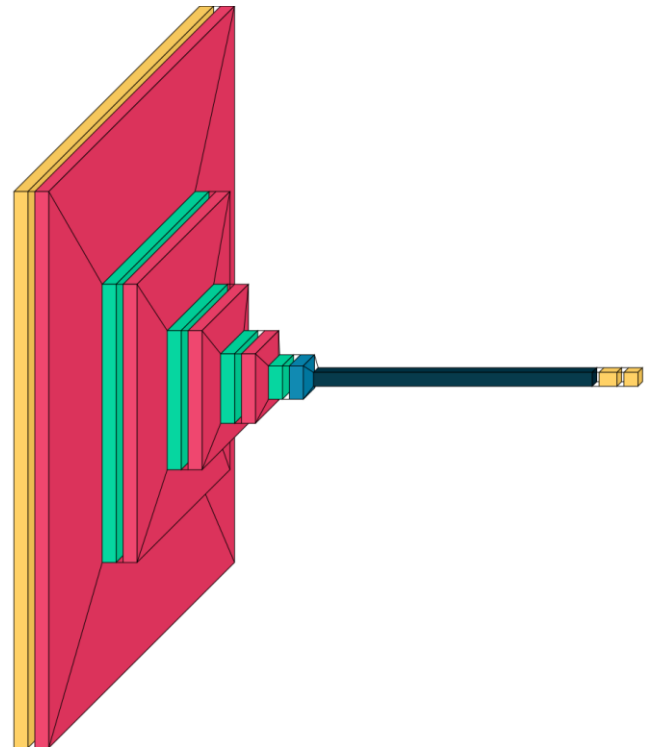
C. Arsitektur

Pada penelitian ini, kami merancang model CNN untuk mendeteksi kantuk pada pengemudi mobil dengan tujuan untuk menguji kelayakan sopir dalam mengendarai mobil dengan harapan mengurangi kemungkinan terjadinya kecelakaan.

Kami memilih untuk mengklasifikasi 4 jenis citra yaitu: mata tertutup, mata terbuka, tidak menguap dan menguap. Berdasarkan hal tersebut, kami merancang arsitektur *neural network* ini dengan menggunakan 4 buah *convolutional layers*.

Layer (type)	Output Shape	Param #
rescaling (Rescaling)	(None, 200, 200, 3)	0
conv2d (Conv2D)	(None, 200, 200, 16)	448
max_pooling2d (MaxPooling2D)	(None, 100, 100, 16)	0
conv2d_1 (Conv2D)	(None, 100, 100, 32)	4640
max_pooling2d_1 (MaxPooling2D)	(None, 50, 50, 32)	0
conv2d_2 (Conv2D)	(None, 50, 50, 64)	18496
max_pooling2d_2 (MaxPooling2D)	(None, 25, 25, 64)	0
conv2d_3 (Conv2D)	(None, 25, 25, 128)	73856
max_pooling2d_3 (MaxPooling2D)	(None, 12, 12, 128)	0
dropout (Dropout)	(None, 12, 12, 128)	0
flatten (Flatten)	(None, 18432)	0
dense (Dense)	(None, 256)	4718848
dense_1 (Dense)	(None, 4)	1028

Gambar 2. Ringkasan Arsitektur CNN yang dirancang



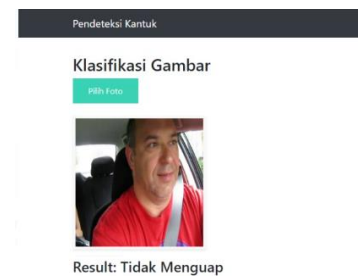
Gambar 3. Visualisasi Arsitektur CNN yang dirancang

Convolutional layers yang pertama memiliki 16 filters, yang kedua 32 filters, yang ketiga 64 filters, dan layer terakhir memiliki 128 filters. Seluruh filter tersebut memiliki size sama yaitu 3x3. Banyaknya jumlah filter yang ada di tiap layer menentukan berapa banyak features yang dapat diekstrak oleh network. Kami menggunakan layer *dropout* untuk memperkecil *overfitting* dari model yang dilatih sehingga dapat meningkatkan model yang dibuat.

Pooling yang diterapkan di tiap layer bertujuan untuk mengurangi dimensi dari *features* yang sudah diekstrak dari gambar. Kami melakukan proses *training* dengan 2900 citra, kemudian dites dengan 30 citra.

D. Tampilan Website

Pada tampilan website kami menggunakan *background* yang tidak terlalu rumit. Tampilan hanya menyediakan tempat untuk memasukkan *image* dan keterangan bahwa image yang telah dimasukkan tersebut masuk ke dalam kelas yang mana seperti gambar berikut.



Gambar 4. Tampilan website

Kami membuat tampilan sederhana agar dapat memudahkan pengguna. Website dibuat menggunakan HTML dengan menggunakan text editor *Visual Studio Code*.

IV. HASIL DAN PEMBAHASAN

Pelatihan (*Training*) dilakukan pada dataset yang sudah kami sediakan sebelumnya, dan kami melakukan pelatihan terhadap 4 model Pure CNN. Dalam melakukan pelatihan modelnya kami mencoba membuat dua jenis, satu menggunakan (*image augmentation*) dan juga satunya tidak menggunakan *image augmentation*.

A. Hasil pelatihan tanpa menggunakan Augmentation

Dari pelatihan model dengan tidak menggunakan *image augmentation* kami mendapatkan nilai akurasi *training* dan *validation* menyentuh nilai 99% dengan 50 epoch. Sedangkan untuk nilai *training loss*nya mencapai 0.01% dan *validation loss*nya mencapai nilai 0.2% pada epoch ke-39.



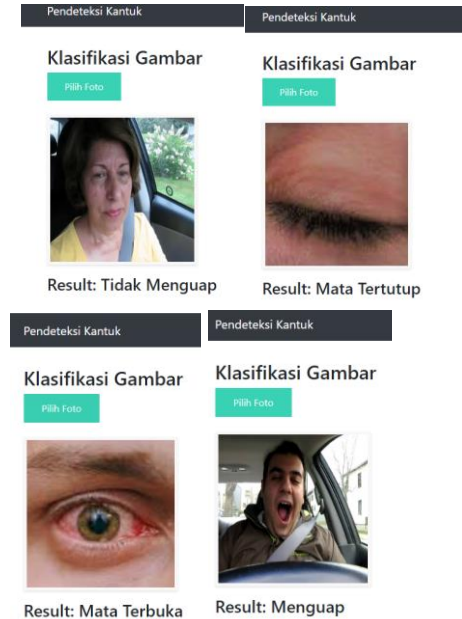
Gambar 5. Nilai hasil training model tanpa Image Augmentation

Dengan jumlah data *testing* sebanyak 30 foto, didapatkan nilai presisi, recall dan f1-score bernilai 1 yang berarti 100%.

	precision	recall	f1-score	support
Closed	1.00	1.00	1.00	8
Open	1.00	1.00	1.00	8
no_yawn	1.00	1.00	1.00	8
yawn	1.00	1.00	1.00	6
accuracy			1.00	30
macro avg	1.00	1.00	1.00	30
weighted avg	1.00	1.00	1.00	30

Gambar 6. Nilai *evaluation metrics* model tanpa Image Augmentation

Pada *website* yang dibuat dengan menggunakan flask, model mampu untuk memprediksi keseluruhan data test dengan tepat.



Gambar7. Performa model tanpa Image Augmentation

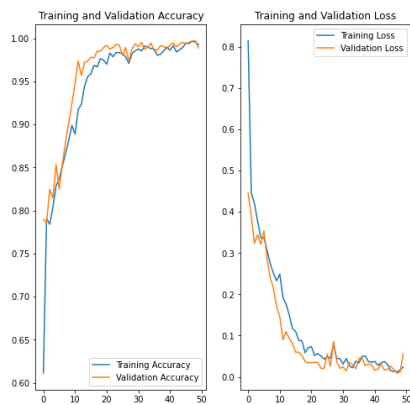
Namun untuk perfora model dalam penggunaan secara *real time* menunjukkan hasil yang kurang memuaskan dikarenakan perbedaan citra dan jenis data yang terdapat pada data uji dan data yang dilatih sangat berbeda.



Gambar 8. Performa model tanpa Image Augmentation secara *real time*

B. Hasil pelatihan menggunakan Augmentation

Pada model yang dilatih dengan menggunakan *image augmentation* kami mendapatkan nilai akurasi *training* dan *validation* yang tidak kalah nilainya dengan nilai model tanpa *image augmentation* yaitu mencapai nilai 99% dengan epoch sebanyak 50. Untuk nilai *training loss* mencapai nilai 1% dan *validation loss* mencapai 0.09% pada epoch ke-48.



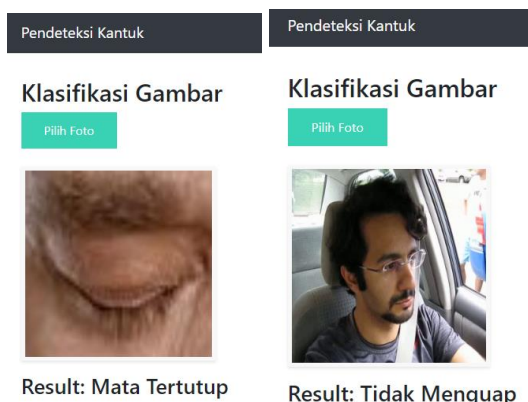
Gambar 9. Nilai hasil training model dengan Image Augmentation

Untuk nilai *evaluation metrics* yang didapatkan dari model yang dilatih dengan *image augmentation* nilainya sama dengan model yang dilatih tanpa *image augmentation* yaitu untuk data test berjumlah 30 didapatkan nilai presisi, *recall* dan *f1-score* bernilai 1.

	precision	recall	f1-score	support
Closed	1.00	1.00	1.00	8
Open	1.00	1.00	1.00	8
no_yawn	1.00	1.00	1.00	8
yawn	1.00	1.00	1.00	6
accuracy			1.00	30
macro avg	1.00	1.00	1.00	30
weighted avg	1.00	1.00	1.00	30

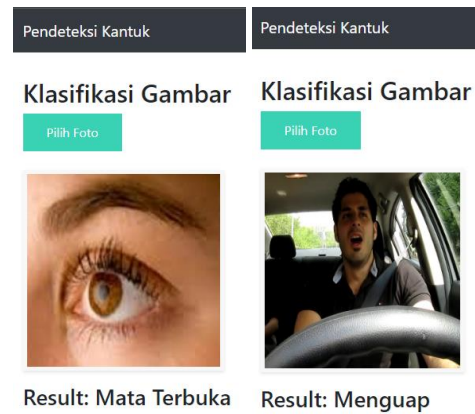
Gambar 10. Nilai *evaluation metrics* model dengan Image Augmentation

Pada *website* yang dibangun, model yang dibuat dengan menggunakan *image augmentation* tidak kalah dengan performa model yang dibuat tanpa *image augmentation*.



Result: Mata Tertutup

Result: Tidak Menguap



Result: Mata Terbuka

Result: Menguap

Gambar 11. Performa model dengan Image Augmentation

Performa model yang dibangun dengan *image augmentation* memberikan hasil yang lebih baik dibandingkan dengan model tanpa *image augmentation*.



Gambar 12. Performa model dengan Image Augmentation secara real time

Hal ini menunjukkan bahwa model yang dibangun dengan augmentasi data dan tidak menunjukkan tidak adanya perbedaan secara signifikan pada saat memprediksi data namun terdapat perbedaan ketika prediksi data secara real time dimana model yang dibangun dengan *image augmentation* memberikan hasil yang lebih baik dibandingkan dengan tanpa *image augmentation*.

V. KESIMPULAN

Pada penelitian ini, kami membuat sebuah model untuk mendeteksi kantuk pada supir saat mengendarai mobil agar dapat meminimalisir terjadinya kecelakaan. Tujuan utama penelitian ini adalah dengan menjaga dan memberi peringatan kepada supir apabila model mendeteksi indikasi kantuk pada supir tersebut. Hasil dari pengujian model memperlihatkan bahwa tingkat akurasi yang dimiliki oleh model yang dibuat menggunakan *image augmentation* maupun tidak, berada pada ambang akurasi sebesar 99%. dengan demikian kami yakin bahwa CNN model yang kami usung memiliki potensi untuk terus dikembangkan menjadi lebih baik dengan jumlah dataset yang lebih banyak.

UCAPAN TERIMA KASIH

Penulis ingin mengucapkan terima kasih kepada Dr.Eng. Suci Dwijayanti, S.T, M.S sebagai dosen pembimbing dan penulis juga berterima kasih kepada Angel Metanosa Afinda, Abdiel Willyar Goni, Muhammad Andrian Maulana, sebagai *Homeroom Coach* kami di Orbit Future Akademi.

DAFTAR PUSTAKA

- [1] N. C. S. Dewi and I. N. Budiantara, “Faktor-Faktor Yang Mempengaruhi Kecelakaan di Provinsi Jawa Timur Menggunakan Regresi Nonparametrik Spline Truncated,” vol. 7, no. 2, 2018.
- [2] H. Fitrianti, “MODEL FAKTOR-FAKTOR BANYAKNYA KECELAKAAN LALU LINTAS PADA KENDARAAN SEPEDA MOTOR DENGAN PENDEKATAN GENERALIZED LINEAR MODEL,” vol. 4, pp. 94–104, 2017.
- [3] S. D. Andryanto, “Bahaya Microsleep Saat Mengemudi, Penyumbang Utama Kecelakaan di Jalanan,” 2021. <https://otomotif.tempo.co/read/1521242/bahaya-microsleep-saat-mengemudi-penyumbang-utama-kecelakaan-di-jalanan> (accessed Jun. 19, 2022).
- [4] A. Y. Rahman, S. Sumpeno, and M. H. Purnomo, “Arca detection and matching using Scale Invariant Feature Transform (SIFT) method of stereo camera,” vol. 2018-Janua, no. July 2018, pp. 66–71, 2018.
- [5] Minaee, S., Kafieh, R., Sonka, M., Yazdani, S., & Jamalipour Soufi, G. (2020). Deep-COVID: Predicting COVID-19 from chest X-ray images using deep transfer learning. *Medical Image Analysis*, 65.
- [6] D. Perumandla, “Drowsiness_dataset,” 2020. <https://www.kaggle.com/datasets/dheerajperumandla/drowsiness-dataset/code> (accessed Jun. 17, 2022).