

DETEKSI MICROSLEEP MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK DENGAN SENSOR ESP32 PADA PENGEMUDI

MICROSLEEP DETECTION USING CONVOLUTIONAL NEURAL NETWORK WITH ESP32 SENSOR ON THE DRIVER

Yuma Akbar¹, Bayu Bintoro², Dadang Iskandar Mulyana³, Sri Lestari⁴

^{1,2,3,4}Sekolah Tinggi Ilmu Komputer Cipta Karya Informatika

yuma.pjj@gmail.com

ABSTRACT

Microsleep is a dangerous condition that occurs when a person falls asleep for a very short time without realizing it, especially while driving. This thesis aims to develop a Convolutional Neural Network (CNN)-based Microsleep detection system with ESP32 sensors in drivers. The system is designed to monitor and analyze the driver's activity patterns directly collected by the ESP32 sensor. The data obtained is then processed and classified using a CNN model to detect Microsleep events. The implementation of this system is expected to improve driving safety, thereby reducing the risk of traffic accidents. Further research is needed to improve detection accuracy and test the system on real conditions on the highway. Based on the results of data processing and discussion in this study, it can be shown that the convolutional neural network (CNN) architecture with the you only look once (YOLO) algorithm can be implemented to detect whether the driver is awake or asleep. In addition, by configuring the model parameters with batch size 16, network size 320x320, the IoU value is obtained with a learning rate of 0.01.

Keyword: Internet Of Thing (Iot), Air Quality Monitoring, Arduino.

ABSTRAK

Microsleep merupakan kondisi berbahaya yang terjadi ketika seseorang tertidur dalam waktu sangat singkat tanpa disadari, terutama saat mengemudi. Skripsi ini bertujuan mengembangkan sistem deteksi Microsleep berbasis Convolutional Neural Network (CNN) dengan sensor ESP32 pada pengemudi. Sistem ini dirancang untuk memantau dan menganalisis pola aktivitas pengemudi secara langsung yang dikumpulkan oleh sensor ESP32. Data yang diperoleh kemudian diproses dan diklasifikasikan menggunakan model CNN untuk mendeteksi kejadian Microsleep. Implementasi sistem ini diharapkan dapat meningkatkan keselamatan berkendara, sehingga dapat mengurangi risiko kecelakaan lalu lintas. Penelitian lebih lanjut diperlukan untuk meningkatkan akurasi deteksi dan menguji sistem pada kondisi nyata di jalan raya. Hasil pengolahan data dan hasil pembahasan pada penelitian ini, dapat ditunjukkan bahwa arsitektur convolutional neural network (CNN) dengan algoritma you only look once (YOLO) dapat diimplementasikan untuk melakukan deteksi pada pengemudi apakah terjaga atau tertidur. Selain itu dengan melakukan konfigurasi parameter model dengan batch size 16, network size 320x320, diperoleh nilai IoU dengan learning rate 0.01.

Kata Kunci: Internet of thing (IoT), monitoring kualitas udara, arduino

PENDAHULUAN

Microsleep adalah kondisi tidur singkat yang berlangsung selama 1-15 detik yang dapat menyebabkan situasi berbahaya dengan potensi yang berakibat fatal[1], *microsleep* dapat menjadi ancaman serius dalam keselamatan berkendara bagi pengemudi, penumpang, dan pengguna jalan lain. Penelitian sebelumnya mengindikasikan bahwa faktor-faktor seperti mengantuk dapat mengurangi konsentrasi pengemudi dan meningkatkan risiko kecelakaan lalu

lintas[2]. Selain itu, kecelakaan lalu lintas dapat disebabkan oleh berbagai faktor, termasuk kondisi fisik dan mental pengemudi, seperti mengantuk dan kelelahan[3].

Data terbaru kecelakaan transportasi darat di Indonesia menunjukkan peningkatan yang perlu mendapat perhatian serius. Menurut data dari Badan Pusat Statistik (BPS), kecelakaan lalu lintas di Indonesia pada tahun 2021 mengalami peningkatan sebanyak 116.411 kejadian atau sekitar 3,62% dari tahun

sebelumnya, yaitu 100.028 kejadian pada tahun 2020[4]. Faktor – faktor yang mempengaruhi kecelakaan lalu lintas angkutan jalan berdasarkan hasil investigasi KNKT antara lain adalah faktor manusia, sarana, prasarana, dan lingkungan. Total faktor penyebab kecelakaan LLAJ yang diinvestigasi oleh KNKT dari tahun 2019 – 2023 yaitu sebanyak 49 faktor penyebab. Faktor manusia merupakan faktor penyebab kecelakaan yang paling dominan dari tahun 2019 – 2023 yaitu sebanyak 28 kecelakaan[5]. Angka-angka ini menunjukkan urgensi untuk mengembangkan solusi yang efektif dalam mendeteksi dan mencegah *microsleep* pada pengemudi.

Mendeteksi *microsleep* pada pengemudi sangat penting untuk mencegah kecelakaan yang disebabkan oleh rasa kantuk pengemudi. Penelitian terbaru mengusulkan penggunaan pendekatan perilaku dengan *Convolutional Neural Network* (CNN) untuk mendeteksi rasa kantuk pada pengemudi melalui pengenalan kondisi mata[6], [7], [8]. Pendekatan ini melibatkan pemrosesan gambar wajah pengemudi untuk menentukan apakah mereka mengantuk atau terjaga, mencapai tingkat akurasi yang tinggi hingga 99,93% dengan model CNN. Mengintegrasikan sensor ESP32 dengan model CNN dapat meningkatkan kemampuan deteksi *microsleep* secara real-time, memberikan solusi yang tidak mengganggu dan efektif untuk memperingatkan pengemudi dan mencegah potensi kecelakaan di jalan.

Oleh karena itu, penelitian ini bertujuan untuk mengembangkan sistem deteksi *microsleep* menggunakan CNN dengan sensor ESP32 pada pengemudi. Sistem ini diharapkan dapat memberikan peringatan dini kepada pengemudi ketika terdeteksi tanda-tanda *microsleep*, sehingga dapat mengambil tindakan pencegahan, seperti beristirahat atau menepi sejenak, sebelum terjadi kecelakaan. Dengan demikian, penelitian

ini diharapkan dapat berkontribusi dalam meningkatkan keselamatan.

TINJAUAN PUSTAKA

Microsleep

Microsleep adalah episode tidur singkat yang berlangsung selama beberapa detik dan dapat disebabkan oleh faktor-faktor seperti kurang tidur, sleep apnea, dan narkolepsi. Hal ini ditandai dengan fragmen tidur yang pendek, biasanya berlangsung kurang dari 15 detik, dan dapat menyebabkan situasi berbahaya dengan hasil yang berpotensi fatal. Terpapar pada pekerjaan monoton dalam waktu yang lama, seperti mengemudi, dapat menyebabkan episode microsleep karena tingkat kantuk yang tinggi. Microsleep dapat menyebabkan hilangnya kesadaran, berdampak pada kinerja tugas, menyebabkan waktu respons yang melambat, kurangnya kesadaran yang terputus-putus, dan kedipan yang berlangsung lebih dari 500 milidetik. Sangat penting untuk mendeteksi dan mencegah microsleep, terutama dalam kegiatan seperti mengemudi, untuk menghindari kecelakaan dan memastikan keselamatan.

Machine Learning

Machine Learning melibatkan pemanfaatan algoritma kecerdasan buatan yang memungkinkan sistem untuk belajar dan berkembang dari pengalaman tanpa pemrograman secara eksplisit [29]. Hal ini telah diterapkan di berbagai bidang seperti kesehatan, keuangan, dan keamanan jaringan, dengan akurasi prediksi mencapai hingga 91% dalam bidang tertentu. Teknik Machine Learning, termasuk metode *supervised and unsupervised learning*, telah memainkan peran penting dalam meningkatkan proses pengambilan keputusan, mengoptimalkan alur kerja, dan memajukan penelitian di berbagai bidang seperti onkologi radiasi dan pencitraan biomedis. Integrasi machine learning dengan teknologi seperti deep learning telah secara signifikan

meningkatkan efektivitasnya, terutama dalam bidang yang kompleks seperti ilmu hidrologi dan keamanan siber. Standarisasi alat dan upaya kolaboratif sangat penting untuk sepenuhnya memanfaatkan potensi pembelajaran mesin di berbagai disiplin ilmu.

Computer Vision

Computer Vision adalah bidang kecerdasan buatan yang berfokus pada pengembangan algoritma dan sistem untuk memungkinkan komputer menafsirkan dan memahami informasi visual dari dunia nyata. Hal ini mencakup tugas-tugas seperti pengenalan gambar, dan deteksi objek. Teknik computer vision memanfaatkan berbagai teknologi seperti jaringan pemrosesan gambar, jaringan saraf tiruan, dan metode pembelajaran mendalam untuk menganalisis dan mengekstrak wawasan yang bermakna dari data visual. Kemajuan algoritma computer vision, terutama dalam hubungannya dengan pendekatan pembelajaran mesin, telah secara signifikan meningkatkan kemampuan sistem untuk memproses dan menginterpretasikan data visual secara akurat dan efisien. Integrasi computer vision dengan teknologi lain seperti komputasi tepi semakin memperluas potensi aplikasinya, memungkinkan pemrosesan waktu nyata dan pengambilan keputusan di tepi jaringan berbagai industri dengan memberikan solusi inovatif untuk tugas-tugas yang membutuhkan persepsi dan pemahaman visual.

Object Detection

Object Detection atau deteksi objek adalah teknik computer vision untuk menemukan lokasi objek dalam gambar atau video. Algoritma deteksi objek biasanya memanfaatkan pembelajaran mesin atau pembelajaran mendalam untuk memberikan hasil yang bermakna. Saat manusia melihat gambar atau video, kita dapat mengenali dan menemukan lokasi

objek yang diinginkan dalam hitungan singkat. Tujuan dari deteksi objek adalah untuk mereplikasi kecerdasan ini menggunakan komputer.

Convolutional Neural Network

Convolutional Neural Network (CNN) adalah salah satu jenis neural network yang biasa digunakan pada data image. CNN bisa digunakan untuk mendeteksi dan mengenali object pada sebuah image. CNN memanfaatkan proses konvolusi dengan menggerakkan sebuah kernel konvolusi (filter) berukuran tertentu ke sebuah gambar, komputer mendapatkan informasi representatif baru dari hasil perkalian bagian.

ESP32-CAM

Modul ESP32-CAM merupakan modul kamera yang dilengkapi dengan WiFi, Bluetooth dan GPIO yang dapat berfungsi sebagai input atau output. Ukurannya yang kecil sangat kompetitif beroperasi secara mandiri dengan minimum sistem. Modul ESP32-CAM memiliki diameter 27 x 40,5 x 4,5 mm dan arus hingga 6 mA. ESP32-CAM termasuk salah satu module dari Espressif and AI-Thinker. Sedangkan development board ESP32-CAM merupakan salah satu produk pengembangan dari ESP32 yang sudah dilengkapi dengan module microSD dan modul camera. Untuk upload dan flash program bisa menggunakan Arduino IDE.

Development board ESP32-CAM termasuk mikrokontroler yang komplit karena memiliki WiFi 2.4 GHz dan Bluetooth. Selain itu, module ini sudah sangat lengkap karena terintegrasi dengan module camera OV2640 dengan resolusi 2MP yang berfungsi sebagai photo atau video dan microSD module untuk menyimpan data di microSD. Module ESP32-CAM memiliki built-in module SD Card, sehingga bisa menyimpan data-data gambar atau logger sensor ke dalam module tersebut dengan menyediakan microSD di pasang ke module tersebut.

Untuk memprogram atau mengupload program ke development board bisa menggunakan Arduino IDE dengan menggunakan alat tambahan yaitu module FTDI atau module serial to usb yang kompatibel, karena tidak memiliki built-in uploader.

FT232RL FTDI USB to TTL Serial Adapter

Modul FT232RL adalah papan sirkuit konverter mini USB ke serial UART TTL 5V 3.3V yang dirancang untuk komunikasi antara port USB komputer dan port pemrograman mikrokontroler. Modul ini memiliki jumper pada papan sirkuit yang memungkinkan papan dikonfigurasi untuk 3.3V atau 5V, baik untuk output daya dan level IO. Konverter UART serial ini dapat digunakan untuk meng-upgrade periferal lama ke USB.

Arduino IDE

Arduino IDE (Integrated Development Environment) adalah perangkat lunak open source yang digunakan untuk memprogram mikrokontroler Arduino. Arduino IDE memiliki antarmuka yang sederhana dan mudah digunakan, bahkan untuk pengguna yang belum berpengalaman dalam pemrograman. Arduino IDE menyediakan banyak library dan contoh program yang dapat digunakan untuk mempermudah pembuatan program.

Arduino IDE dapat digunakan untuk membuat sketch pemrograman, mengedit, meng-upload ke board yang ditentukan, dan meng-coding program tertentu. Arduino IDE dapat berjalan pada Windows, Mac OS X, dan Linux.

Visual Studio Code

Visual Studio Code (VSCode) adalah sebuah editor kode sumber yang dibuat oleh Microsoft untuk sistem operasi Windows, macOS, dan Linux. VSCode adalah sebuah aplikasi yang dapat digunakan untuk membuat kode dalam

berbagai bahasa pemrograman, seperti JavaScript, TypeScript, Node.js, C++, C#, Java, Python, PHP, Go, dan .NET. VSCode memiliki banyak fitur, seperti penyorotan sintaksis, penyelesaian kode, kutipan kode, merefaktor kode, pengawakutuan, dan Git. VSCode juga memiliki ekosistem ekstensi yang luas yang dapat mendukung bahasa pemrograman dan runtime lainnya.

Label Studio

Label Studio adalah alat pelabelan data sumber terbuka (open source) yang mendukung banyak proyek, pengguna, dan tipe data dalam satu platform.

YOLOv5

YOLOv5 adalah model dalam keluarga model visi komputer You Only Look Once (YOLO). YOLOv5 umumnya digunakan untuk mendeteksi objek. YOLOv5 hadir dalam empat versi utama: kecil (s), sedang (m), besar (l), dan ekstra besar (x), yang masing-masing menawarkan tingkat akurasi yang semakin tinggi. Setiap varian juga memerlukan waktu pelatihan yang berbeda.

Deteksi objek, kasus penggunaan yang menjadi tujuan perancangan YOLOv5, melibatkan pembuatan fitur dari gambar input. Fitur-fitur ini kemudian dimasukkan melalui sistem prediksi untuk menggambar kotak di sekitar objek dan memprediksi kelasnya.

Python

Python adalah bahasa pemrograman tingkat tinggi yang bersifat interpretatif, interaktif, dan berorientasi objek. Diciptakan oleh Guido van Rossum pada tahun 1991, Python dirancang dengan filosofi yang menekankan keterbacaan kode dan sintaksis yang jelas, sehingga mudah dipelajari dan digunakan bahkan oleh pemula sekalipun.

METODE

Metode penelitian yang digunakan berupa Metode eksperimen, Metode

eksperimen adalah metode penelitian kuantitatif. Pada penelitian ini peneliti menggunakan metode eksperimen yaitu dengan cara mengambil data berulang kali sampai mendapatkan hasil yang diinginkan.

Tahapan penelitian dimulai dari pengumpulan data dan informasi, analisis kebutuhan dan perancangan, perancangan *software* dan *hardware*, pengujian alat, dan implementasi sistem.

Tahap ini dimulai dengan pre-processing, yang mencakup pemberian label pada dataset dan pembagian dataset menjadi data latih dan data uji. Setelah pre-processing selesai, dilakukan konfigurasi pada Google Colaboratory dan konfigurasi parameter model untuk memulai proses pelatihan (training) dan pengujian (testing) data.

Pembahasan hasil dapat dimulai setelah model yang telah dibuat mampu membedakan antara pengemudi yang tertidur (sleep) dan terjaga (awake) saat dilakukan pengujian dengan memasukkan gambar ke dalam model. Jika model belum mampu melakukan deteksi, maka tahap pre-processing data perlu diulang. Namun, jika model berhasil mendeteksi namun akurasi di bawah 80%, maka konfigurasi parameter model perlu disesuaikan. Pembahasan hasil sesuai dengan pengolahan data yang telah dilakukan baru akan dilaksanakan jika akurasi model telah mencapai lebih dari 80%.

Tahap akhir penelitian ini diakhiri dengan merangkum kesimpulan dari seluruh proses yang telah dilakukan, dengan fokus menjawab rumusan masalah yang telah ditetapkan sebelumnya. Selain itu, saran-saran untuk penelitian selanjutnya juga diberikan sebagai referensi bagi pengembangan studi di masa yang akan datang.

HASIL DAN PEMBAHASAN

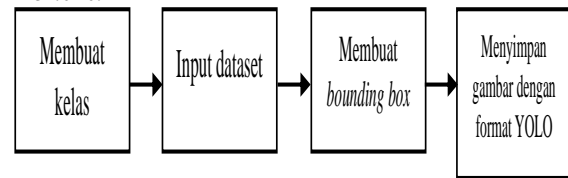
Hasil Pengujian YOLOv5

Penelitian kali ini menggunakan YOLOv5 untuk mendeteksi wajah

pengemudi. Selanjutnya, Convolutional Neural Network (CNN) yang telah dilatih dengan gambar mata terbuka dan tertutup digunakan untuk menentukan kondisi pengemudi, apakah terbuka atau tertutup. Sistem akan menampilkan status mata pengemudi (terjaga atau tertidur).

Pada tahap pre-processing data dilakukan proses pelabelan dataset dan split dataset. Pelabelan dataset digunakan untuk memberi label pada dataset yang sudah dikumpulkan, sedangkan split dataset digunakan untuk melakukan pembagian persentase data training dan data testing.

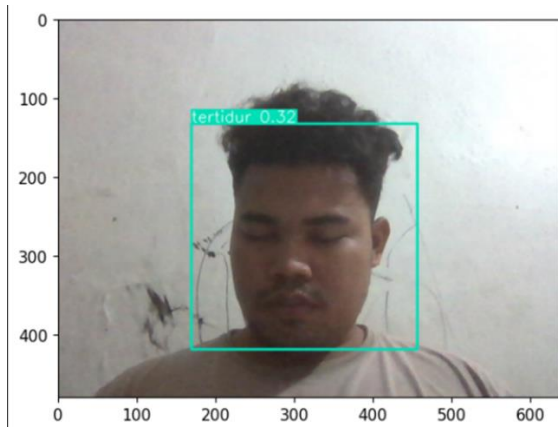
Pada tahap ini dataset diberi label terjaga atau tertidur menggunakan aplikasi Label Studio. Output dari pelabelan ini adalah gambar yang terlabeli dengan format YOLO dan tersimpan dalam tipe file .txt.



Gambar 1. Diagram Proses Pelabelan Dataset

Testing Dataset

Pengujian dilakukan menggunakan sumber dataset gambar yang berasal dari data yang dikumpulkan menggunakan VSCode. Pengujian dilakukan dalam bentuk gambar yang di input ke dalam model. Hasil deteksi menunjukkan akurasi 32% tertidur dengan kecepatan prediksi 13.8ms seperti pada gambar 1. Nilai akurasi menunjukkan tingkat kepercayaan (confidence) sistem dalam melakukan deteksi.



Gambar 2. Hasil Testing pada gambar tertidur

Kemudian pada pengujian gambar yang kedua digunakan gambar dengan mata terbuka yang menjadi salah satu ciri terjaga. Hasil deteksi menunjukkan akurasi 42% terjaga dengan kecepatan prediksi 7.1ms seperti pada gambar 3.



Gambar 3. Hasil Testing pada gambar terjaga

Pembahasan

Berdasarkan sistem deteksi kantuk pada pengemudi yang sudah dirancang, diperoleh hasil algoritma deteksi objek you only look once (YOLO)

dapat diimplementasikan untuk melakukan deteksi pada pengemudi dengan label terjaga dan tertidur.

Proses mengumpulkan dan menandai data (pelabelan) merupakan langkah penting dalam membangun sistem deteksi objek menggunakan YOLO. Semakin banyak gambar yang digunakan untuk melatih model, semakin baik kemampuan model dalam mengidentifikasi objek dengan benar, terutama dalam berbagai pose dan

pencahayaannya saat pengujian. Hal ini karena model akan belajar dari variasi data yang luas.

Ada beberapa hal yang menyebabkan nilai akurasi pada penelitian ini tidak terlalu tinggi dan juga kekeliruan dalam melakukan prediksi. Dataset yang digunakan saat proses pelatihan pada penelitian ini menggunakan gambar yang general yaitu hanya menggunakan gambar wajah untuk area deteksi, tidak menggunakan dataset dengan area yang lebih spesifik seperti dataset mata tertutup dan dataset mulut yang terbuka sebagai indikasi pengemudi sedang mengantuk, hal ini menyebabkan keterbatasan sistem dalam mengenali ragam ekspresi pengemudi. Pada penelitian ini hanya menggunakan 100 dataset gambar pengemudi yang terjaga dan tertidur.

Hal lain yang dapat menjadi penyebab akurasi yang tidak tinggi adalah terjadinya human error saat melakukan proses labelling karena dilakukan secara manual. Contoh human error yang dimaksud terjadi ketika membuat bounding box untuk setiap gambar, ada bounding box yang dibuat terlalu lebar melebihi bentuk wajah dan terdapat bounding box yang dibuat terlalu pas sehingga tidak semua wajah terlabeli dengan baik.

Sistem deteksi kantuk ini diharapkan dapat dikembangkan menjadi sistem peringatan dini saat mengemudi. Dengan begitu, potensi kecelakaan akibat kantuk dapat diminimalisir. Namun, sistem ini masih perlu perbaikan karena akurasi belum optimal.

PENUTUP Kesimpulan

Berdasarkan penelitian dan penerapan algoritma Convolutional Neural Network menggunakan model YOLOv5 dengan sensor ESP32 pada pengemudi dapat disimpulkan:

1. Penelitian yang telah dilakukan mengenai deteksi microsleep menggunakan CNN dari 100 dataset

dengan dua kelas yaitu terjaga dan tertidur.

2. Berdasarkan dari hasil pengolahan data dan hasil pembahasan pada penelitian ini, dapat ditunjukkan bahwa arsitektur convolutional neural network (CNN) dengan algoritma you only look once (YOLO) dapat diimplementasikan untuk melakukan deteksi pada pengemudi apakah terjaga atau tertidur. Selain itu dengan melakukan konfigurasi parameter model dengan batch size 16, network size 320x320, diperoleh nilai IoU dengan learning rate 0.01.

Saran untuk pengembangan alat ini :

Adapun saran-saran yang disampaikan berdasarkan hasil penelitian dan analisa selama melakukan penelitian ini adalah:

1. Untuk penelitian selanjutnya dapat mengembangkan penelitian ini dan menghubungkan dengan metode lain sehingga dapat menyentuh akurasi yang lebih bagus lagi. Dari hasil penelitian ini belum lengkap dan efisien, diharapkan ada yang mengembangkan penelitian ini dengan menggunakan atau menambahkan metode lain agar dalam menentukan suatu objek lebih cepat dan akurat.
2. Dari penelitian ini, diharapkan untuk bisa menambahkan fitur alarm

DAFTAR PUSTAKA

- [1] J. Skorucak, A. Hertig-Godeschalk, P. Achermann, J. Mathis, dan D. Schreier, "Automatically Detected Microsleep Episodes in the Fitness-to-Drive Assessment," *Frontiers in Neuroscience*. 2020. doi: 10.3389/fnins.2020.00008.
- [2] N. A. Ramadhani, S. Aulia, E. Suhartono, dan S. Hadiyoso, "Deteksi Kantuk Pada Pengemudi Berdasarkan Penginderaan Wajah Menggunakan PCA Dan SVM," *Jurnal Rekayasa Elektrika*. 2021. doi: 10.17529/jre.v17i2.19884.
- [3] A. Afif Mauludi, Z. Djunaidi, dan L. Saiful Arif, "Perilaku Berisiko Sebagai Faktor Penyebab Kecelakaan Pada Pengemudi Sepeda Motor Komersial: Systematic Review," *Jurnal Keselamatan Transportasi Jalan (Indonesian Journal of Road Safety)*, vol. 8, no. 1, hlm. 12–25, Jun 2021, doi: 10.46447/ktj.v8i1.307.
- [4] K. Kendaraan dkk., "Creative Education of Research in Information Technology and Artificial Informatics".
- [5] S. Tjahjono, A. Priyanto, dan S. Bachri, *Buku Statistik Investigasi Kecelakaan Transportasi 2023*. Jakarta Pusat: Komite Nasional Keselamatan Transportasi, 2023.
- [6] S. Bhandarkar, T. Naxane, S. Shrungare, S. Rajhance, M. P. Deshmukh, dan T. Kute, "Neural Network Based Detection of Driver's Drowsiness," *CC-BY-SA*, vol. 4, no. 23, 2020, doi: 10.36227/techrxiv.14816010.v1.
- [7] S. K. B. Sangeetha, S. K. Mathivanan, V. Muthukumar, N. Pughazendi, P. Jayagopal, dan M. S. Uddin, "A Deep Learning Approach to Detect Microsleep Using Various Forms of EEG Signal," *Math Probl Eng*, vol. 2023, 2023, doi: 10.1155/2023/7317938.
- [8] Puli Suhas Reddy, Dr. Rama Chandra, D Bhavya Rishitha, dan K. Varunraj, "Detecting Driver Drowsiness and Buzzer Alert using CNN Algorithm," *International Journal of Advanced Research in Science, Communication and Technology*, hlm. 179–184, Mei 2023, doi: 10.48175/ijarsct-9676