

Perancangan Alat Deteksi Dini Kondisi Kantuk untuk Mengurangi Risiko Kecelakaan Kerja Berbasis Pengolahan Citra Digital

ADITIYA D.N¹, CAHYADI NUGRAHA², HENDRO PRASSETIYO³

Program Studi Teknik Industri, Fakultas Teknologi Industri, Institut Teknologi Nasional, Jl. PHH Mustofa No 23, Bandung, 40124, Indonesia
Email: aditiyadn@gmail.com

Received DD MM YYYY | *Revised* DD MM YYYY | *Accepted* DD MM YYYY

ABSTRAK

Kantuk adalah salah satu penyebab kecelakaan yang signifikan yang sering kali tidak disadari sampai akhirnya terjadi kesalahan dalam pengambilan keputusan. Telah banyak dirancang alat deteksi kantuk namun perlu penggunaan alat yang diletakkan pada faal tubuh pengguna dan penggunaan alat yang berdimensi besar sehingga kurang efektif dan efisien. Oleh karena itu perlu dirancang alat peringatan kantuk untuk mengurangi potensi kecelakaan secara *real time*. Penelitian ini merancang sistem deteksi kondisi kantuk dengan fitur peringatan jika terdeteksi kantuk berupa suara dari pengeras suara, sehingga operator/ pengemudi dapat mengantisipasi tindakannya untuk menghindari bahaya. Rancangan menggunakan sistem untuk mendeteksi mata terbuka dan tertutup melalui aspek rasio mata menggunakan *web camera* yang kemudian diproses menggunakan Raspberry Pi 3B, dengan menggunakan pustaka OpenCV dan DLIB, serta menggunakan bahasa pemrograman Python. Metode yang digunakan adalah *Eye Aspect Ratio* (EAR) dan *Histogram of Oriented Gradients* (HOG). Hasil rancangan menunjukkan bahwa alat yang dibuat berjalan dengan baik namun mengalami jeda waktu karena pengaruh rancangan sistem dari *coding program* yang dibuat dan spesifikasi alat yang digunakan. Alat yang dibangun dapat diterapkan secara ringkas, portabilitas, efektif dan efisien dalam berkendara sehingga aman dan nyaman digunakan.

Kata kunci: *kantuk, web camera, Raspberry Pi, OpenCV, Dlib, Python, Eye Aspect Ratio (EAR) dan Histogram of Oriented Gradient (HOG).*

ABSTRACT

Drowsiness is a significant cause of accidents that often goes unnoticed until errors in decision making occur. Many drowsiness detection devices have been designed, but it is necessary to use a tool that is placed on the user's body physiology and the use of a tool with large dimensions so that it is less effective and efficient. Therefore, it is necessary to design a drowsiness warning tool to reduce the potential for accidents in real time. This study designed a drowsiness detection system with a warning feature if drowsiness is detected in the form of sound from a loudspeaker, so that the operator/driver can anticipate their actions to avoid danger. The design uses a system to detect open and closed eyes through the eye aspect ratio using a web camera which is then processed using a Raspberry Pi 3B, using OpenCV and DLIB libraries, and using the Python programming language. The methods used are Eye Aspect Ratio (EAR) and Histogram of Oriented Gradient (HOG). The results of the design show that the tools made run well but experience a time lag due to the influence of the system design from

the coding program made and the specifications of the tools used. The tool that is built can be applied in a concise manner, portability, effective and efficient in driving so that it is safe and comfortable to use.

Keywords : *Drowsiness, webcam, Raspberry Pi, OpenCV, Dlib, Python, Eye Aspect Ratio (EAR), and Histogram of Oriented Gradient (HOG).*

1. PENDAHULUAN

1.1 Pengantar

Manusia adalah makhluk yang selalu dipenuhi aktivitas dalam hidupnya. Salah satu aktivitas yaitu melakukan sebuah aktivitas fisik seperti contohnya mengemudi kendaraan. Mengemudi kendaraan membutuhkan konsentrasi yang tinggi, dengan tingkat konsentrasi yang tinggi memerlukan kondisi tubuh yang prima dalam pelaksanaannya. Pengemudi dituntut untuk tetap dalam kondisi prima dalam jangka waktu tertentu. Hal ini membutuhkan suatu energi yang diperoleh untuk memenuhi kebutuhan tersebut, namun energi yang diperoleh lama-kelamaan akan habis terkuras sehingga pengemudi membutuhkan waktu untuk beristirahat. Apabila manusia tidak melakukan istirahat secara optimal maka akan timbul kondisi kelelahan yang bisa menjadi faktor utama kondisi kantuk. Pengemudi juga tidak dapat memprediksi akan datangnya serangan kantuk (Kaida et al., 2007 dalam Fachrudin, Wahyuning, & Yuniar, 2015). Keadaan mengantuk ketika mengemudi tergolong kedalam salah satu tindakan yang berbahaya. Kantuk dalam pekerjaan seperti mengemudi dipengaruhi oleh tiga faktor utama yaitu faktor pekerjaan, karakteristik individu, dan tidur yang dilakukannya (Matibs et al., 2009 dalam Fachrudin, Wahyuning, dan Yuniar, 2015).

1.2 Identifikasi Masalah

Di Indonesia, angka kecelakaan lalu lintas di setiap tahunnya semakin meningkat namun belum diperoleh berapa banyak angka aktual kecelakaan akibat kantuk, namun demikian menurut Noy., dkk (2011) menyatakan kelelahan yang menjadi fase awal kondisi kantuk dianggap sebagai kontributor utama morbiditas dan mortalitas di tempat kerja dan jalan raya. Dilansir dalam penelitian studi tingkat kecelakaan lalu lintas jalan di Indonesia oleh data KNKT (Komite Nasional Keselamatan Transportasi) dari Tahun 2007-2016, terdapat 698 orang korban jiwa dan 1171 korban luka-luka, jumlah peningkatan tersebut disebabkan oleh beberapa faktor di antaranya manusia, sarana, dan prasarana. Faktor manusia merupakan faktor tertinggi penyebab kecelakaan lalu lintas dimana persentasenya mencapai 69.70 persen (Saputra, 2017). Faktor manusia bisa disebabkan akibat lengah, mengantuk, tidak terampil, mabuk, kecepatan tinggi, dan tidak menjaga jarak (Artiani, 2016).

Tingkat kantuk dapat diukur dengan metode objektif dan subjektif. Metode subjektif yang diteliti menurut Wahyuning dan Irianti (2018), deteksi kantuk dapat menggunakan metode *Karolinska Sleepiness Scale (KSS)*, dan *Visual Analogue Scale (VAS) of Alertness* kepada responden pengemudi. Sedangkan metode objektif saat ini terdapat banyak pengembangan sistem untuk mengurangi potensi kecelakaan. Seperti sistem untuk melakukan pengukuran rasa kantuk yang dapat diketahui dengan berbagai macam metode dan indikator yang telah diteliti oleh beberapa peneliti sebelumnya.

Menurut Putra, Desrianty, & Yuniar (2015), pendeteksian kantuk diukur dengan metode Elektroensefalografi (EEG) dan *Heart Rate Variability (HRV)*, isi penelitiannya mengenai Elektroensefalografi (EEG) berupa suatu sensor (elektroda) yang diletakkan pada bagian kepala manusia untuk merekam aktivitas gelombang otak, kemudian diolah menggunakan komputer untuk mengetahui tingkat kelelahan sebagai fase awal kondisi kantuk.

Menurut Efendi dkk., (2020), pendeteksian kantuk dapat diukur menggunakan metode *Heart Rate Variability* (HRV), isi penelitiannya berupa perancangan alat deteksi kantuk melalui pengukur sinyal detak jantung yang digunakan untuk mendeteksi kantuk dengan bantuan sensor *pulse* yang diletakkan pada ujung jari pengemudi dan kotak alat pemancar diletakkan pada tangan pengemudi menggunakan Raspberry Pi.

Menurut Amirullah dkk., (2018), isi penelitiannya berupa rancangan alat yang terdiri dari mikrokontroler dan sensor yang digunakan pada kepala pengemudi, kondisi kantuk dapat dilihat dari posisi kepala tegak jika pengemudi dalam kondisi tubuh normal dan akan mengalami perubahan posisi secara tiba-tiba jika pengemudi mengantuk dan memberikan peringatan suara. Dari ketiga penelitian ini diketahui bahwa penggunaan alat untuk mendeteksi kantuk harus dipasangkan pada faal tubuh pengguna menggunakan alat-alat dari ketiga penelitian tersebut, hal ini menyebabkan kemungkinan pengguna akan mengalami rasa tidak nyaman karena menggunakan alat yang dipasang pada faal tubuh pengguna sehingga tidak cocok diterapkan untuk mengemudi.

Selain itu deteksi kantuk dapat dideteksi dengan cara pengolahan citra digital secara *computer visual* melalui perubahan pola wajah, salah satu parameternya adalah dari kedipan mata secara antarmuka antara manusia dan mesin. Menurut Kuswara (2013), deteksi kantuk dapat diketahui menggunakan perangkat keras yaitu kamera yang digunakan sebagai *input* untuk memuat video, kemudian dari *input* tersebut diproses menggunakan komputer dan diolah menggunakan perangkat lunak pendukung untuk mengetahui *Eye Aspect Ratio* (EAR) menggunakan metode *haar cascade classifier* yang dikembangkan oleh Viola (2001).

Penggunaan metode *haar cascade classifier* ini dapat digolongkan sebagai metode yang sudah umum digunakan, sedangkan terdapat metode lain untuk pengolahan citra digital seperti penelitian menurut Cech dan Soukupova (2016), menjelaskan penggunaan metode *Histogram Of Oriented Gradients* (HOG) dan pustaka DLIB yang digunakan untuk mengekstraksi fitur pada objek gambar dengan menggunakan objek manusia untuk mengetahui kedipan mata.

Berdasarkan penelitian yang telah dijelaskan bahwa pendeteksian rasa kantuk dapat dideteksi dengan membutuhkan pengoperasian alat yang harus digunakan pada faal tubuh pengguna. Masalah yang timbul yaitu kurang efektif dan efisien jika digunakan dalam kondisi beraktivitas yaitu mengemudi kendaraan karena dapat mengganggu dan bahkan bisa menjadikan sebagai sumber kecelakaan kerja lainnya, dan penelitian sebelumnya mengenai pengolahan citra digital dalam penelitian tersebut masih menggunakan PC/laptop yang secara ukuran memiliki dimensi besar sehingga sulit diterapkan secara ringkas dan portabilitas pada saat mengemudi di kendaraan sehingga kurang efektif dan efisien, maka dari itu diperlukan perancangan alat deteksi dini lebih lanjut yang memungkinkan dalam penggunaan pada saat mengemudi.

Letak penelitian ini perlu adanya perancangan alat agar dapat memberi peringatan dini untuk meminimalisasi risiko kecelakaan mengemudi tanpa diletakkan pada faal tubuh pengemudi dengan menggunakan perangkat keras *web camera* yang dibuat secara *real time* berbasis *computer vision* untuk sebagai *input* pengolahan citra digital, serta dengan penggunaan perangkat keras untuk sebagai sistem pengolah data perancangan yaitu Raspberry Pi yang sudah termasuk didalam bagian *mini-PC* yang secara dimensi lebih kecil dari pada penggunaan Laptop/PC yang telah digunakan pada peneliti sebelumnya, sehingga dapat digunakan secara ringkas, portabilitas dan *real time* guna mencapai tujuan penggunaan alat yang efektif dan efisien.

2. METODOLOGI PENELITIAN

2.1 Pemilihan Metode

Terdapat beberapa metode objektif yang dapat digunakan untuk melakukan deteksi dini kondisi, metode objektif yang bisa digunakan antara lain yaitu pendeteksian kantuk dapat diukur dengan metode Elektroensefalografi (EEG), metode *Heart Rate Variability* (HRV), metode perubahan posisi kepala, metode *Haar Cascade Classifier*, *Eye Aspect Ratio* (EAR), dan *Histogram Of Oriented Gradient* (HOG) dan lain-lain yang tidak dibahas di dalam penelitian ini. Berdasarkan permasalahan yang ditemukan maka diperlukan perancangan alat deteksi dini kondisi kantuk secara ringkas, portabilitas, efektif dan efisien dengan pemanfaatan pengolahan citra digital secara *real time* berbasis *computer vision*. Metode yang cocok digunakan untuk mendeteksi kondisi kantuk adalah *Eye Aspect Ratio* (EAR) dan *Histogram Of Oriented Gradient* (HOG) yaitu metode yang dapat melihat dari adanya mata seseorang tertutup dan terbuka. Metode ini memiliki keakuratan yang tinggi dalam mendeteksi suatu objek. Kelebihan metode ini dibandingkan metode *haar cascade classifier* adalah proses ekstraksi citra digital pada bagian wajah untuk mendeteksi mata sangat akurat karena berbentuk titik-titik sudut koordinat yang akan berubah secara *real time* mengikuti pergerakan video, jika metode *haar cascade classifier* masih berbentuk kotak persegi, serta metode tersebut tergolong metode yang sudah lama.

2.2 Tahapan Perancangan

Tahapan rancangan yang akan dibuat dalam penelitian ini hingga alat jadi serta implementasi penggunaannya. Perancangan sistem meliputi identifikasi sistem studi kasus, rancangan pemodelan sistem, rancangan perangkat keras, rancangan perangkat lunak, dan rancangan program. Hingga pembuatan alat deteksi dini kondisi kantuk

Penjelasan setiap perancangan sistem adalah sebagai berikut:

1. Rancangan Pemodelan Sistem
Tahapan perancangan sistem pertama, yaitu rancangan pemodelan sistem, dalam rancangan pemodelan sistem terdapat aktivitas pemodelan sistem yang digunakan untuk menggambarkan seluruh sistem yang akan dibuat menggunakan tahapan-tahapan yang dijelaskan menggunakan *Unified Modelling Language* (UML) yang terbagi menjadi 3 pembahasan, antara lain *use case diagram*, *activity diagram*, dan *sequence Diagram*
2. Rancangan perangkat keras
Tahapan perancangan sistem ke-dua, yaitu rancangan perangkat keras, dalam tahapan ini merupakan penjabaran perangkat keras apa saja yang akan digunakan. Dalam penelitian ini menggunakan perangkat keras utama yaitu Raspberry Pi 3B, *web camera* dan perangkat keras pendukung untuk menjalankan Raspberry Pi 3B.
3. Implementasi perangkat keras
Implementasi perangkat keras menjelaskan secara detail perangkat keras yang digunakan dalam perancangan alat deteksi dini kondisi kantuk ini. Beserta langkah-langkah perancangan sistem perangkat keras untuk deteksi dini kondisi kantuk.
4. Rancangan perangkat lunak
Tahapan perancangan sistem ke-tiga, yaitu rancangan perangkat lunak, dalam tahapan ini merupakan penjabaran perangkat lunak apa saja yang akan digunakan untuk penelitian ini, menggunakan perancangan perangkat lunak berupa *software*, *library*, dan aplikasi, antara lain yaitu sistem operasi Raspbian OS, bahasa pemrograman Python, Thony Ide Python, OpenCV, DLIB, dan *shape-predictor-68-face-landmark* yang digunakan untuk mendeteksi area wajah.
5. Implementasi perangkat lunak
Implementasi perangkat lunak menjelaskan secara detail perangkat lunak berupa *software*, *library*, dan aplikasi yang digunakan dalam perancangan perangkat lunak

- untuk alat deteksi dini kondisi kantuk ini. Menjelaskan tahapan instalasi dan pengujian perangkat lunak.
6. Rancangan program
Tahapan perancangan sistem keempat, yaitu rancangan pembuatan *coding program* dengan mengubah hasil rancangan berdasarkan rancangan pemodelan sistem kedalam suatu perintah data *code* untuk menghasilkan suatu hasil desain program yang telah dirancang.
 7. Implementasi program
Implementasi program menjelaskan tentang penerapan *coding program* yang telah dibuat untuk melakukan deteksi dini kantuk. Penjelasan dibuat dalam diagram alir agar dimulai dari program itu berjalan hingga program itu selesai.
 8. Pembuatan alat deteksi dini kondisi kantuk
Pembuatan alat adalah menggabungkan seluruh rangkaian rancangan yang telah dibuat dan menjadi satu kesatuan untuk melakukan proses deteksi dini kondisi kantuk menjadi sebuah alat utuh siap pakai serta penerapan penggunaan alat di kendaraan.
 9. Implementasi penggunaan alat
Implementasi penggunaan alat ini akan menjelaskan bagaimana cara langkah-langkah penggunaan alat yang dibuat untuk tahap pengujian.

3. STUDI LITERATUR

3.1 Studi Literatur

Studi literatur merupakan landasan teori yang akan digunakan untuk melakukan perancangan alat deteksi dini kondisi kantu.

3.1.1 Kantuk

Kantuk didefinisikan sebagai sebuah proses yang dihasilkan dari ritme sirkadian dan kebutuhan untuk tidur. Titik awal terjadinya kantuk tidak dapat diprediksi secara pasti. Pengemudi juga tidak dapat memprediksi akan datangnya serangan kantuk (Kaida et al., 2007). Kondisi mengantuk atau tidak prinsip dasarnya adalah mata tertutup untuk jangka waktu lebih lama dibandingkan pada saat orang itu aktif. Karenanya jika nilai durasi kedipan mata melebihi di atas ambang tertentu maka orang tersebut dapat dikatakan berada dalam keadaan mengantuk (Cech & Soukupova, 2016).

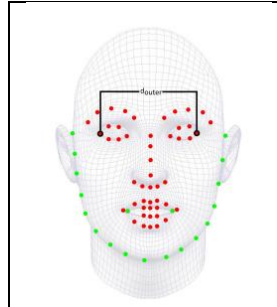
Menurut Kuswara (2013), menjelaskan bahwa parameter mengantuk dapat dilihat berdasarkan kondisi mata. Dalam keadaan normal atau bebas dari stres rata-rata kedipan mata adalah 15 sampai 20 kali permenit. Frekuensi ini menurun sampai 3 kali permenit ketika membaca. Frekuensi tersebut meningkat dalam keadaan stress, tertekan ataupun ketika menutupnya mata saat dibutuhkan. Indikator untuk mengetahui seseorang sedang mengantuk dapat dideskripsikan ketika kondisi normal (tidak mengantuk) posisi kelopak mata membuka lebar sebelum menutup, ketika menutup memiliki interval waktu yang cepat (kurang dari satu detik). Ketika seseorang mulai lelah dan mengantuk, jarak antara kedua kelopak mata semakin menyempit dan frekuensi kedipan semakin menurun hingga tertidur.

3.1.2 Pengolahan Citra Digital

Menurut Fadilsyah (2007), pengolahan citra (*image processing*) adalah pengolahan suatu citra (gambar) dengan menggunakan komputer secara khusus, untuk menghasilkan suatu citra yang lain. Menurut Andono dkk., (2017), ada beberapa tipe citra yang sering digunakan untuk penelitian, antara lain adalah citra biner, citra *grayscale*, citra warna.

3.1.3 Deteksi Wajah

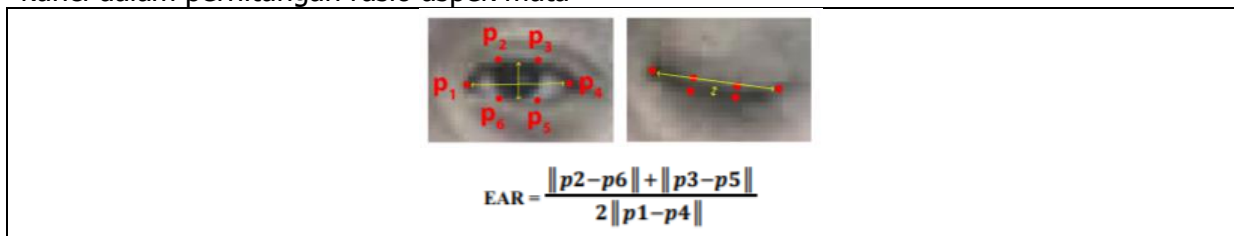
Pendeteksian area wajah dapat dideteksi melalui *facial landmark detection* merupakan metode terbaru yang bekerja dengan mengalokasikan titik-titik pada wajah untuk menentukan suatu bentuk biologis dari wajah manusia (Purwanto & Utaminigrum, 2020). *Metode facial landmark detection* dapat menampilkan hasil berupa angka yang berfungsi memetakan bagian-bagian dari wajah manusia dan jumlah *landmark* tergantung pada dataset atau aplikasi yang digunakan. *Facial Landmark* yang digunakan adalah anotasi 68 Titik pada dataset iBUG 300-W (Shen, 2015) Berikut salah satu *facial landmark detection* yaitu 68 *landmarks* dapat dilihat pada Gambar 3.1



Gambar 3.1 Facial Landmark 68
Sumber: Shen, et al. (2015)

3.1.4 Metode *Eye Aspect Ratio* (EAR)

Menurut Pandey & Muppalaeni (2021), *Eye Aspect Ratio* (EAR) adalah salah satu metode untuk menghitung jarak antara kelopak mata atas dengan kelopak mata bawah berdasarkan titik geometri wajah pada mata. EAR menyusut dengan cepat menuju nol selama keadaan mengantuk. EAR adalah perbandingan panjang mata vertikal dengan panjang horizontal. Seperti dapat terlihat pada Gambar 3.2 bahwa 6 landmark. *Landmark* ini memainkan peran kunci dalam perhitungan rasio aspek mata



Gambar 3.2 Rumus dan Contoh *Eye Aspect Ratio*

Sumber : <https://www.pyimagesearch.com/2017/04/24/eye-blinkdetection-opencv-python-dlib/>

3.1.5 *Histogram of Oriented Gradient* (HOG)

Menurut Dalal & Triggs (2005), HOG digunakan untuk mengekstraksi fitur pada objek gambar dengan menggunakan objek manusia. Berdasarkan langkahnya, proses awal pada metode HOG adalah mengkonversi citra RGB (*Red, Green, Blue*) menjadi *grayscale*, yang kemudian dilanjutkan dengan menghitung nilai gradien setiap piksel. Setelah mendapatkan nilai gradien, maka proses selanjutnya yaitu menentukan jumlah bin orientasi yang akan digunakan dalam pembuatan histogram. Proses ini disebut *spatial orientation binning*. Namun sebelumnya pada proses *gradient computer* gambar pelatihan dibagi menjadi beberapa *cell* dan dikelompokkan menjadi ukuran lebih besar yang dinamakan *block*. Proses normalisasi *block* digunakan perhitungan geometri HOG. Citra berwarna terdiri dari merah, hijau dan biru. Masing-masing komponen pada citra *true colour* mempunyai nilai dengan 256 kemungkinan nilai. Nilai tersebut dimulai dari nol untuk warna hitam dan 255 untuk warna putih. Konversi Citra *true*

colour ke *grayscale* mengubah nilai piksel dari *Red*, *Green*, dan *Blue* menjadi satu nilai yaitu keabuan.

3.1.6 Raspberry Pi

Menurut Edi (2020), Raspberry Pi adalah komputer papan tunggal (*single-board circuit; SBC*). Raspberry Pi atau biasa disebut Raspi merupakan komputer papan tunggal seukuran kartu kredit dengan operating system seperti komputer pada umumnya yang dapat melakukan pengolahan data, pengoperasian game komputer, program *computing* dan juga dapat digunakan untuk memutar video dari resolusi rendah hingga tinggi.

3.1.7 Bahasa Pemrograman Python

Menurut Sianipar dan Wadi (2015). Python lahir atas dasar keinginan untuk mempermudah seorang programmer dalam menyelesaikan tugas-tugasnya dengan cepat. Python dirancang untuk memberikan kemudahan yang sangat luar biasa kepada programmer baik segi efisiensi waktu, maupun kemudahan dalam pengembangan program dan dalam hal kompatibilitas dengan sistem.

3.1.8 OpenCV

OpenCV adalah sebuah *library computer vision* yang *open source* (terbuka). OpenCV didesain untuk komputasi yang efisien dan difokuskan pada aplikasi *real time*. Salah satu tujuan OpenCV adalah menyediakan infrastruktur *computer vision* yang mudah digunakan dan dapat membantu pengembang dalam membangun aplikasi citra dengan mudah. *Library* OpenCV mengandung lebih dari 500 *function* (Bradski dan Kaehler, 2008).

3.1.9 Dlib

Menurut Purwanto & Utaminingrum (2020) Dlib dalam implementasinya banyak digunakan untuk mendeteksi wajah. Dlib adalah suatu library yang berfungsi dengan cara menganalisis bagian wajah dengan mengekstrak nilai gambar. Dengan mengekstrak nilai pada wajah manusia dilibatkan menghasilkan 128 dimensional *feature vector* dlib akan digunakan untuk membantu mengolah wajah pada metode *facial landmarks*.

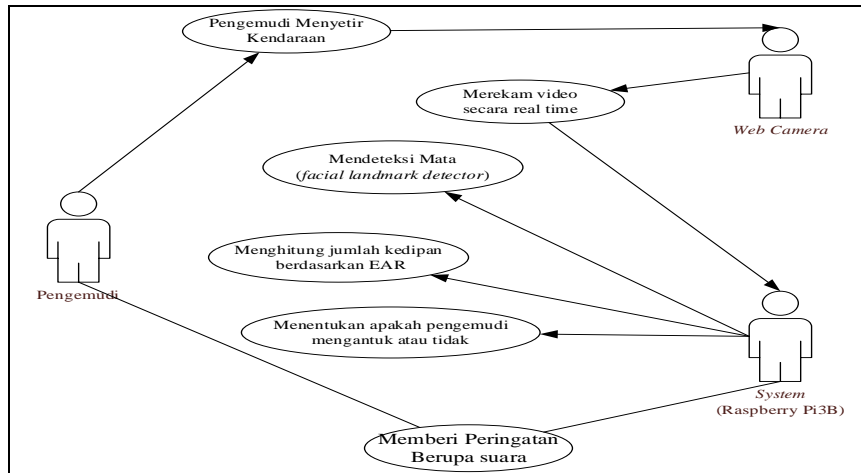
4. PERANCANGAN SISTEM DAN ANALISIS

4.1 Rancangan Pemodelan Sistem

Tahapan perancangan sistem pertama yaitu rancangan pemodelan sistem, terdapat aktivitas pemodelan sistem yang digunakan untuk menggambarkan seluruh sistem yang akan dibuat menggunakan tahapan yang dijelaskan menggunakan *Unified Modelling Language* (UML).

1. *Use Case Diagram*

Berikut merupakan hasil pemodelan sistem menggunakan *use case diagram* dapat dilihat pada Gambar 4.1.

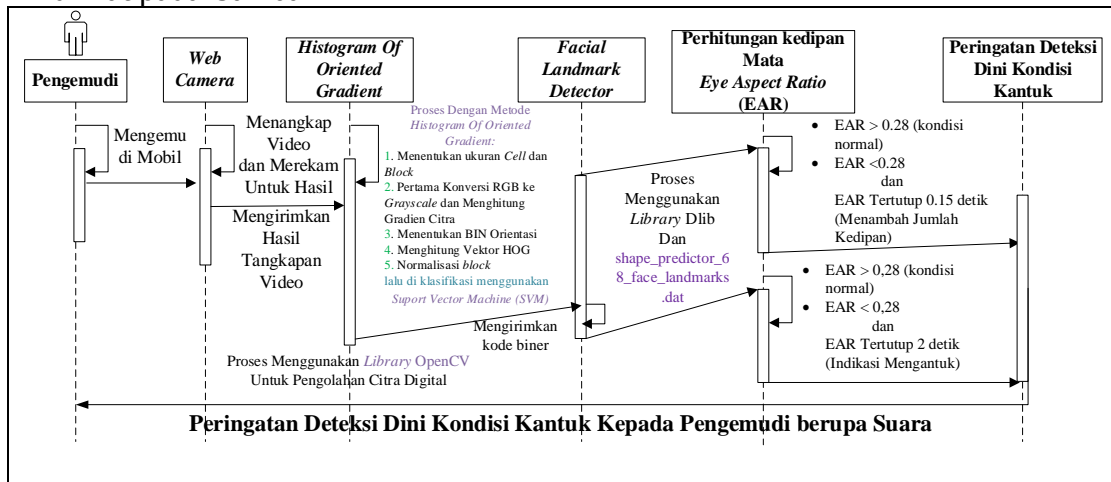


Gambar 4.1 Pemodelan Menggunakan Use Case Diagram

Penjelasan singkat mengenai *use case diagram* diatas pengemudi menyetir kendaraan lalu *web camera* menangkap gambar berupa video secara *real time* kepada subyek pengemudi, hasil video tersebut diolah oleh sistem Raspberry Pi 3B, pengolahan yang dimaksud untuk mendeteksi mata, menghitung jumlah kedipan mata, menentukan apakah pengemudi mengantuk atau tidak. Jika mengantuk sistem akan memberi peringatan kepada pengemudi berupa suara.

2. Sequence Diagram

Berikut merupakan hasil pemodelan sistem menggunakan *sequence diagram* dapat dilihat pada Gambar 4.2.



Gambar 4.2 Pemodelan Menggunakan Use Case Diagram

Penjelasan yaitu pertama pengemudi melakukan aktivitas yaitu mengemudi mobil, kemudian *web camera* menangkap rekaman berupa video, lalu hasil tangkapan video dikirimkan ke dalam sistem yang akan diolah melalui pengolahan citra digital secara *computer vision*. Hasil tangkapan layar masih berupa citra warna RGB (*red, green, blue*) dengan menggunakan metode *histogram of oriented gradient* (HOG) citra warna tersebut diubah menjadi citra *grayscale* dengan tujuan untuk mempermudah hasil citra agar dapat diolah ke tahapan selanjutnya yang terdiri dari 5 tahapan pada metode HOG. Setelah didapatkan kode biner, maka akan dilakukan proses *facial landmark* menggunakan *shape_predictor_68_face_landmark.dat* untuk mengidentifikasi wajah terutama pada penelitian ini adalah parameternya adalah mata. Dilanjutkan dengan perhitungan aspek rasio mata dengan dua proses yaitu untuk melakukan perhitungan jumlah kedipan mata, dan proses untuk mengetahui apakah pengemudi mengantuk

atau tidak. Pada pemodelan sistem ini dibuatlah Batasan untuk aspek rasio mata yaitu sebesar 0,28. Hal ini ditujukan untuk mengetahui apakah mata tertutup atau tidak, dan apakah mata sedang di kondisi normal atau mengantuk. Jika mengantuk maka akan ada suatu peringatan berupa suara untuk memperingati pengemudi agar tidak tertidur guna untuk menghindari kecelakaan kerja.

4.2 Rancangan Perangkat Keras dan Perangkat Lunak

Tahapan perancangan sistem kedua dan ketiga, yaitu perancangan perangkat keras dan perancangan perangkat lunak, dalam tahapan ini merupakan penjabaran perangkat keras apa saja yang akan digunakan. Dalam penelitian ini menggunakan perangkat keras utama yaitu Raspberry Pi 3B, dan perangkat keras pendukung untuk menjalankan Raspberry Pi. Perangkat Keras dapat dilihat pada Tabel 4.1 dibawah ini.

Tabel. 4.1 Perangkat Keras

Perangkat Keras	Perangkat Lunak
Raspberry Pi 3B	Raspbian OS
SD card 32GB Sandisk Extreme Pro A1	Python 3.7
Logitech C922 Pro Stream Webcam	Thony IDE Python
Logitech MK220 Keyboard dan Mouse Wireless (Optional)	VNC viewer
Smartphone Android/PC/Monitor(Optional)	OpenCV
Kabel Hdmi (optional)	DLIB
Jack Aux Audio (optional)	shape-predictor-68-face-landmarks
Speaker (optional)	
Sumber Arus Listrik DC 5 Volt 2A (optional)	
Jaringan Internet / Wifi	

4.3 Rancangan Program

Tahapan perancangan sistem keempat, yaitu rancangan pembuatan *coding program* menggunakan *software* Thony IDE Python di dalam sistem operasi Raspbian OS dengan mengubah hasil rancangan kedalam suatu perintah data *code* untuk menghasilkan suatu hasil desain program yang telah dirancang.

4.4 Pembuatan Alat Deteksi Dini Kondisi Kantuk

Pembuatan alat adalah menggabungkan seluruh rangkaian rancangan yang telah dibuat dan menjadi satu kesatuan sehingga terciptalah sebuah alat. Berikut merupakan gambar jelasnya dapat dilihat pada Gambar 4.3.



Gambar 4.3 Alat Deteksi Dini Kondisi Kantuk

Berikut ini merupakan penerapan penggunaan alat deteksi dini kondisi kantuk yang diterapkan pada kendaraan. Dapat dilihat pada Gambar 4.4



Gambar 4.4 Penerapan Alat Deteksi Dini Kondisi Kantuk

4.5 Analisis

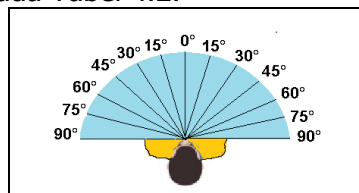
Analisis disini menjelaskan mengenai pembahasan hasil dari pembuatan alat deteksi dini kondisi kantuk yang telah dibuat dan dilakukan pengujian terhadap objek pengemudi untuk mendapatkan hasil apakah alat yang dibuat berjalan dengan baik atau tidak. Sub bab ini terdiri dari pengujian dan analisis.

4.5.1 Pengujian

Pengujian dilakukan dalam beberapa macam pengujian, antara lain adalah:

1. Pengujian deteksi pola wajah

Pengujian pertama ini adalah pengujian terdeteksi atau tidak nya mata, dilakukan dengan beberapa variasi posisi sudut ke arah kiri dan ke kanan, diantaranya adalah : 0° (normal), 15°, 35°, 45°, 90° seperti ditunjukkan pada Gambar 4.5, Gambar 4.6 dan pengujian deteksi pola pada Tabel 4.2.



Gambar 4.5 Ilustrasi Sudut Pergerakan Wajah Secara Horizontal

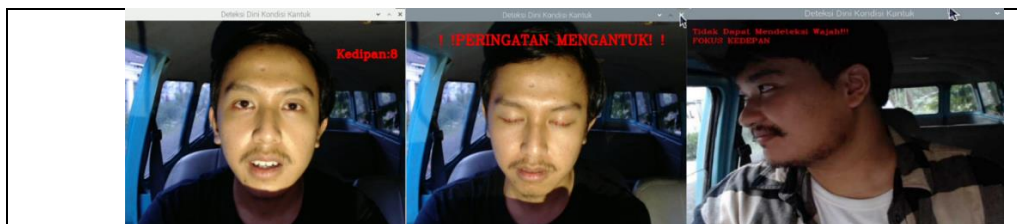
Keterangan :

Tanda "√" menunjukkan mata terdeteksi oleh sistem

Tanda "X" menunjukkan mata tidak terdeteksi oleh sistem

Tabel 4.2 Pengujian Deteksi Pola Wajah

Subjek Pengguna	Normal	15°	35°	45°	90°
1	√	√	√	X	X
2	√	√	√	√	X
3	√	√	√	X	X
4	√	√	√	X	X
5	√	√	√	√	X
6	√	√	√	X	X
7	√	√	√	X	X



Gambar 4.6 Contoh Hasil Pengujian

Penjelasan gambar diatas yaitu tampilan *frame* sistem Ketika perhitungan kedipan mata, peringatan mengantuk berupa tampilan dan *output* peringatan suara, dan peringatan wajah tidak terdeteksi oleh sistem yang diharapkan agar fokus kedepan.

2. Pengujian jarak terhadap wajah

Pengujian ke-dua yaitu berupa parameter deteksi dengan jarak dari objek pengguna dengan alat berupa jarak dengan radius sekitar 20cm, 40cm, 60cm, 80cm dan 100 cm. Parameter ini juga sangat berpengaruh pada bagian pendeteksian wajah kali ini adalah jarak antara subjek pengguna dengan *web camera* yang digunakan apakah terdeteksi atau tidak. Berikut dapat dilihat pada Tabel 4.3.

Tabel 4.3 Pengujian Jarak Terhadap Wajah

Subjek Pengguna	Jarak Mata Dengan Alat (cm)					
	20	40	60	80	100	>120
1	√	√	√	√	X	X
2	√	√	√	√	√	X
3	√	√	√	√	X	X
4	√	√	√	√	√	X
5	√	√	√	√	√	X
6	√	√	√	√	√	X
7	√	√	√	√	√	X

Dari pengujian diatas dapat dianalisis dan disimpulkan bahwa semakin jauh jarak wajah terhadap kamera makan akan tidak terdeteksi, dan disimpulkan bahwa jarak optimum berada di jarak 20 cm hingga 80 cm.

3. Pengujian berdasarkan kondisi waktu

Tahapan ketiga ini dilakukan pada kondisi waktu siang hari, sore hari dan malam hari. Asumsi pengujian untuk siang hari yaitu rentang waktu antara pukul 09:00-13:00, asumsi sore hari yaitu rentang waktu pukul 13:00-17:00, dan asumsi malam hari yaitu rentang waktu pukul 17:00-24:00. Pengujian dilakukan selama 15 menit berkendara. Berikut dapat dilihat pada Tabel 4.4 dibawah ini.

Tabel 4.4 Pengujian Berdasarkan Kondisi Waktu

Subjek Pengguna	Kon di si Hari		Intensitas Cahaya (Lux)	Terdeteksi atau Tidak	Jumlah Kedipan	Peringatan Berbunyi
1	Sian g		83	√	114	√
	Sor e		114	√	123	√
	Mal am		80	√	97	√
2	Sian g		94	√	101	√
	Sor e		101	√	130	√
	Mal am		57	√	69	√

3	Sian g		98	√	134	√
	Sor e		91	√	132	√
	Mal am		49	X	X	X
4	Sian g		106	√	108	√
	Sor e		82	√	143	√
	Mal am		26	X	X	X
5	Sian g		121	√	125	√
	Sor e		79	√	81	√
	Mal am		4	X	X	X
6	Sian g		137	√	74	√
	Sor e		76	√	113	√

	Mal am		0	X	X	X
7	Sian g		132	√	145	√
	Sor e		68	√	131	√
	Mal am		0	X	X	X

4.5.2 Analisis Hasil Pembuatan Alat

Berdasarkan keterkaitan antara rancangan perangkat keras dengan perangkat lunak saling berkaitan sehingga terciptalah alat yang bisa digunakan untuk deteksi dini kondisi kantuk dengan pemanfaat pengolahan citra digital. Keberhasilan *coding* program pada penelitian ini semuanya berfungsi dengan baik, semua perintah mulai dari proses perhitungan jumlah kedipan mata berjalan dengan baik, proses pada saat pandangan sedang tidak terjangkau dan terdeteksi oleh *web camera* sehingga memberikan pesan berupa tulisan "tidak dapat mendeteksi wajah" & "harap fokus kedepan" berjalan dengan baik, dan yang paling berguna yaitu *output* peringatan suara berfungsi dengan baik sehingga dapat membantu pengemudi untuk mengurangi risiko kecelakaan kerja akibat mengantuk.

Guna keberhasilan dalam penggunaan alat ini diperlukan terlebih dahulu kalibrasi penyesuaian peletakan alat yang akan digunakan dengan penempatan pada bagian depan yang memungkinkan pendeteksian wajah akan berjalan dengan sempurna dan harus mengetahui terlebih dahulu apakah jenis kondisi mata pengemudi berada pada kategori jenis kelopak mata apa, guna untuk mengestimasi batas EAR yang akan digunakan. Batasan EAR yang sering umum digunakan di luar negeri contohnya pada penelitian Kamarudin, N. et al. (2019) membatasi sebesar 0,25, sedangkan di penelitian ini penulis membatasi 0.28 guna agar lebih sedikit fleksibilitas terhadap jenis mata kategori sipit. Tentunya Batasan EAR ini tergantung pada jenis kelopak mata pengemudi yang berkaitan dengan ras manusia, usia, kelainan genetik dan penyakit.

Berdasarkan hasil pembuatan alat diketahui bahwa alat mengalami jeda waktu kurang lebih selama 0,5-3 detik tidak menentu namun umumnya jeda waktu berapa di kisaran range 0,2-2 detik. Hal ini disebabkan karena untuk melakukan pendeteksian kondisi kantuk ini membutuhkan 2 macam *library* yang berbeda yaitu OpenCV dan Dlib yang digunakan untuk pengolahan citra digital serta faktor lain yaitu isi pemrograman dalam *coding program* yang dibuat untuk melakukan proses perhitungan kedipan mata & proses perekaman hasil tangkapan layar dalam format .AVI selama penggunaan alat hingga terdeteksi kondisi kantuk, sehingga memakan kapasitas CPU *usage* sehingga berpengaruh terhadap performa kecepatan alat tersebut yang mengakibatkan jeda waktu.

Hal ini yang menjadikan perangkat keras Raspberry Pi 3B mengalami sedikit proses yang cukup memakan beban *core* atau *capacity performance* dari spesifikasi perangkat Raspberry Pi nya itu, sehingga mengalami sedikit jeda waktu, ditambah hal yang kemungkinan menjadi pengaruh adalah selama *coding program* sedang dijalankan maka akan menampilkan *frame* di *interfaces desktop* Raspberry Pi 3b, selain itu faktor spesifikasi *webcam* yang digunakan yaitu tipe Logitech C9200 *Pro Stream Webcam* dengan spesifikasi 1080p dan 30fps yang sangat berpengaruh terhadap kecepatan tangkap dan proses pada sistem, sistem mendeteksi *frame rate per-second* (FPS) sebesar 27-30fps pada saat kalibrasi yang dapat dilihat di dalam terminal karena penulis membuat proses *coding program* ini guna untuk mengetahui pengaruh FPS terhadap kecepatan reaksi dengan menggunakan FPS yang cukup tinggi. Sebelumnya penguji juga telah mencoba penggunaan *webcam* tipe lain dengan spesifikasi piksel yang rendah dengan fps yang tinggi yang ternyata mengalami perbedaan terhadap kecepatan reaksi tangkap. Namun Penulis tidak memasukan penelitian perbandingan penggunaan *webcam* yang digunakan, akan tetapi penulis dapat memastikan bahwa spesifikasi *webcam* berkaitan dengan nilai piksel dan fps sangat mempengaruhi hasil.

4.5.3 Analisis Hasil Pengujian

Analisis hasil pengujian dilakukan dalam beberapa macam pengujian, antara lain adalah:

- a. Analisis dari pengujian deteksi pola wajah
Berdasarkan hasil pengujian deteksi pola wajah dengan variasi sudut yang dibuat diketahui bahwa sudut 15° dan 30° dapat terdeteksi karena tingkat kemiringan sudut ini masih dapat dibaca oleh *facial landmark detector 68*, pada sudut 45° Terkadang terdeteksi dan terkadang tidak, kemudian untuk sudut pandang yang terakhir sudah dipastikan tidak terdeteksi hal ini disebabkan tidak adanya acuan untuk mendeteksi area wajah.
- b. Analisis dari pengujian jarak terhadap wajah
Diketahui bahwa parameter deteksi dengan jarak dari subjek pengguna dengan alat berupa jarak dengan radius sekitar 20cm, 40cm, 60cm, 80cm dan 100 cm. Berdasarkan hasil pengujian bahwa nilai optimum jarak berada pada 20cm sd 80 cm, jika lebih dari 100 cm maka sistem tidak dapat membaca bagian mata. Pengujian ini dicoba dengan merubah tuas jauh dekat kursi secara horizontal.
- c. Analisis dari pengujian berdasarkan kondisi waktu
Kondisi waktu dalam penelitian ini dibagi menjadi siang hari, sore hari dan malam hari dengan asumsi batasan jam yang telah ditentukan. Dalam pengujian ini kondisi optimum berada di siang hari dan sore hari karena intensitas cahaya sangat mempengaruhi untuk melakukan pendeteksian wajah, berbeda dengan malam hari sistem tidak berjalan dengan baik bahkan tidak terdeteksi sebab intensitas cahaya sangat kurang. Hal ini membutuhkan lampu tambahan di dalam kendaraan agar sistem dapat mendeteksi/penggantian kamera yang digunakan menjadi *camera night vision*.

5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Kesimpulan dari penelitian ini yaitu telah menghasilkan alat deteksi dini kondisi kantuk untuk mengurangi risiko kecelakaan kerja berbasis pengolahan citra digital. Berdasarkan temuan penelitian yang dilakukan selama ini, dapat disimpulkan bahwa hasil perancangan dan pembuatan alat telah tercapai. Kesimpulan nya antara lain sebagai berikut.

1. Perancangan alat deteksi dini kondisi kantuk berjalan dengan sempurna berdasarkan penggunaan metode *Eye Aspect Ratio* (EAR) dan metode *Histogram of Oriented Gradient* (HOG).
2. Sistem yang dibangun dapat mendeteksi kantuk melalui parameter aspek rasio mata berdasarkan batasan kedipan mata.

3. Alat yang dibangun dapat diterapkan secara ringkas, portabilitas, efektif dan efisien dalam berkendara sehingga aman dan nyaman digunakan.
4. Performansi alat yang dibuat mengalami jeda waktu kurang lebih selama 0,5-3 detik tidak menentu namun umumnya jeda waktu berapa di kisaran range 0,2-2 detik hal ini disebabkan beberapa faktor yang mempengaruhi.

5.2 Saran

Alat dan sistem yang dibangun dapat dikembangkan menjadi lebih baik lagi dengan melakukan *upgrade* Raspberry PI atau menambahkan *coding*, *library*, dan perangkat keras/lunak tambahan sehingga lebih bervariasi. Berikut adalah beberapa saran yang bisa penulis berikan kepada pembaca untuk studi lebih lanjut.

1. Disarankan untuk penelitian selanjutnya dikembangkan ke platform *Internet Of Thing*. Seperti contoh pengiriman data hasil *output* rekaman video pada alat yang telah dibuat ini dikirimkan secara *real time* ke database atau dikirimkan melalui social media contohnya telegram, guna mengetahui kondisi pengemudi jika mengalami mengantuk. Atau inovasi perkembangan lainnya.
2. Disarankan pendeteksian deteksi dini kantuk selanjutnya bisa ditambahkan tidak hanya mendeteksi kedipan mata saja.
3. Disarankan melakukan upgrade alat terutama Raspberry Pi 3B ke versi terbarunya yaitu Raspberry Pi 4 atau menggunakan mikrokontroler lainnya dengan spesifikasi yang lebih tinggi agar tidak mengalami *delay* / keterlambatan respon.
4. Disarankan malam hari menggunakan *camera night vision* agar alat dapat digunakan.

DAFTAR PUSTAKA

Rujukan Buku:

- Andono, P. N., Sutojo, T., & Muljono. (2017). Pengolahan citra digital. Yogyakarta: Penerbit ANDI.
- Fadilisyah, (2007). *Computer Vision dan Pengolahan Citra*. Yogyakarta. Penerbit: Andi.

Rujukan Jurnal:

- Amirullah, M., Kusuma, H., dan Tarsipan. (2018). Sistem Peringatan Dini Menggunakan Deteksi Kemiringan Kepala pada Pengemudi Kendaraan Bermotor yang Mengantuk. *Jurnal Teknik ITS* Vol. 7, No. 2. ISSN: 2337-3539. Fakultas Teknologi Elektro Institut Teknologi Sepuluh Nopember.
- Artiani, P.G.(2016). Analisis Faktor Penyebab Kecelakaan Lalu lintas Sebagai Acuan Perencanaan Jalan Untuk Meningkatkan Keselamatan. *Jurnal Teknik Sipil Sekolah Tinggi Teknik-PLN (STT-PLN)*. Vol. 5, No.2. ISSN: 2356-1491. Jurnal Forum Mekanika.
- Bradski, Gary dan Kaehler, Adrian. (2008). *Learning OpenCV Computer Vision with OpenCV library*.
- Cech, Jan dan Soukupova, Tereza. (2016). *Real-Time Eye Blink Detection using Facial Land marks*. Computer Vision Winter Workshop.
- Dalal, Navneet, dan Triggs, Bill. (2005). *Histograms of Oriented Gradients for Human Detection*. France.
- Efendi, Y., Putri, N.A., Rahmadden, & Imardi, S. (2020). Prototype Alarm Deteksi Mata Kantuk Menggunakan Sensor Pulse Berbasis Raspberry Pi 3. *JOISIE Journal Of Information System And Informatics Engineering*, 4 (2). pp. 77-83. ISSN 2527-3116.
- Fachrudin, F., Wahyuning, S.C., & Yuniar. (2015). Analisis Pengaruh Tingkat Kantuk Terhadap Kecepatan Reaksi Masinis Daerah Operasi II Bandung. *Reka Integra* ISSN: 2338-5081. Jurnal Online Institut Teknologi Nasional. Jurusan Teknik Industri. No 01. Vol 03.
- Kaida et, al., (2007), Use of Subjective and Physiological Indicators of Sleepiness to Predict Performance during a Vigilance Task. *Industrial Health*.

- Kamarudin, N. et al. (2019) "Implementation of haar cascade classifier and eye aspect ratio for driver drowsiness detection using raspberry Pi," Universal Journal of Electrical and Electronic Engineering. Eng., vol. 6, no. 5, pp. 67–75.
- Kuswara, Randy. (2013). Aplikasi pendeteksi mata mengantuk berbasis citra digital menggunakan metode haar classifier secara real time. Bandung: UNIKOM.
- Matib, J. dan Hess, W.C. (2009). Sleepiness and Vigilance Test. Bern University Hospital. Switzerland
- Noy, dkk (2011). Future directions in fatigue and safety research. Accident Analysis and Prevention 43. 495-497.
- Putra, A.G., Desrianty, A., dan Yuniar. (2015). Analisis Tingkat Kelelahan dan Kantuk Pada Supir Bus Berdasarkan Heart Rate Variability (HRV) dan Electroencephalograms (EEG)*. Reka Integra ISSN: 2338-5081. Jurnal Online Institut Teknologi Nasional. Jurusan Teknik Industri. No 04. Vol 03.
- Saputra, D. A. (2017). Studi Tingkat Kecelakaan Lalu Lintas di Indonesia Berdasarkan Data KNKT (Komite Nasional Keselamatan Transportasi Dari tahun 2007-2016).
- Shen, J. et al. (2015). The First Facial Landmark Tracking in-the-Wild Challenge: Benchmark and Results. 2015 IEEE International Conference on Computer Vision Workshop (ICCVW).
- Sianipar R.H., Wadi, H. (2015). Pemrograman Python (Teori dan Implementasi). Bandung: Informatika.
- Pandey, N. N., dan Muppalaneni, N. B. (2021). Real-Time Drowsiness Identification based on Eye State Analysis. 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS).
- Purwanto, A., & Utaminigrum, F. (2020) Deteksi Pergerakan Mata dan Kedipan Untuk Memilih Empat Menu Display Menggunakan Probabilitas Berdasarkan Facial Landmark. Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, vol. 3, no. 11, p. 10865-10874, feb. 2020. ISSN 2548-964X.
- Viola, Paul dan J. Michaels. (2001). *Rapid Object Detection using boosted Cascade of Simple Features*. IEEE CVPR.

Rujukan Sumber *Online*:

- Edi. (2020). "Mengenal Raspberry Pi Komputer Papan Tunggal Seukuran Credit Card" <https://www.anakteknik.co.id/educationgood12/articles/mengenal-raspberry-pi-komputer-papan-tunggal-seukuran-credit-card> (Tanggal Akses: 09 Agustus, 2022).
- Rosebrock, Adrian. (2017) "Eye blink detection with OpenCV, Python, and dlib-PyImage Search". <https://www.pyimagesearch.com/2017/04/24/eye-blink-detection-opencv-python-dlib/>. (Tanggal Akses: 22 Mei, 2020).