

## Sistem Pendeteksi Kantuk Pengemudi berbasis *Eye Aspect Ratio* dan *Mouth Opening Ratio* menggunakan Algoritme C-LSTM

Auliya Firdaus<sup>1</sup>, Fitri Utamingrum<sup>2</sup>, Edita Rosana Widasari<sup>3</sup>

Program Studi Teknik Komputer, Fakultas Ilmu Komputer, Universitas Brawijaya  
Email: <sup>1</sup>auliyafirdaus@ub.ac.id, <sup>2</sup>f3\_ningrum@ub.ac.id, <sup>3</sup>editarosana@ub.ac.id

### Abstrak

Tercatat 103.645 kecelakaan lalu lintas di Indonesia pada tahun 2021, meningkat 3,6% dari tahun sebelumnya. Kendaraan penyebab kecelakaan kedua terbesar adalah angkutan barang dengan persentase 12%. Menurut Komite Nasional Keselamatan Transportasi Republik Indonesia (KNKT), 80% kecelakaan disebabkan oleh faktor kelelahan pengemudi yang menyebabkan *microsleep*. Untuk mengatasi masalah ini, diperlukan sistem deteksi dini kelelahan pengemudi. Sistem ini menggunakan *eye aspect ratio* (EAR) dan *mouth opening ratio* (MOR) sebagai parameter utama untuk mendeteksi *microsleep* dan menguap sebagai tanda kelelahan. Dengan adaptive threshold, akurasi sistem dalam mendeteksi *microsleep* sebesar 97%. Pendeteksian kondisi menguap pada sistem menggunakan model Convolutional Neural Network (C-LSTM). Model C-LSTM dipilih karena merupakan kombinasi antara CNN untuk pengenalan fitur yang lebih baik dan LSTM untuk pembelajaran sekuensial. Akurasi sistem deteksi menguap ini sebesar 98%. Dapat disimpulkan bahwa sistem ini dapat berjalan dengan baik dalam mendeteksi kelelahan pengemudi.

**Kata kunci:** Kelelahan Pengemudi, C-LSTM, Eye Aspect Ratio, Mouth Opening Ratio

### Abstract

There were 103,645 traffic accidents in Indonesia in 2021, an increase of 3.6% from the previous year. The second leading cause of accidents was freight transport, at a percentage of 12%. According to the National Committee for Transportation Safety of the Republic of Indonesia (KNKT), 80% of accidents were caused by driver fatigue, which resulted in *microsleep*. To address this problem, a system for early detection of driver fatigue is needed. This system uses the *eye aspect ratio* (EAR) and *mouth opening ratio* (MOR) as the main parameters for detecting *microsleep* and yawning as a sign of fatigue. With an adaptive threshold, the accuracy of the system in detecting *microsleep* is 97%. The system's detection of yawning uses a Convolutional Neural Network (C-LSTM) model. The C-LSTM model was chosen because it is a combination of CNN for better feature recognition and LSTM for sequential learning. The accuracy of the yawn detection system is 98%. It can be concluded that this system works well in detecting driver fatigue.

**Keywords:** Driver Drowsiness, C-LSTM, Eye Aspect Ratio, Mouth Opening Ratio

### 1. PENDAHULUAN

Data Korlantas Polri yang dipublikasikan Kementerian Perhubungan, kecelakaan lalu lintas di Indonesia pada tahun 2021 mencapai 103.645 kasus. Jumlah tersebut meningkat sebanyak 3.6% dari tahun sebelumnya. Angkutan barang merupakan penyebab kecelakaan kedua terbesar di Indonesia dengan persentase 12%. Komite Nasional Keselamatan Transportasi Republik Indonesia (KNKT) pada tahun 2021 menyebutkan sebanyak 80%

kecelakaan dipicu oleh faktor kelelahan pengemudi (*driver fatigue*) yang menyebabkan terjadinya *microsleep*.

Durasi mengemudi jangka panjang menyebabkan pengemudi mengalami kelelahan dan penurunan kondisi pengemudi secara signifikan. Lama waktu mengemudi yang akan menyebabkan kelelahan dan penurunan kondisi tubuh adalah 12-16 jam (Prabaswara, 2013). Kelelahan pengemudi menghasilkan dorongan untuk tidur untuk dapat memulihkan kelelahan yang dialami

sehingga timbulah rasa kantuk. Rasa kantuk dapat mengurangi reaksi, konsentrasi dan menurunkan kewaspadaan saat mengemudi kendaraan, sehingga dapat menyebabkan kecelakaan lalu lintas (Stutts, 2001).

Penelitian Geetavani (2021), membandingkan nilai *Eyes Aspect Ratio* (EAR) dan *Mouth Opening Ratio* (MOR) dengan nilai ambang batas (threshold) yang telah ditetapkan sebelumnya sebesar 0.2 hingga 0.3 (Dewi, 2022). Apabila terdapat penyimpangan terhadap nilai ambang batas awal dan nilai EAR mendekati 0 maka kantuk akan terdeteksi dan sistem akan memberikan peringatan. Namun, sebuah ambang batas (*threshold*) tidak bisa digunakan pada semua orang karena setiap orang memiliki nilai *Eyes Aspect Ratio* (EAR) yang berbeda secara genetik

Analisis kondisi menguap memerlukan analisis yang lebih kompleks variasi lebar dan durasi pembukaan mulut yang berbeda pada setiap orang. Performa tertinggi dari penelitian sebelumnya diperoleh oleh metode kombinasi antara CNN dan RCNN dengan akurasi sebesar 96.05% (Gite, 2021). Sebagian besar penelitian mengenai yawn detection berfokus pada klasifikasi citra pada video maupun gambar sehingga memerlukan waktu komputasi dan sumber daya sistem yang tinggi. Penelitian Khan (2020), mengekstraksi fitur *eye aspect ratio* (EAR), *mouth aspect ratio* (MAR), *mouth over eye aspect ratio* (MOE) dan *pupil circularity* (PC) menggunakan *Dlib's pretrained face detector*. Setelah melakukan ekstraksi, seluruh fitur digabungkan dan di train menggunakan CNN dan LSTM. Dengan melakukan ekstraksi fitur terlebih dahulu proses komputasi dan sumber daya sistem yang dibutuhkan lebih sedikit dibandingkan dataset gambar maupun video. Metode LSTM yang merupakan pengembangan dari RCNN juga terbukti memberikan performa yang baik pada data sekuensial. Pengembangan yawn detection pada penelitian ini akan menggunakan kombinasi antara CNN untuk ekstraksi fitur secara optimal dan LSTM untuk pembelajaran secara sekuensial. Data EAR dan MOR akan diekstraksi terlebih dahulu sebelum dilakukan proses training agar komputasi menjadi lebih ringan dan *compatible* untuk di tanamkan pada Raspberry Pi 4.

## 2. METODOLOGI

Tipe penelitian ini adalah implementatif pengembangan yaitu penelitian yang melakukan implementasi perangkat keras dan lunak, tidak hanya analisis data. Penelitian ini merupakan pengembangan dari penelitian yang telah dilakukan sebelumnya.



Gambar 1 Tahapan Metodologi Penelitian

### 2.1 Teknik Pengumpulan dan Analisis Data

Pengumpulan data dalam penelitian ini dilakukan dengan mengambil video pada 6 responden dengan kondisi menguap (*yawn*), terjaga (*alert*), dan berbicara (*talk*) selama masing-masing 1 menit. Selanjutnya akan dilakukan ekstraksi EAR dan MOR untuk dianalisis. Pada saat menguap terdapat pola perubahan pada EAR dan MOR yang selalu berbanding terbalik. Korelasi ini bisa didefinisikan sebagai *mouth over eyes ratio* (MOE). Total data yang didapatkan adalah 19.500 data dengan 3 input yaitu EAR, MOR, dan MOE dan 2 kelas yaitu *yawn* dan *no yawn*.



Gambar 2 Contoh Dataset

### 3. REKAYASA KEBUTUHAN

#### 3.1. Gambaran Umum Sistem

Sistem memiliki fungsi utama untuk mendeteksi kantuk pada pengendara dengan parameter *microsleep* dan menguap (*yawning*). Pendeteksian *microsleep* pada sistem menggunakan algoritme *adaptive thresholding* yang menghitung rata-rata pembukaan mata pada 50 frame pertama. Pendeteksian menguap (*yawning*) pada sistem menerapkan model C- LSTM yang merupakan kombinasi dari model CNN untuk ekstraksi fitur yang lebih baik dan LSTM untuk pembelajaran secara sekuensial. Apabila sistem mendeteksi *microsleep* atau menguap (*yawning*) maka buzzer pada sistem akan berbunyi.

#### 3.1. Rekayasa Kebutuhan Fungsional

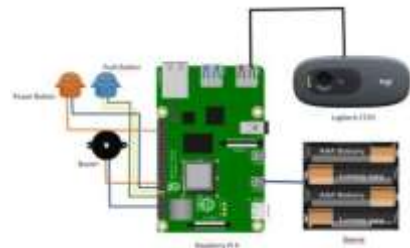
Berdasarkan gambaran umum sistem maka dapat dirumuskan kebutuhan fungsional sistem sebagai berikut,

1. Kamera pada sistem dapat menangkap citra wajah pengemudi untuk menjalankan algoritma facial landmark.
2. Sistem dapat mendeteksi *microsleep* (*microsleep detection*) berdasarkan analisis nilai eye aspect ratio (EAR).
3. Sistem dapat mendeteksi menguap (*yawning detection*) berdasarkan nilai mouth opening ratio (MOR) dan eye aspect ratio (EAR).
4. Sistem dapat menyalakan alarm berupa *active buzzer* saat kantuk pada pengemudi terdeteksi

### 4. PERANCANGAN DAN IMPLEMENTASI

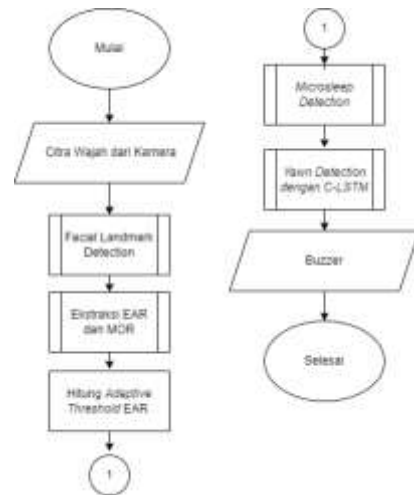
#### 4.1. Perancangan Perangkat Keras

Perangkat keras utama pada sistem terdiri atas kamera Logitech C270 sebagai input, Raspberry Pi 4 sebagai pemroses, dan *buzzer* sebagai output suara sistem. Blok Diagram dan Skematik lengkap sistem dapat dilihat pada Gambar 3.



Gambar 3 Skematik Sistem

#### 4.2 Perancangan Perangkat Lunak



Gambar 4 Flowchart Program Utama

Algoritme diagram alir program utama dijelaskan pada Gambar 4. Saat sistem pertama kali dinyalakan sistem akan mengambil citra wajah pengemudi dari kamera sebagai input. Kemudian akan dijalankan algoritme *Facial Landmark Detection* untuk menentukan titik-titik landmark pada mata. Setelah titik-titik landmark ditemukan akan dilakukan ekstraksi fitur EAR dan MOR yang merupakan fitur wajah utama yang akan digunakan dalam sistem. Sistem juga akan menghitung rata-rata EAR pada 50 frame pertama untuk dijadikan *threshold*. *Threshold* tersebut akan dijadikan sebagai acuan dalam *microsleep* detection pada sistem. Dalam melakukan yawn detection sistem akan menggunakan model C-LSTM yang telah dilatih sebelumnya. Apabila sistem mendeteksi adanya *microsleep* atau menguap pada pengemudi maka sistem akan

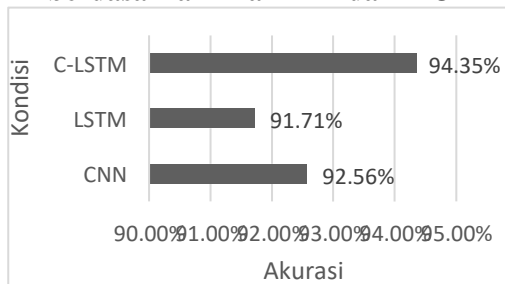
mengeluarkan output berupa *buzzer*.

## 5 PENGUJIAN DAN ANALISIS

### 5.1 Pengujian Model untuk *Yawn Detection*

#### *Detection*

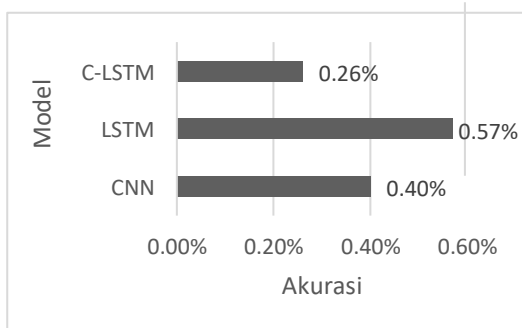
#### berdasarkan nilai EAR dan MOR



Tabel 3. Perbandingan Akurasi Model

Tabel 4. Perbandingan *Loss Function* Model

Dari data-data tersebut dapat disimpulkan bahwa metode C-LSTM memiliki performa akurasi paling tinggi dan loss function paling rendah dibandingkan kedua model lainnya. Pengujian efektivitas model yang dipilih dilakukan dengan membandingkan performanya dengan kedua model lain. Pengujian

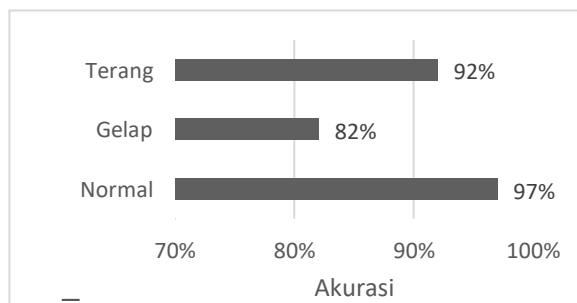


menggunakan 33 sampel kondisi menguap yang didapatkan dari 6 responden. Model CNN dapat memprediksi 30 sampel dengan benar dengan akurasi 90.2%. Model LSTM berhasil memprediksi 28 sampel dengan benar dengan akurasi 85.3%. Akurasi tertinggi didapatkan model C-LSTM sebesar 94.8% dengan prediksi benar sebesar 31 sampel. Dapat disimpulkan bahwa model C-LSTM dalam sistem dapat mengenali kondisi menguap dengan baik.

### 5.2 Pengujian Akurasi *Microsleep Detection*

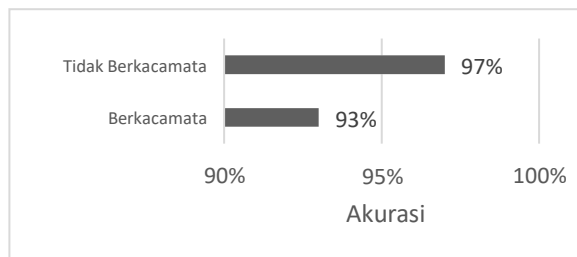
Tabel 5. *Microsleep Detection* dalam berbagai kondisi

pencahayaannya



Pengujian *microsleep detection* dalam berbagai kondisi cahaya menunjukkan penurunan performa model yang cukup signifikan yaitu sebesar 9% pada kondisi pencahayaan yang gelap. Hal ini disebabkan sistem kesulitan dalam mengenali fitur-fitur pada wajah dengan cahaya minim terutama fitur dengan ukuran kecil seperti mata.

Tabel 6. *Microsleep Detection* pada Objek Berkacamata



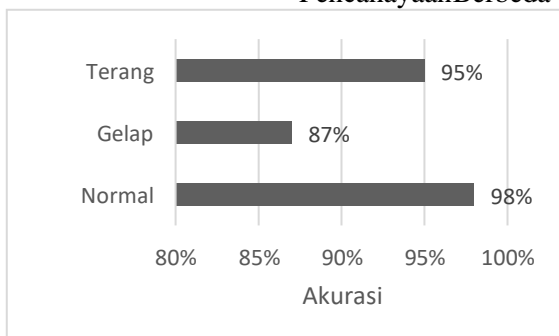
Perbandingan pengujian antara 3 responden berkacamata dan 3 responden tidak berkacamata menunjukkan hasil akurasi yang cukup baik. Terdapat penurunan akurasi sebesar 4% pada deteksi *microsleep* responden berkacamata dibandingkan responden tidak berkacamata. Penurunan ini tidak terlalu signifikan sehingga dapat disimpulkan bahwa sistem dapat berjalan dengan baik bagi pengguna dengan kacamata dengan akurasi 93%

### 5.3 Pengujian Akurasi *Yawn Detection*

Hasil pengujian *yawn detection* dalam berbagai kondisi pencahayaan menunjukkan hasil yang cukup baik. Sama seperti *microsleep detection* penurunan akurasi yang cukup signifikan juga timbul pada pencahayaan gelap karena sistem kesulitan dalam mengenali fitur pada wajah. Akurasi *yawn detection* dengan pencahayaan normal mencapai 98%,

pencahayaan terang sebesar 95%, dan pencahayaan gelap sebesar 82%.

Tabel 6. *Yawn Detection* pada Pencahayaan Berbeda



### 5.4 Pengujian Efisiensi Sistem

Pada analisis efisiensi sistem dihitung besar daya komputasi, kecepatan sistem dalam melakukan prediksi, dan penggunaan daya listrik. Pada pengujian daya komputasi dilakukan perbandingan antara jumlah sumber daya CPU dan RAM saat sistem berjalan dan idle. Dari data yang didapatkan terlihat peningkatan penggunaan CPU sebanyak 9 kali lipat dari 6% menjadi 56%. Pada RAM terjadi peningkatan sebanyak 4.6 kali lipat dari 278 MB menjadi 1304 MB.

Tabel 6. *Yawn Detection* pada Pencahayaan

	<i>Idle</i>	<i>Drowsiness Detection System (MB)</i>
<b>CPU</b>	6%	278
<b>RAM</b>	56%	1304

## 6 KESIMPULAN

Berbeda

klasifikasi sistem dalam membedakan kelas menguap dan tidak menguap (terjaga dan berbicara). Akurasi klasifikasi yang didapatkan sebesar 87%.

5. Dalam menjalankan sistem diperlukan rata-rata RAM sebesar 1304MB dan penggunaan CPU pada Raspberry Pi 4 sebesar 56%. Sistem dapat berjalan secara realtime dengan rata-rata waktu komputasi pendeteksian

Berdasarkan pengujian dan analisis yang telah dilakukan sebelumnya, maka penulis dapat menarik kesimpulan sebagai jawaban

atas rumusan masalah. Temuan-temuan dalam penelitian ini diantaranya:

1. Metode *adaptive threshold* terbukti memiliki performa yang lebih baik dibandingkan global threshold dalam menentukan nilai ambang batas eye aspect ratio (EAR) untuk pendeteksian *microsleep* dengan yang didapatkan dengan algoritma *adaptive threshold* mencapai 94%.
2. Model machine learning yang memiliki performa terbaik dalam yawning detection pada penelitian ini adalah model C-LSTM. Model ini dipilih untuk menggabungkan kelebihan dari model CNN dan LSTM. CNN digunakan untuk klasifikasi fitur yang lebih baik sedangkan LSTM digunakan untuk melakukan pembelajaran secara sekuensial. Model C-LSTM ini memiliki akurasi sebesar 94% dan loss function sebesar 0.26.
3. Dalam pengujian pencahayaan, sistem masih cenderung stabil dan bisa mempertahankan akurasinya pada saat cahaya terang ekstrim dengan akurasi sebesar 92%. Berbeda dengan pencahayaan gelap, sistem mengalami kesulitan dalam mendeteksi fitur pada mata sehingga akurasi hanya mencapai 82%. Pengujian *microsleep detection* pada pengguna berkacamata juga memiliki performa akurasi yang baik yaitu sebesar 93%.
4. Pengujian performa yawning detection dilakukan dengan pengujian sistem di berbagai kondisi pencahayaan juga. Pada pencahayaan normal akurasi sistem mencapai 98%, terang 95%, dan gelap 87%. Akurasi penguapan pada cahaya gelap lebih besar dibandingkan akurasi *microsleep* pada cahaya gelap dikarenakan fitur mulut yang lebih besar daripada mata sehingga lebih mudah dideteksi pembukaan dan penutupannya oleh sistem. Pengujian juga dilakukan untuk mengetahui performa facial landmark sebesar 0.3 detik dan waktu komputasi prediksi sebesar 0.002 detik.

## 7. DAFTAR PUSTAKA

- Puteri, R., & Utaminigrum, F. (2020). *Deteksi Kantuk Menggunakan Kombinasi Haar Cascade dan Convolutional Neural*

Network. <http://j-ptiik.ub.ac.id>

- Udayana, I. P. A. E. D., & Supartha, I. K. D. G. (2021). Implementasi Kombinasi Metode Mean Denoising dan Convolutional Neural Network pada Facial Landmark Detection. *Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI)*, 10(1), 1. <https://doi.org/10.23887/janapati.v10i1.29779>
- Savas, B. K., & Becerikli, Y. (2018, October 1). Real time driver fatigue detection based on SVM Algorithm. *2018 6th International Conference on Control Engineering and Information Technology, CEIT 2018*. <https://doi.org/10.1109/CEIT.2018.8751886>
- Kang, H. B. (2013). Various approaches for driver and driving behavior monitoring: A review. *Proceedings of the IEEE International Conference on Computer Vision*, 616–623. <https://doi.org/10.1109/ICCVW.2013.85>
- Sagonas, C., Tzimiropoulos, G., Zafeiriou, S., & Pantic, M. (t.thn.). *A semi-automatic methodology for facial landmark annotation*. Diambil kembali dari <http://ibug.doc.ic.ac.uk/>
- Christianni, N., & John, S.-T. (2000). *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press.
- Wang, L., & Springer, B. (2005). *Support Vector Machines: Theory and Applications*
- Bakheet, S., & Al-Hamadi, A. (2021). A framework for instantaneous driver drowsiness detection based on improved HOG features and naïve bayesian classification. *Brain Sciences*, 11(2), 1–15. <https://doi.org/10.3390/brainsci11020240>
- Dewi, C., Chen, R. C., Chang, C. W., Wu, S. H., Jiang, X., & Yu, H. (2022). Eye Aspect Ratio for Real-Time Drowsiness Detection to Improve Driver Safety. *Electronics (Switzerland)*, 11(19). <https://doi.org/10.3390/electronics11193183>
- Geetavani, B., Krishna Rao, N. v., Shalini, N., Abhishek Reddy, V., Amrutha, A., & Sultana, A. (2021). Drivers Fatigue Detection and Alarming Using Ear and MOR Values. *Proceedings of the 5th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud), I-SMAC 2021*, 1023–1029. <https://doi.org/10.1109/ISMAL2021.9640964>
- Geiger-Brown, J., Rogers, V. E., Trinkoff, A. M., Kane, R. L., Bausell, R. B., & Scharf, S. M. (2012). Sleep, sleepiness, fatigue, and performance of 12-hour-shift nurses. *Chronobiology International*, 29(2), 211–219. <https://doi.org/10.3109/07420528.2011.645752>
- Gite, S., Pradhan, B., Alamri, A., & Kotecha, K. (2021). ADMT: Advanced driver's movement tracking system using spatio-temporal interest points and maneuver anticipation using deep neural networks. *IEEE Access*, 9, 99312–99326. <https://doi.org/10.1109/ACCESS.2021.3096032>
- Jabbar, R., Shinoy, M., Kharbeche, M., Al-Khalifa, K., Krichen, M., & Barkaoui, K. (2020). *Driver Drowsiness Detection Model Using Convolutional Neural Networks Techniques for Android Application*. <http://arxiv.org/abs/2002.03728>
- Jacobé de Naurois, C., Bourdin, C., Stratulat, A., Diaz, E., & Vercher, J. L. (2019). Detection and prediction of driver drowsiness using artificial neural network models. *Accident Analysis and Prevention*, 126, 95–104. <https://doi.org/10.1016/j.aap.2017.11.038>
- Jie, Z., Mahmoud, M., Stafford-Fraser, Q., Robinson, P., Dias, E., & Skrypchuk, L. (2018). Analysis of yawning behaviour in spontaneous expressions of drowsy drivers. *Proceedings - 13th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2018*, 571–576. <https://doi.org/10.1109/FG.2018.00091>
- Kuwahara, A., Nishikawa, K., Hirakawa, R., Kawano, H., & Nakatoh, Y. (2022). Eye fatigue

estimation using blink detection based on Eye Aspect Ratio Mapping(EARM). *Cognitive Robotics*, 2, 50–59.

<https://doi.org/10.1016/j.cogr.2022.01.003>

Luyster, F. S., Strollo, P. J., Zee, P. C., & Walsh, J. K. (2012). Sleep: A health imperative. In *Sleep* (Vol. 35, Issue 6, pp. 727–734). <https://doi.org/10.5665/sleep.1846>

Muhammad Fahad Khan, F. A. (2012). *Efficient Car Alarming System for Fatigue Detection during Driving*. <http://www.ijimt.org/papers/280-Z060.pdf>

Shi, M., Wang, K., Li, C., & IEEE International Conference on Computer and Information Science (ICIS) 2019. (n.d.). *A C-LSTM with Word Embedding Model for News Text Classification*.

Stutts, J. C., Wilkins, J. W., Osberg, J. S., & Vaughn, B. v. (2003). Driver risk factors for sleep-related crashes. *Accident Analysis and Prevention*, 35(3), 321–331. [https://doi.org/10.1016/S0001-4575\(02\)00007-6](https://doi.org/10.1016/S0001-4575(02)00007-6)

Teive, H. A. G., Munhoz, R. P., Camargo, C. H. F., & Walusinski, O. (2018). Yawning in neurology: A review. In *Arquivos de Neuro-Psiquiatria* (Vol. 76, Issue 7, pp. 473–480). Associacao Arquivos de Neuro-Psiquiatria. <https://doi.org/10.1590/0004-282x20180057>

Zhang, W., Cheng, B., & Lin, Y. (2012). Driver Drowsiness Recognition Based on Computer Vision Technology \*. In *TSINGHUA SCIENCE AND TECHNOLOGY ISSN11007* (Vol. 17, Issue 3).