Smart Cabin Application - Driver Health Monitoring System based on ML

Gowrishankar S[1], Jyothi A P[2]

[1]Ph.D Scholar, M. S. Ramaiah University of Applied Sciences, Bengaluru, Karnataka, India
Email: *shankar.iq[at]gmail.com*
22ETRP729003[at]msruas.ac.in

[2]Assistant Professor, M. S. Ramaiah University of Applied Sciences, Bengaluru, Karnataka, India
Email: *jyothiarcotprashant[at]gmail.com*
jyothi.cs.et[at]msruas.ac.in

Abstract: ***To avoid Road accidents caused due to medical emergency conditions, Proposing on Smart cabin application using Machine Learning. Introducing safety feature into Automotive vehicles to avoid Road accidents caused due to medical emergency conditions. Using ML (Machine Learning) to monitor health status of driver and provide audio/voice announcement warning in case of health issues identified. Health issues can be identified are Cardiac issue Identification and Corona symptoms Identification. Design, Implement, Simulation and Testing the Automotive Safety feature with respect to Health monitoring system in Cabin during Driving. Design to collect the Drivers Data & Develop/Utilize Machine Learning Algorithms to monitor health status of driver and predict Health issues using the past Driver Training dataset. Implement audio/voice announcement warnings in case of health issues identified. Integration of Developed solution and testing.***

Keywords: Sensors, Adaptive Autosar, Machine learning, RCar-H3e, Raspberry Pi, Android, QUALCOMM 8295

1. Introduction

As per Govt of India-Ministry of Road Transport & Highways Transport Research Wing-New Delhi, In India 4,12,432 accidents reported in 2021 Year, 1,53,972 Deaths & 3,84,448 major Injuries reported. Age profile of Fatal Road accident victims is 35-60 years during 2021 it is about 38.2%. In India : From 2017 to 2021, there was a 22% increase in death due to heart attacks. In the last 10 years, from 2012 to 2021, there had been a 54% increase in heart attack deaths, according to the NCRB (National Crime Records Bureau). Road Accidents & Heart Attach accounts 30% of death in India every year. Around 21 cases of death caused during driving during 2015 to 2019. Age between 35-60 years is more prone to cardiovascular diseases. According to World Health Organization(WHO) more than 25 million people killed worldwide annually in road accidents. National Highway Traffic Safety Administration (NHTSA) says 60% of road accidents are due to driver distraction during driving.

Distracted driving has become a dominant cause of traffic accidents. Major reason for driver distractions are stress, fatigue, drowsiness and underlying health issues of driver such as Cardiac issues [6][12]. To research and propose on Health Monitoring system in Automotives that helps in reducing the road accidents and saving life in case of Cardiac related emergencies[1][3][8]. ML based driver's health monitoring system capable of predicting medical emergency situation can significantly reduce accidents due to diver distraction [3][4].

Prediction of Cardiac issue & Warnings voice messages during driving, based on the Driver Heart Beats, Body Temperature, SPO2(Saturation of Peripheral Oxygen), Hand Pressure and past driver Training Dataset use Machine Learning Algorithm and predict if any Cardiac related issues. audio/ voice announcement: ECG(Electrocardiogram) warning, Slow down, park the vehicle, visit the nearest hospital.

Prediction of Corona disease & Warnings voice messages during driving, based on the Driver Body Temperature, Heart Beats, SPO2 and past driver Training Dataset use Machine Learning Algorithm and predict if Corona related symptoms are observed. audio/voice announcement: Isolate yourself and visit the nearest hospital.

This paper introduces a smart cabin Driver Health Monitoring Application over HPC (High performance computing) that detects Cardiac related issues and Corona Symptoms the allows drivers to avoid Road accidents in emergencies based on Machine learning without driver intervention based on their physiological data. Figure.1 represents the Driver Health Monitoring application and Figure.2 represents the use case scenario.

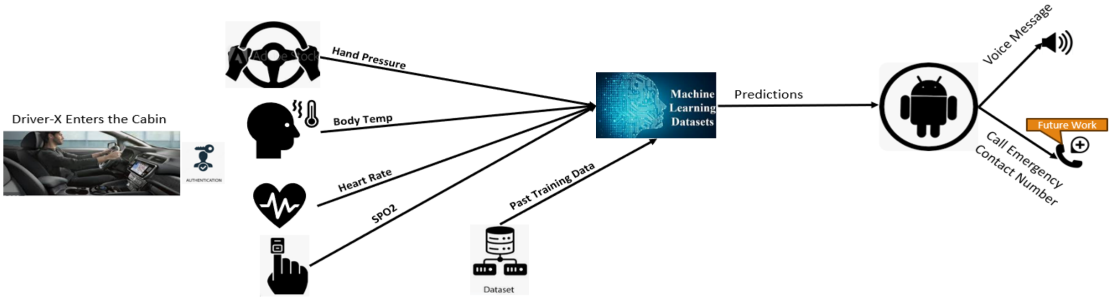

Figure 1: Smart cabin- Driver Health Monitoring application.

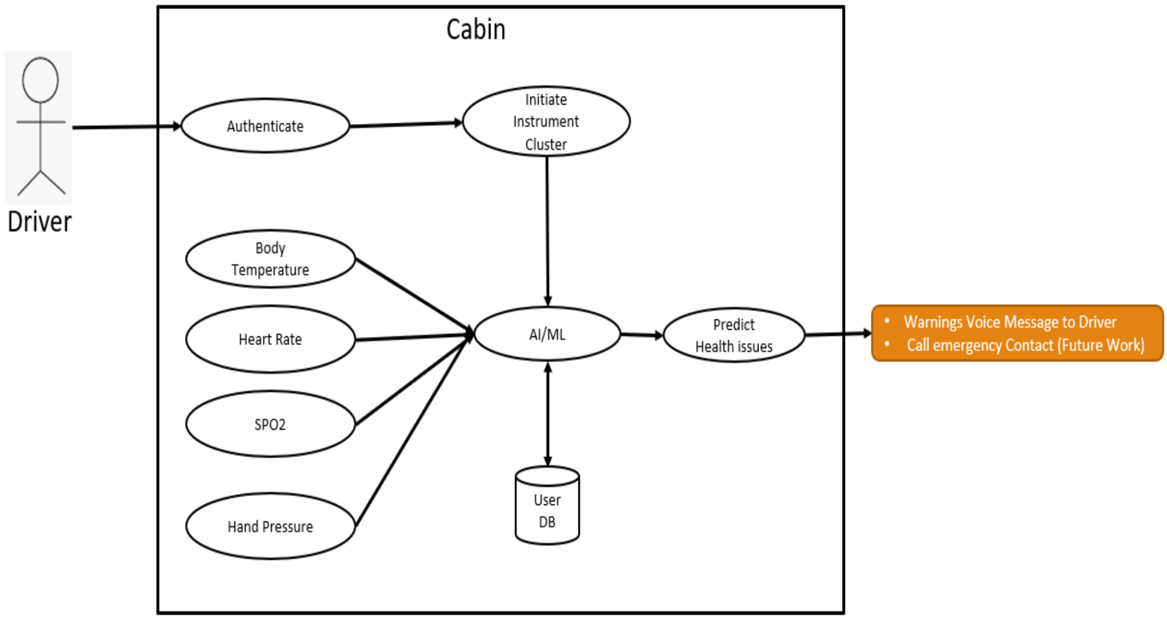

Figure 2: Flow Chart of the Use case Scenario.

This paper is structured as follows: Section 2 on adaptive Autosar (Automotive Open System Architecture) overview, Section 3 presents the experiments conducted for selecting appropriate machine learning algorithms while Section 4 presents the H/W implementation of application with results followed by conclusion and future scope.

2. Adaptive Autosar Architecture

Adaptive AUTOSAR is a new standard for automotive software architecture that is designed to address the challenges of future vehicles, such as autonomous driving and connectivity. It provides a more flexible and scalable platform for developing and deploying software, and it supports the use of high-performance computing (HPC) architectures [5][13].

Here are some of the key features of Adaptive AUTOSAR: Service-oriented architecture (SOA) [14][15]. Adaptive AUTOSAR is based on a service-oriented architecture, which allows for the independent development and deployment of software components. This makes it easier to update and upgrade software, and it also allows for greater scalability. High-performance computing (HPC). Adaptive AUTOSAR supports the use of HPC architectures, which are necessary for running complex and demanding applications such as autonomous driving[10]. Flexibility. Adaptive AUTOSAR is a flexible platform that can be adapted to the specific needs of different vehicles. This makes it a good choice for both traditional and next-generation vehicles.

Here are some of the benefits of using Adaptive AUTOSAR: Increased flexibility and scalability[9]. Adaptive AUTOSAR's service-oriented architecture and support for HPC architectures make it a more flexible and scalable platform than Classic AUTOSAR. This makes it easier to update and upgrade software, and it also allows for greater scalability. Improved performance. Adaptive AUTOSAR's support for HPC architectures can help to improve the performance of automotive software. This is important for applications such as autonomous driving, which require high-performance computing. Reduced development costs. Adaptive AUTOSAR's standardized development methodology can help to reduce development costs. This is because it provides a common framework for developing and deploying software, which can help to improve efficiency.

Overall, Adaptive AUTOSAR is a new standard that offers a number of advantages over Classic AUTOSAR. It is a more flexible, scalable, and performant platform that can help to reduce development costs. As a result, it is a good choice for vehicles that require high-performance computing and flexibility[11].

Here are some of the differences between Adaptive AUTOSAR and Classic AUTOSAR: Architecture. Adaptive AUTOSAR is based on a service-oriented architecture (SOA), while Classic AUTOSAR is based on a signal-based architecture. Target platforms. Adaptive AUTOSAR is designed for high-performance computing (HPC) platforms, while Classic AUTOSAR is designed for embedded platforms. Development methodology. Adaptive AUTOSAR has a standardized development methodology, while Classic AUTOSAR does not. Applications. Adaptive AUTOSAR is well-suited for applications that require high performance and flexibility, such as autonomous driving [10]. Classic AUTOSAR is well-suited for applications that require high real-time performance, such as safety-critical applications. Figure. 3 represents the Adaptive Autosar Architecture diagram [18].

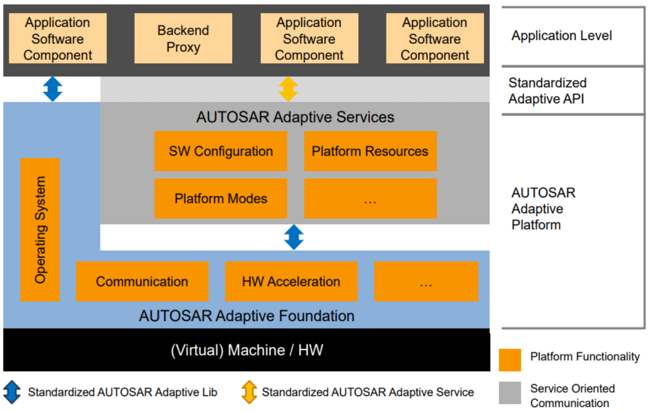

Figure 3: Architecture of the AUTOSAR Adaptive Platform

3. Selection of Machine Learning Algorithm

We have developed an Driver Health Monitoring system with Health issue prediction module. The predictor considers the variables Hand Pressure on Steering, Body Temperature, Heart Rate and SPO2 (Saturation of Peripheral Oxygen) to predict the target variable i.e. Cardiac issue & Corona symptom. The methodologies incorporated to develop the module is described in the further sections. For initial database creation, a set of 10000 data points of Hand Pressure on Steering, Body Temperature, Heart Rate and SPO2 from Kaggle [17] which is a data science community containing variety of datasets. As there were no direct dataset available matching out requirement we looked for different categories where these variables were individually available and then combined them together as per our requirement[16].

Body Temperature, Heart Rate and SPO2 are the inputs to identify the corona symptom and in addition to above data Hand Pressure is introduced to predict the Heart related issues. Once all the data was converted in preferred format these data were transferred to a database (.db) file using python's sqlite3 and pandas package. Pandas provide the access for file and Sqlite3 package allows to transfer contents of a .csv file into database file. This database file will then be used in further stages to train the prediction model. Finding the relationship between variables used for prediction helps in knowing whether they have a strong relationship with each other [2]. For example, if two variables have a strong relationship between each other, then, one out of the 2 variables would be enough to predict the output variable. It also proves useful in finding which algorithm could possibly be used for prediction. For example, if the variables are linearly related then linear regression could be used for prediction. In order to find the association between data present in the data points collected by us we calculated covariance and correlation coefficients between the independent variables and the data points.

Relation	HP	BT	HR	SPO2
Covariance	0.236	-0.1201	0.236	1.7399
Pearson's correlation	0.499(Moderate)	-0.096(Weak)	0.158 (Weak)	0.447(Moderate)
Spearman's correlation	0.54(Moderate)	-0.11(Weak)	0.2686(Weak)	0.44(Moderate)

As seen in the above table, we didn't find any significant linear relationship between the variables and hence using an algorithm like linear regression was ruled out at this stage.

To further identify any possible relationship between the independent and target variable we plotted the data points to check if there's a possibility of probably an exponential relationship between the data points. Package used : matplotlib for plotting data.

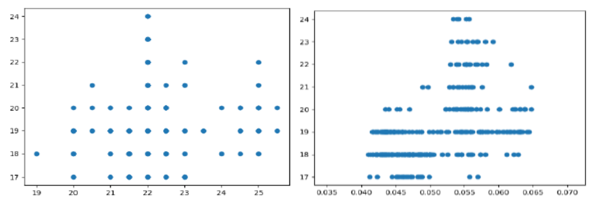

Figure 4: Sklearn library's preprocessing for Hand Pressure and Body Temperature

As seen in the above plots, there was no linear, nonlinear or exponential relationship that could be derived out of the data points. Hence, we tried to check if normalizing the data could be of help in bringing out any relation between the data variables.

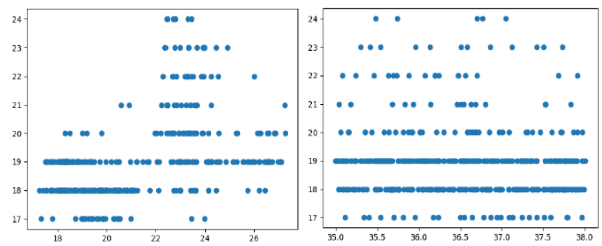

Figure 5: Sklearn library's preprocessing Heart Rate and SPO2.

Using the Python library Scikit learn package and other packages tested different algorithms since the relationship cannot be derived from above tests. below listed of algorithms are tested [2].

a) Support vector Machine (SVM)
b) Decision Tree Regressor
c) Random Forest Regressor
d) Polynomial Regressor
e) Adaboost Regressor

f) XGBOOST

Dataset splitted into Train & Test data with 80:20, 70:30, 90:10 and Tested these algorithms.

We used Mean Square Error (MSE) and Mean Absolute Error (MAE) statistics to find the errors in predictions. Mean absolute error calculates an average of differences between predicted and expected value, gives an idea about how wrong the predictions are. MAE minimum or value equal to zero i.e. MAE= 0 indicates perfect prediction. Similar to MAE , Mean squared error takes also squares the differences between the actual and predicted value and lower value of MSE gives better the predictions with less outliers. We selected algorithms having the lower value of MSE and MAE.

Input	Algorithm	Output
Hand Pressure	Decision Tree	Cardiac Issue
Body Tempaerature	Adaboost	Cardiac Issue
Heart Rate	Polynomial	Cardiac Issue
SPO2	Decision Tree	Cardiac Issue

Input	Algorithm	Output
Body Tempaerature	Adaboost	Corona Symptom
Heart Rate	Polynomial	Corona Symptom
SPO2	Decision Tree	Corona Symptom

As mentioned in above table, 4 Algorithms are used for the Cardiac Issues predictions and 3 Algorithms for the Corona symptom identifications.

Improvement of Accuracy

To improve the accuracy adopted with K-fold cross validation. The dataset is divided into K-folds. Then the entire dataset (10000) points will be divided into 10 folds i.e k=10, roughly 1000 values per fold which will be used as test data and the remaining values would be used to train the model. By this method, the prediction model will be able to train, and test will all possible sections of the data. And hence, we would have a better view of how the prediction module behaves with different kinds of test and training data. With Sklearn library's KFold module for diving data into K folds and Sklearn library's mean absolute error and mean squared error modules are used to find MAE and MSE.

Once all the blocks were developed, we modularized the code into 3 separate modules.

1) Cardiac_prediction_module
2) Corona_prediction_module
3) Prediction_finalization_module

From Figure. 6 represents the accuracy as below, Corona Symptom prediction accuracy of 97%, Cardiac Issues prediction accuracy of 93.4%.

Cardiac Issues Prediction

Number of values in the training database	Number of test values?	Are these test values new to the prediction module?	Did database already contain these values?	Deviation +/- 2	Deviation +/- 3	Deviation +/- 4	Deviation +/- 5	Deviation +/- 6	Accuracy percentage	Accuracy percentage with dev +/2
10000	160	Yes	No	25	13	5	0	0	26.4	67.4
10000	10000	Yes	No	47	14	14	6	1	76.4	89.4
10000	5000	No	Yes	55	12	2	0	0	79.2	93.4
10000	5000	Yes	No	67	21	9	5	2	75.6	87.04
10000	2000	No	No	62	21	8	2	3	72.3	88.59
10000	3000	No	Yes	52	23	16	5	0	71.13	85.989

Corona Prediction

Number of values in the training database	Number of test values?	Are these test values new to the prediction module?	Did database already contain these values?	Deviation +/- 2	Deviation +/- 3	Deviation +/- 4	Deviation +/- 5	Deviation +/- 6	Accuracy percentage	Accuracy percentage with dev +/2
10000	160	Yes	No	25	13	5	0	0	30.6	74.9
10000	10000	Yes	No	47	14	14	6	1	79.2	92.91
10000	5000	No	Yes	55	12	2	0	0	84.6	97
10000	5000	Yes	No	67	21	9	5	2	73.4	90.3
10000	2000	No	No	62	21	8	2	3	77.1	93.34
10000	3000	No	Yes	52	23	16	5	0	76.8	90.66

Figure 6: Reliabilty check with two sets of 10000 datapoints.

4. Implementation of Driver Health Monitoring System application

Driver Health Monitoring system implementation is divided to two platforms, MAX30102 sensor for Body Temperature, Heart Rate, SPO2 connected to Raspberry Pi 4 through I2C (Inter-Integrated Circuit) protocol, Robodo SEN38 sensor for Hand Pressure connected to ADS1115 ADC and connected with Raspberry Pi 4 through I2C protocol for data acquisition.

Data acquired from RaspBerry Pi using two sensors are transmitted to Renesas R-CAR H3e Board through TCP I/P (Transmission Control Protocol/Internet Protocol). R-Car H3e board ported with Adaptive Autosar hosting Machine learning algorithms performs as High computing machine. The final Machine learning predicted Health issue i.e warning messages will be sent to another Android System called IVI (in-vehicle infotainment) ADP-Qualcomm 8295 Android device with Android-Q version running device for announcement of alert message. The Platform Interaction and High-level hardware interaction diagrams are shown in Figure 7 and 8 respectively.

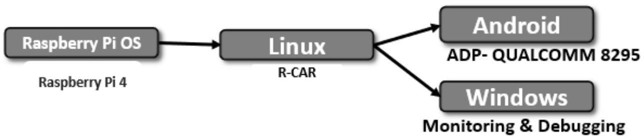

Figure 7: Platform Interactions

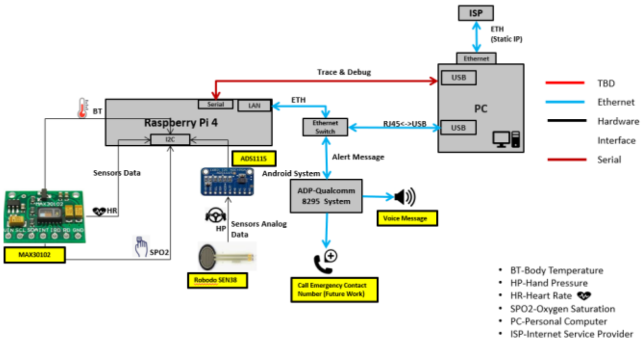

Figure 8: High level Hardware interaction diagram.

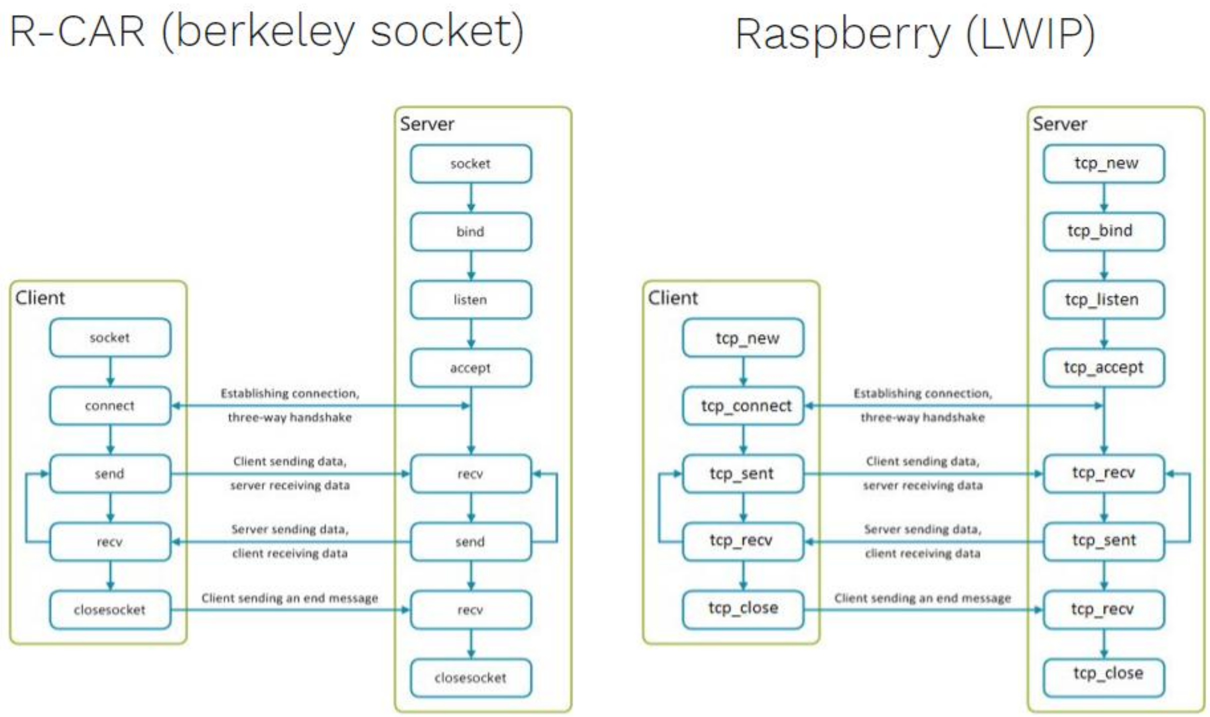

Figure 9: TCP/IP communication mechanism

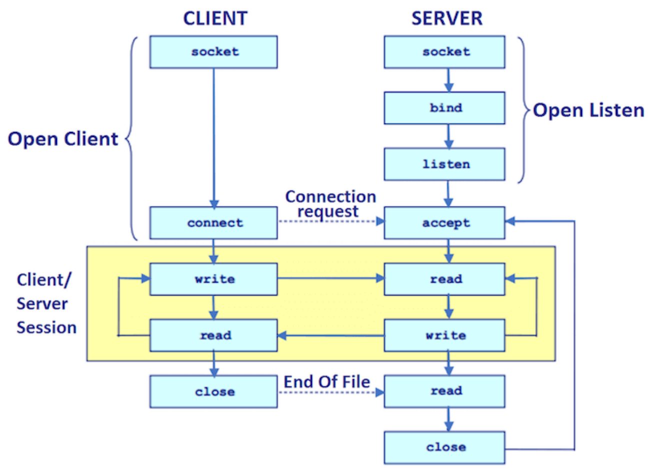

Figure 10: Android IVI - socket programming over TCP/IP communication mechanism.

TCP/IP communication scheme is elaborated in Fig 10 & 11. Communication starts with socket creation using Port and IP (Internet Protocol address) and on successful connection establishment with acknowledgement, RCAR will Receive the data continuously from Raspberry Pi, Once after prediction using ML with respective message is prepared, Alert message will be communicate to ADP-Qualcomm 8295 Android IVI system using TCP/IP communication with same socket communication system.

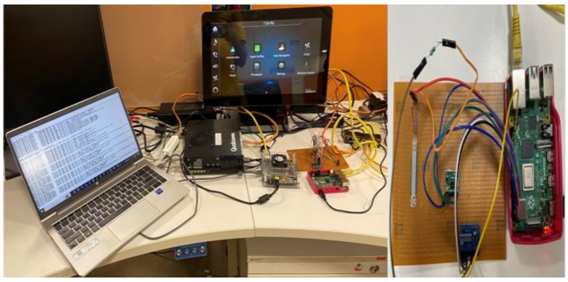

Figure 11: Driver Health Monitoring demo setup

Acknowledgment

I would like to take this opportunity to express our gratitude to M. S. Ramaiah University of Applied Sciences Computer Science Department staff for the support during research work and KPIT technologies for providing the Environment setup mainly on R-Car and IVI environment setup support throughout this research.

I would like to take this opportunity to express my gratitude to Jyothi A P - Asst. Professor of Computer Science Department of M. S. Ramaiah University of Applied Sciences for her exceptional support and guidance throughout this research and development as well as for giving us the chance to work under her guidance.

5. Conclusion and Future Scope

We have showcased latest Adaptive Autosar technologies in Automotive HPC machine by developing Driver Health Monitoring System Application, Integration of different platform is also achieved. ML based application predicts Issues accurately. We can extend it with using SOA SOME/IP (Scalable Service-Oriented Middleware over IP) or DDS (Data Distribution Service) based communication between R-CAR and ADP-Qualcomm 8295 Android System and introduce cloud computing to improve accuracy. Introducing the Emergency Number contact calling mechanism for this design will be adding good feature [7]. Implementing it on a real car is the immediate future work in this field with diverse occupants setting to understand the effects of certain extremes conditions and improve ML predictions.

References

[1] Prasanta Kumar Sahoo, Pravalika Jeripothula, "Heart Failure Prediction Using Machine Learning Techniques", doi: 10.2139/ssrn.3759562 December 2020.
[2] Gowrishankar S. Sreevishakh KP. Shruthe R. "Smart Cabin Application on High Performance Computing Machine – AC personalisation", doi : 10.21275/SR23710054300, vol-12, July 2023.
[3] Avelino J. Gonzalez a, Josiah M. Wong , "Detection of driver health condition by monitoring driving behavior through machine learning from observation". doi : 10.1016/j.eswa.2022.117167. vol:199, August 2022.
[4] Vidhya Balakrishnan, Sibhia J, Dr. G. Mohanraj, "Driver Health Monitoring System". doi: 10.22214/ijraset.2022.40841, vol 10, April 2022.
[5] S Fürst, M Bechter, "AUTOSAR for connected and autonomous vehicles: The AUTOSAR adaptive platform", doi : 10.1109/DSN-W.2016.24, July 2016.
[6] Sihem NITA, Salim BITAM, Abdelhamid MELLOUK, " A Body Area Network for Ubiquitous Driver Stress Monitoring based on ECG Signal ", 2019 International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob).
[7] Marius Minea et al, " Non-Intrusive Driver Condition Monitoring in Highly Automated Vehicles with Medical Information Support for Emergency Calling", University Politehnica of Bucharest, Transports Faculty Bucharest, Romania.
[8] Savita Patil, "In-Vehicle Driver Health Monitoring System" vol-3, 2016.
[9] Mia Stepanovic, Milan Bjelica, Ivan Kastelan, Gordana Velikic, " Scalable approach to extending automotive software using AUTOSAR Adaptive stack ",University of Novi Sad Novi Sad, Serbia.
[10] Simon Fürst, Dr.-Ing Markus Bechter," AUTOSAR for Connected and Autonomous Vehicles The AUTOSAR Adaptive Platform", BMW Group Munich, Germany.
[11] Gopu G.L. Kavitha K.V. James Joy, " Service Oriented Architecture based connectivity of Automotive ECUs ", 2016 International Conference on Circuit, Power and Computing Technologies [ICCPCT].
[12] Branko Babusiak. "Design of Smart Steering Wheel for Unobtrusive Health and Drowsiness Monitoring" doi : 10.3390/s21165285, August 2021.
[13] Robert Šandor, Mia Stepanović, " Vehicle2X communication proposal for Adaptive AUTOSAR " , 2018 IEEE 8th International Conference on Consumer Electronics - Berlin (ICCE-Berlin).
[14] M. Wagner, D. Zöbel and A. Meroth. "Towards an Adaptive Software and System Architecture for Driver Assistance Systems". in Proc. of the 4th IEEE International Conference on Computer Science andInformation Technology, 2011, vol. 4 pp. 174-178.
[15] M. Wagner, D. Z¨obel, and A. Meroth, "SODA: Service-oriented Architecture for runtime adaptive Driver Assistance Systems," in Proceedings of the 17th IEEE Computer Society symposium (ISORC). Reno, NV: IEEE Computer Society, 2014.
[16] J. Singh, R. Kanojia, R. Singh, R. Bansal, and S. Bansal, "Driver drowsiness detection system: An approach by machine learning application," arXiv [cs.LG], 2023. Accessed: Jul. 12, 2023. [Online]
[17] https://www.kaggle.com/datasets/rishanmascarenhas/covid19-temperatureoxygenpulse-rate?resource=download
[18] http://www.autosar.org
[19] https://datasheets.raspberrypi.com/rpi4/raspberry-pi-4-datasheet.pdf
[20] https://www.analog.com/media/en/technical-documentation/data-sheets/max30102.pdf
[21] https://electrosome.com/wp-content/uploads/2014/12/Force-Sensitive-Resistor-Datasheet.pdf
[22] https://cdn-shop.adafruit.com/datasheets/ads1115.pdf

Author Profile

Gowrishankar Shivashankara chari completed his Master degree in 2021 from BITS Pilani and Bachelor degree in 2006 from Visvesvaraya Technological University, India. He is currently pursuing his Doctoral degree in the domain of Artificial Intelligence at the Department of Computer Science and Engineering, Ramaiah University of Applied Sciences, Ramaiah Technology Campus, Bengaluru, Karnataka, India. He has 17 years of Industry experience and working as Solution Architect. His research interest is in the field of Machine Learning, Deep Learning, AI and Cloud Computing. He can be contacted at email: shankar.iq@gmail.com.

Jyothi Arcot Prashant completed her PhD in 2020, Master degree in 2009, Bachelor degree in 2002 from Visvesvaraya Technological University, India. She is currently working as Faculty of Engineering and Technology, Department of Computer Science and Engineering, Ramaiah University of Applied Sciences, Ramaiah Technology Campus, Bengaluru, Karnataka, India. She has 16 years of experience in teaching and has published many research papers in journals indexed in SCI/SCIE, WOS, SCOPUS etc. and presented papers in several National and International conferences. Her research interest is in the field of Wireless sensor network, IOT, Embedded systems, AI, ML and Deep Learning. She can be contacted at email: jyothi.cs.et@msruas.ac.in.