



IKER
GAZTE
NAZIOARTEKO
IKERKETA EUSKARAZ

VI. IKERGAZTE

NAZIOARTEKO IKERKETA EUSKARAZ

2025eko maiatzaren 28, 29 eta 30a
Bilbo, Euskal Herria

ANTOLATZAILEA:
Udako Euskal Unibertsitatea (UEU)



Aitortu-PartekatuBerdin 4.0

INGENIARITZA ETA ARKITEKTURA

**Adimen Artifiziala Erosotasunera
Bideratuta: Zorabioaren
Iragarpena Sistema Txertatu
batean**

*Jon Ander Ruiz Colmenares,
Óscar Mata-Carballeira
eta Estibaliz Asua Uriarte*

101-108 or.

<https://dx.doi.org/10.26876/ikergazte.vi.03.12>

ANTOLATZAILEA



BABESLEAK



EUSKO JAURLARITZA
GOBIERNO VASCO



LAGUNTZAILEAK



Adimen Artifiziala Erosotasunera Bideratuta: Zorabioaren Iragarpena Sistema Txertatu batean

Jon Ander Ruiz Colmenares¹, Óscar Mata-Carballeira¹ eta Estibalitz Asua Uriarte¹

Elektrizitate eta Elektronika Saila, Zientzia eta Teknologia Fakultatea, UPV/EHU, 48940 Leioa, Espainia.

jonander.ruiz@ehu.eus

Laburpena

Lan honek sistema txertatu batean zorabioa iragartzeko sistema baten garapena aztertzen du. Sistema, ESP32 mikrokontrolagailu batean oinarritzen da, eta ibilgailuaren CAN busetik, GPS eta IMU sentoreetatik datuak eskuratzen ditu denbora errealean. Eskuratutako datuak, CatBoost ikaskuntza automatikoko eredu baten bitartez prozesatzen dira, zorabioaren probabilitatea iragartzeko. Eredua optimizatu da baliabide mugatuko sistema txertatuetan eraginkortasunez implementatzeko, ereduaren kudeaketa dinamikoa eta BIN formatua erabiliz. Emaitzek sistema eraginkorra dela erakusten dute, inferentzia-denbora azkarra eta memoriaren erabilera optimizatua lortuz. Etorkizuneko lanetan, eragingailuak integratzea eta ereduaren errendimenduaren hobetzea izango dira helburuak.

Hitz gakoak: Mikrokontroladoreak, Adimen artifiziala, Erosotasuna, Gidariari Laguntzeko Sistema Aurreratuak

Abstract

This work explores the development of a motion sickness prediction system on an embedded platform. The system is based on the ESP32 microcontroller and acquires real-time data from the vehicle's CAN bus, GPS, and IMU sensors. The acquired data is processed using a CatBoost machine learning model to predict the probability of motion sickness. The model is optimized for efficient implementation on resource-constrained embedded systems, employing dynamic model management and the BIN format. The results demonstrate that the system is effective, achieving fast inference times and optimized memory usage. Future work will focus on the integration of actuators and the improvement of model performance.

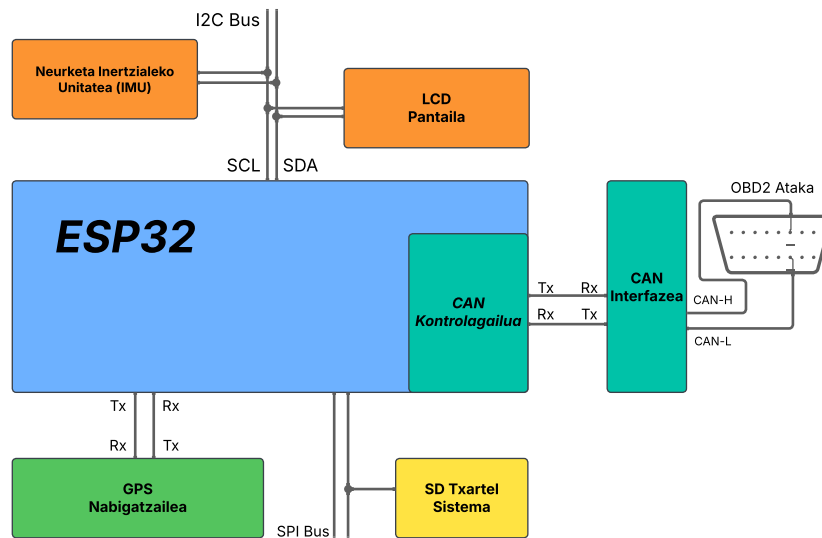
Keywords: Microcontrollers, Machine Learning, Confort, Advanced Driver Assistance Systems

1 Sarrera eta motibazioa

Sistema txertatuak, gaur egun, ingeniartzaren eta zientziaren arlo askotan ezinbestekoak dira (Gill et al. (2024)), aplikazio sorta zabal baterako soluzio eraginkor eta merkeak eskainiz (El-Khozondar et al. (2024), Yadav Kanneboina et al. (2024), Khan et al. (2021), Devi et al. (2023)). Horien artean, mikrokontrolagailuak nabarmentzen dira, hala nola ESP32 bezalakoak, errendimendu handiko prozesatze-ahalmena eta konektibitate integrala uztartzen baitituzte, Programmable System-on-a-Chip (PSoC) arkitektura konplexuagoak, Xilinx ZynQ plataforma bezalakoak, besteak beste, alde batera utzi gabe. Gailu hauek, gero eta gehiago erabiltzen dira funtzionalitate adimendunak txertatzeko, automobilgintzaren sektorea adibide argi bat izanik.

Izan ere, automobilgintzan, Gidariei Laguntzeko Sistema Aurreratuak (ADAS) gero eta konplexuagoak bihurtzen ari dira, denbora errealeko datu prozesamendua eta ikaskuntza automatikoko eredu aurreratuak eskatuz (Samantaray (2023), Mandumula et al. (2023)). Sistema hauek, gidatzea errazteaz gain, bidaiarien segurtasuna eta erosotasuna hobetzeko dute helburu, azken hau, ibilgailu autonomoen garapenarekin batera gero eta garrantzi handiagoa hartzen ari delako. Hala ere, erosotasunaren alorrean, oraindik gaizto beharreko beste erronka garrantzitsu bat dago: zorabioa.

Zorabioa, ibilgailuetan bidaiatzean sortzen den ondoreza, ezaguna da, eta, neurri handi batean, sentazio subjektiboa bada ere, badaude sentazio hauek objektiboki aztertu eta ebaluatzeko aukera ematen diguten estandarrak, hala nola ISO2631-1 estandarrean eta *Human Vibration* Eskuliburuan zehazten diren metrikak, besteak beste (ISO 2631-1:1997 (1997), Griffin (2012)). Hau kontuan hartuta, lan honetan, gidariei laguntzeko sistema baten oinarria



1. Irudia: Hardware sistema osoa

diseinu eta garapen-prozedura proposatzen da, zorabioarekin loturiko metrikak denbora errealean iragartzeko gai dena.

Sistema honek, kostu baxua, erreproduzigarritasuna eta ibilgailuan modifikaziorik gabeko inplementazioa ditu ezaugarri, eta ESP32 mikrokontrolagailuan oinarritzen da. Hardwarearen alorrean, CAN eskuratzeko-sistema, GPS eta IMU moduluen integrazioa, eta SD txartel zein pantaila baten kudeaketa aztertuko dira. Softwareari dagokionez, denbora errealean egikaritzen den datu-katea eta zorabioarekin erlazionatutako metrikak iragartzeko eredu baten inplementazioa azalduko dira. Laburbilduz, lan honen helburua zorabioa arintzeko eta gidatze-esperientzia hobetzeko erabil daitekeen gidariei laguntzeko sistema baten oinarriak ezartzea da.

2 Ikerketaren Muina

2.1 Hardware sistemak

Atal honetan, sistema osatzen duten hardware elementu nagusiak aurkeztuko dira. Lehenik, sistema osoaren diagrama blokea aztertuko dugu, osagai nagusiak eta haien arteko konexioak ikuspegi orokor batean erakutsiz. Ondoren, osagai bakoitzaren funtzionalitatea eta proiektuan betetzen duen zeregina zehaztuko dira, hardware ezarpenaren alderdi garrantzitsuenak azpimarratuz.

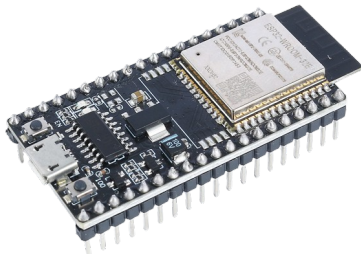
2.1.1 Hardware Bloque Diagrama

Diagraman (1. Irudia) ikus daitezkeen blokeek sistema elektronikoaren osagai nagusiak eta haien arteko konexioak laburbiltzen dituzte. Sistema honen muina ESP32 mikrokontrolagailua da, beste osagai guztiak koordinatzen dituena eta komunikazio protokolo desberdinak kudeatzeko gaitasuna duena. Diagraman ikus daitezkeen bezala, ibilgailuaren datuak OBD2 atakatik eskuratzen dira, CAN interfaze baten bidez. ESP32-k, integratuta duen CAN kontrolagailuari esker, seinale hauek interpretatu ditzake. Mugimenduaren datuak (azelerazioa, biraketa, posizioa) kanpoko IMU eta GPS moduluek biltzen dituzte, ESP32-ra konektatuta daudenak. SD txartel-sistema bat ere badago, datuak eta ereduak biltegitatzeko eta LCD pantaila bat, informazio sinplea bistaratzeko.

2.1.2 ESP32 Mikrokontrolagailua: Sistemaren bihotza

ESP32 mikrokontrolagailua, proiektu honen oinarri teknologikoa eta sistema osoaren adimenaren erdigunea da. Espressif Systems-ek garatua, gailu honek errendimendu handiko prozesamendua, konektibitate-aukera zabalak eta periferikoen aniztasuna konbinatzen ditu, guztiak energia-kontsumo txikiko diseinu batean. Bere arkitekturaren muinean Xtensa LX6 nukleo bikoitzeko prozesadorea dago, 240 MHz-ko erloju-abiadura funtzionatzen duena.

Nukleo bikoitzeko konfigurazio honek, zereginen banaketa eraginkorra eta denbora errealeko erantzun azkarra ahalbidetzen ditu, ikaskuntza automatikoko algoritmo konplexuak egikaritzeko eta inferentzia azkarra burutzeko ahalmena emanez. Erabilitako bertsioa, 2. Irudian ikusi ahal den ESP32-WROOM-32E plaka da, bere tamaina txikiagatik eta errendimendu handiagatik aukeratua.



2. Irudia: ESP32 Mikrokontrolagailua

Memoria alorrean, ESP32-k 520 KB SRAM ditu, datu-base eta algoritmoen egikaritzeko memoria bizkorra eskainiz, eta 4 MB Flash memoria. Memoria ahalmena mugatua izan arren, optimizazio teknikak eta eredu eraginkorrak erabiliz, errendimendu handiko adimen artifizialeko aplikazioak garatzea posible da.

Konektibitateari dagokionez, ESP32-k Wi-Fi eta Bluetooth haririk gabeko interfazeak barneratzen ditu. Hala ere, proiektu honetan, jomuga batez ere kableen bidezko komunikazioak izan dira helburu nagusia, CAN, UART, SPI eta I2C bezalako unitate periferiko estandarrak erabiliz. Periferiko hauen sorta zabalak, hainbat sentsore eta eragingailu integratzeko eta sistema pertsonalizatuak garatzeko aukera ematen dute.

Azkenik, azpimarratzekoa da ESP32 hardware irekiko plataforma dela, modulu ia guztien iturri kodea libreia izanik. Gainera, Espressif IDF garapen ingurunea, iturri irekikoa ere bada.

3. Irudia: CAN Interfazea

2.1.3 CAN Bus Interfazea: Ibilgailuaren Datuen Eskuratzeko Denbora Errealean

Kontrolagailu-gune Sarea (CAN) bus interfazea proiektu honen funtsezko atala da, ibilgailuaren datu dinamikoak denbora errealean eskuratzeko aukera ematen duena. Sistema ibilgailuaren CAN busarekin konektatzeko, bi osagai nagusi erabiltzen dira: ESP32 mikrokontrolagailuan integratutako CAN kontrolagailua eta kanpoko CAN Interfaze modulu bat (ikus 3. Irudia). OBD2 (Barne Diagnostikoko Sistema II) konektore estandarraren bidez egiten da konexio fisikoa ibilgailuaren CAN busera, protokolo eta konektore estandarizatu hau erabiliz auto moderno gehienetan.

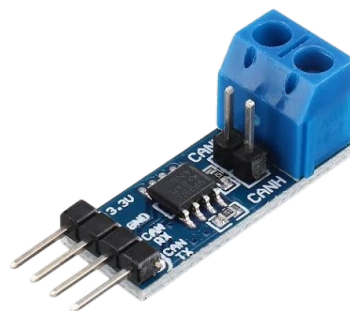
ESP32 mikrokontrolagailuak CAN kontrolagailu periferiko integratu bat dauka, CAN 2.0B estandararekin bateragarria dena. Kontrolagailu honek CAN busetik datuak transmititzeko eta jasotzeko gaitasuna eskaintzen du, software bidez konfiguragarriak diren iragazkiak eta bufferrak erabiliz. Hala ere, ESP32-ren CAN kontrolagailua zuzenean ibilgailuaren CAN busera konektatzeko ez da posible, busa diferentziala den bitartean kontrolagailuaren tentsio-maila polobakarra baita. Horregatik, CAN transzeptore modulu osagarri bat erabiltzen da.

CAN transzeptore moduluak hainbat funtzio garrantzitsu betetzen ditu. Alde batetik, maila elektrikoko konbertsioa egiten du, ESP32-ren 3.3V logika maila eta CAN busaren maila diferentziala (CAN-H eta CAN-L) bat etortzea bermatuz. Bestetik, busaren inpedantzia ezaugarria bermatzen du, zirkuitulaburrak edo tentsio gorabeherak gertatuz gero.

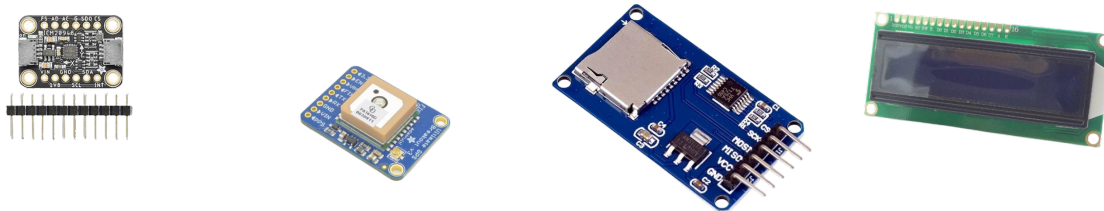
Konexio fisikoei dagokienez, CAN transzeptore moduluak bi konexio nagusi ditu ibilgailuaren OBD2 ataka-rekin: CAN-H eta CAN-L. Konexio hauek OBD2 konektorearen pin egokietara konektatzen dira, CAN busaren seinale diferentzialak igortzeko eta jasotzeko. ESP32-rekiko konexioa, berriz, CAN transzeptore moduluaren Tx (igorri) eta Rx (jaso) pinen bidez egiten da, ESP32-ren CAN kontrolagailuaren Rx eta Tx pinetara konektatuz, hurrenez hurren. Konexio hauek datu-truke bidirekzionala ahalbidetzen dute.

2.1.4 IMU: Ibilgailuaren mugimenduaren neurtzailea

Adafruit ICM20948 Neurketa Inertzialeko Unitatea (IMU) 4. Irudian azter daiteke. Ibilgailuaren mugimenduaren datu zehatzak eta integralak eskuratzeko funtsezkoa. Zehazki, ICM20948 IMU-ak bederatzi askatasun-gradu



eskaintzen ditu, hiru sentsore-mota integratuz: hiru ardatzeko azelerometro bat, hiru ardatzeko giroskopio bat eta hiru ardatzeko magnetometro bat. Sentsore-konbinazio honek ibilgailuaren azelerazio lineala, abiadura angeluarra eta orientazio magnetikoa aldi berean neurtzeko aukera ematen du, mugimenduaren analisisa oso zehatza eta osoa eginez.



4. Irudia: IMU Sentsorea 5. Irudia: GPS Modulua 6. Irudia: SD Modulua 7. Irudia: LCD Pantaila

Komunikazioari dagokionez, Adafruit ICM20948 IMU-a I2C (Inter-Integrated Circuit) komunikazio protokoloa erabiliz konektatzen da ESP32 mikrokontrolagailuarekin. I2C busak bi hari erabiltzen ditu: SDA (Serie Datuak) eta SCL (Serie Erlojua). Hari hauek ESP32-ren I2C periferikoko pinetara konektatzen dira. Gainera, Adafruit-ek eskaintzen duen liburutegi espezifikoak IMUaren integrazioa eta konfigurazioa errazten ditu, prototipoen garapena bizkortuz.

2.1.5 GPS Modulua: Munduko posizioa

Adafruit Ultimate GPS Breakout v3, 5. Irudian ikusten den modulua da, ibilgailuaren kokapenari eta abiadurari buruzko datu zehatzak eta fidagarriak eskuratzeko aukera ematen duena. GPS modulu honek, zehazki, posizionatze globaleko sistema (GPS) sateliteen seinaleak jasotzen eta prozesatzen ditu, latitudea, longitudea, altitudea, abiadura eta denbora bezalako informazio garrantzitsua denbora errealean zehaztuz. Gainera, 10 Hz-ko lagintzetarako segundoko 10 posizio-datu eskuratzeko ahalbidetzen du. Modulu honen muina PA1616S txipa da, L1 banda erabiltzen duena GPS seinaleak jasotzeko. Modulu honek, sentikortasun-maila oso ona izanik, seinale ahuleko inguruneetan ere posizionamendu fidagarria lortzen du. Zehaztasunari dagokionez, moduluak 3 metroko batez besteko zehaztasuna lortzen du.

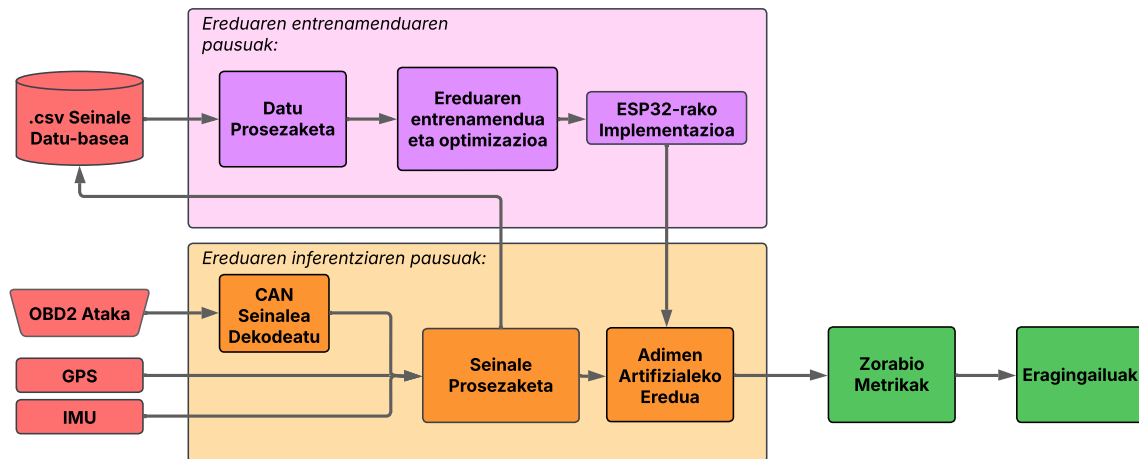
ESP32 mikrokontrolagailuarekin konektatzeko, GPS moduluak UART interfaze asinkronoa erabiltzen du. UART-ek, serieko komunikazio protokolo sinple eta hedatua, GPS datuen transmisio erraz eta fidagarria ahalbidetzen du. Gainera, Adafruit-ek eskaintzen duen liburutegiak, GPS moduluen integrazioa eta datu-interpretazioa errazten ditu Arduino garapen-ingurunean.

2.1.6 SD Txartelaren Kudeaketa: Datuen irakurketa eta idaztea

Hurrengo modulua, Micro SD memoria txartelak irakurgailua da (6. Irudia ikusi). SD txartelak datuak biltegiratzeko eta ikaskuntza automatikoko ereduak irakurtzeko erabiltzen direnez, modulu honek aurrean eragiketa hauek ahalbidetzen ditu. Konexioei dagokienez, SD modulu hau SPI busaren bidez konektatzen da ESP32-ra. Zehazki, SPI busak lau hari erabiltzen ditu: MOSI (Jabea kanpora, Esklabua barrura), MISO (Jabea barrura, Esklabua kanpora), SCK (Serie Erlojua) eta CS (Txip selekzioa). Hari hauek ESP32-ren SPI periferikoko pinetara konektatzen dira, datu-trukea ahalbidetuz.

2.1.7 LCD Pantaila: Informazioa ikusi eta aztertu

Azkenik, datu sinpleak eta proiektuaren kudeaketa errazteko, 7. Irudian ikusi ahal den, Arduino LCD Pantaila 16x2 instalatu da. Pantaila honek, bi lerrotan 16 karaktere erakusteko gaitasuna du, testu sinplea eta zenbakiak bistaratzeko egokia izanik. Batez ere, sistemaren egoera, sentsoreen neurketak edo ereduaren iragarpenaren emaitzak erakusteko erabiltzen da, erabiltzaileari erantzun bisuala emanez. Pantaila I2C komunikazio-protokoloa erabiliz konektatzen da ESP32 mikrokontrolagailuarekin, IMU-aren eskema berdina jarraituz.



8. Irudia: Software Prozesamendu-katea: Entrenamendua eta Inferentzia

2.2 Software: Prozesamendu-katea eta ereduaren inplementazioa

Atal honetan, zorabioa iragartzeko sistemaren software arkitektura eta inplementazioa zehaztuko dira, denbora errealeko funtzionamenduan arreta berezia jarritz. Software prozesamendu-kate osoa bi fase nagusitan banatuko da: lehenik, ereduaren entrenamendu-fasea, datu-base handi bat erabiliz eredu prediktibo bat sortzeko; eta bigarrenik, ereduaren inferentzia-fasea, eredu hori ESP32 plataforma txertatuan denbora errealean egikaritzeko eta zorabioaren iragarpena egiteko.

2.2.1 Software Prozesamendu-katearen Ikuspegi Orokorra

Software prozesamendu-katea, funtsean, bi fase nagusitan bana daiteke, 8. Irudian ikus daitekeen moduan: ereduaren entrenamendua (offline) eta ereduaren inferentzia (online). Diagrama honek, datu-fluxua eta prozesamendu-pausoak laburbiltzen ditu bi faseetan zehar, softwarearen funtzionamenduaren ikuspegi orokorra eskainiz.

Ereduaren entrenamendu-fasea (offline), konputagailu indartsuago batean egiten da, ESP32-tik kanpo. Fase honen helburua, zorabioa ereduztatzeko gai den ikaskuntza automatikoko eredu bat sortzea da. Horretarako, .csv formatuko datu-base handi bat erabiltzen da, ibilgailuaren seinaleak eta sentsoreen datuak biltzen dituen. Entrenamendu fase honetan, hainbat eredu entrenatzen dira, eta optimizatutakoan errendimendurik onena ematen duena eta sistema txertatuan eraginkortasunez inplementatu daitekeena aukeratzen da.

Ereduaren inferentzia-fasea (online), berriz, ESP32 plataforma txertatuan denbora errealean gertatzen da. Fase honen helburua, aurrez entrenatutako ereduak ibilgailuan txertatutako sisteman egikaritzea da, zorabio-maila denbora errealean lortzeko. Horretarako, ESP32-k OBD2 atakatik, GPS sentsoretik eta IMU sentsoretik datuak eskuratzen ditu etengabe. Datu hauek prozesatzen dira eta Adimen Artifizialeko Ereduari elikatzen zaizkio, metrikak kalkulatzeko. Metrika hauek, azkenik, eragingailuetara bidali daitezke, erosotasuna hobetzeko, sistema erantzunkor eta autonomo bat sortuz.

2.2.2 Ereduaren Entrenamendurako Datu Eskuratzeta eta Aurreprozesamendua

Ereduaren entrenamendurako datu-basea .csv formatuko fitxategi-bilduma bat da, gidatze-bidaia errealean bildutako seinaleak eta sentsoreen datuak biltzen dituen. Datu-base honen helburua zorabioa ereduztatzeko beharrezkoak diren datuak eskaintzea da. Datu-basea, hainbat egunetan hartutako bidaia luzeen erregistroetan oinarritzen da, ereduak egoera anitzetan orokortzeko gaitasuna izan dezan, gutxinaka bidaia gehiago eginez gero eta datu-base handiagoa eraikiz. Datu-base datu-zutabe hauek ditu: Denbora, Bolantearen Angelua, Motorraren Bira Minutuko, Abiadura, Martxa, Balazta, Azeleragailu, Indar-parea, Kontsumoa, Azelerazioa X, Azelerazioa Y, Azelerazioa Z, Girokopio X, Girokopio Y, Girokopio Z, GPRMC, GPVTG, GPGGA. Azpimarratu behar da azken hiru hauek NMEA formatua, hau da, GPS periferikoei datuak partekatzeko erabiltzen duten estandarra, jarraitzen dutela.

Datu-basea osatu ondoren, aurreprozesamendu-fase bat aplikatzen zaio, ereduaren entrenamendua optimizatzeko eta errendimendua hobetzeko. Aurreprozesamendu-fase honetan, hainbat pauso jarraitzen dira. Lehenik, datuen garbiketa egiten da, balio galduak tratatuz eta zarata murriztuz. Bigarrenik, ezaugarrien ingeniariak egiten da, datu gordinetatik erabilgarriak diren ezaugarriak lortuz (adb., ezaugarri bakoitzaren deribatuak, erroreak, maximoak, minimoak, etab.). Eta azkenik, datuen transformazioa egiten da, ezaugarri guztiak eskala berdinerara eramanez, algoritmoak errendimendu hobea izan dezan. Aurreprozesamendu fase hau Python programazio-lengoaia erabiliz inplementatzen da.

2.2.3 CatBoost Ereduaren Entrenamendua, Implementazioa eta Optimizazioa

Ikaskuntza automatikoko eredu gisa, CatBoost algoritmoa aukeratu da (Prokhorenkova et al. (2019)), bere errendimendu handia eta sistema txertatuetan eraginkortasunez ezartzeko gaitasuna kontuan hartuta. CatBoost, *gradient boosting* teknikan oinarritutako algoritmo bat da, errendimendu handiko iragartze-eredu sortzeko gai dena, baita datu-multzo txiki eta ertainekin ere. Bere abantaila nagusien artean, gaintrenamenduari aurre egiteko erregular-tasun indartsuak, eta inferentzia azkarra egiteko ahalmena daude, sistema txertatuetan inplementatzeko ezinbestekoak diren ezaugarriak.

Ereduaren errendimendua ebaluatzeko, RMS (batez besteko koadratikoa) eta MAE (batez besteko errore absolutua) metrikak erabili dira, iterazio bidezko optimizazio sare batean. Teknika honek, hiperparametroen bilaketa-espazioa eraginkortasunez arakatzeko aukera ematen du, errendimendu hoberena ematen duten konbinazioak azkar aurkituz. Hala ere, optimizazio-prozesuak mugapen eta erronka garrantzitsu batzuk planteatu ditu, batez ere sistema txertatuko inplementazioa kontuan hartuta.

Sistema txertatuko ereduaren inplementazio eraginkorra bermatzeko, ereduaren tamaina murriztuz eta inferentzia abiadura handituz, modeloaren optimizazio fase bat ere egin da. Optimizazio honetan, zuhaitzaren sakonera eta iterazio-kopurua bezalako parametroak murrizten dira, errendimenduaren galera minimoa mantenduz. ESP32 mikrokontrolagailuaren memoria-mugapenak ereduaren tamaina mugatzera behartu gaitu, errendimendu potentziala murriztuz. Ereduak kudeatzeko CPP, JSON eta BIN fitxategi formatu desberdinak aztertu ditugu, hauetako bakoitzak abantaila eta desabantaila desberdinak izanik. Azkenik, denbora errealeko egikaritzearen mugak, inferentzia-abiadura ahalik eta azkarrena izatea eskatzen du, sistema erantzunkorra izan dadin. Muga hauek kontuan hartuta, ereduaren kudeaketa dinamiko sistema bat diseinatu da, ziklo batean oinarritzen dena. Kudeaketa-sistema honetan, ziklikoki, eredu SD txartelatik memorian kargatzen da, inferentzia egiten da eta eredu memoriatik ezabatzen da, hurrengo eredu kargatzeko lekua askatuz. Ikuspegi honek, memoriaren erabilera optimizatzea ahalbidetzen du, eredu kopuru handiagoa kudeatu ahal izateko.

Ereduak kargatzeko formatuari dagokionez, hasiera batean eredu zuzenean kodean txertatu zen CPP formatuan, C++ kodean oinarritutako formatu honek errendimendu handiko inferentzia ahalbidetzen baitu. Hala ere, ereduaren kudeaketa dinamikoa inplementatzerakoan, CPP formatua ez zen egokiena, karga eta ezabatze-prozesuak konplexuak eta astunak baitziren. Hori dela eta, JSON formatua, zeinek datuak testu-formatuan gordetzen dituen, aldatzea erabaki zen, ulergarritasuna eta plataforma anitzeko bateragarritasuna bermatzeko asmoz. JSON formatuak, ereduaren karga eta ezabatzea erraztu zuen, baina inferentzia-abiadura eragozten zuen. Azkenik, errendimendua are gehiago optimizatzeko, datuak formatu bitarrean gordetzea erabaki zen, hau da, BIN formatuan, tamaina minimoa eta karga-abiadura azkarrena ahalbidetzen baititu. BIN formatuak, ereduaren karga dinamiko eraginkorragoa bihurtu zuen, inferentzia-abiadura handia lortuz eta memoria erabilera ondo kudeatuz.

3 Emaitzak

Proiektu honetan garatutako sistemaren ezarpen eta konexioak arrakastatsuak izan dira, eta sistema funtzionatzen duela egiaztatu da. Honen froga gisa, ibilgailuaren CAN busetik eta GPS modulutik denbora errealean datuak eskuratu eta prozesatu ahal izan direla aipa daiteke. Hona hemen dekodifikazio prozesuaren bi adibide:

```
0x44d 8 02 58 00 00 00 00 00 00
```

Kasu honetan, 0x44d ibilgailuaren abiadura adierazten duela dakigu, badakigu ere, lehenengo bi byte-etaik ezagutzen dugun kilometro orduko zenbakia lortzeko, lehenengo byte-a bider 256 egin behar dela, bigarrena gehitu eta emaitza horri zati 100 egin. Datu hauekin, $(0x02 \times 256 + 0x58) / 100 = 600 / 100 = 6 \text{ km/h}$.

```
0x208 8 0A 2C 32 10 00 00 00 00
```

Beste kasu honetan, 0×208 ibilgailuaren motorraren ezaugarriak ditu. Lehenengo eta bigarren byte-ak Motorraren Birak Minutuko, hirugarrenak Indar-parea eta laugarrenak Azelerazio Orokorra neurtzen dituzte. Motorraren birak dekodeatzeko, bider 256 egin behar zaio lehenengo byte-ari, bigarrena gehitu eta ondoren zati zortzi egin, kasu honetan 325.5 bira minutuko lortuz. Indar-parearen eta Azelerazioaren eskalak absolutuak ez dakizkigunez balio erlatiboekin lan egiten dugu.

GPS modulutik datuak eskuratzeko, lehen aipatutako GPRMC, GPVTG eta GPGLL formatuak ulertu behar dira. Hona hemen horren beste adibide bat:

GPRMC,192416.300,A,43.30,N,00809.4347,W,17.21,55.20,221224,,A

Karaktere kate honen esanahia, 2024ko abenduaren 22ko 19:24:16 UTC orduan, GPS hartzailea 43 gradu, 30.0420 minutu Ipar latitudean eta 8 gradu, 09.4347 minutu Mendebaldeko longitudean zegoen, Ferrol hirian. Ibilgailua 17.21 korapiloko ($\approx 31.87\text{km/h}$) abiaduran 55.20 gradu ko noranzkoan mugitzen ari da, eta posizionamendu modua autonomoa da.

1. Taula: CatBoost Ereduen Inferentzia Denbora Konparatiboa Formatu eta Konfigurazio Desberdinetan

Ereduak	Denbora (s)
JSON osoa	67.31
BIN osoa	3.58
Simplifikatutako JSON	7.14
Simplifikatutako BIN	1.65

Denbora errealeko sistema ere arrakastaz funtzionatzen duela egiaztatu da. ESP32 mikrokontrolagailuak datuak eskuratu, prozesatu eta Adimen Artifizialeko Ereduari elikatzen dizkio etengabe, inferentzia denbora oso azkarrean burutuz. Emaizten taulan (1. Taula) ikus daitekeen bezala, ereduaren inferentzia denborak oso baxuak dira, BIN formatuan optimizatutako ereduak 1.65 segundoko inferentzia denbora lortuz. Denbora hauek, sistema denbora errealean funtzionatzeko gai dela erakusten dute, eta eragingailuetara seinaleak bidaltzeko helburua (5-10 segundoko latentziarekin) askoz gaintzen dute.

Laburbilduz, aurkeztutako sistemak soluzio eraginkor eta merke bat dela frogatu da. ESP32 mikrokontrolagailuan oinarritutako plataforma honek, arrakastaz gauzatzen ditu datu-baseak sortzeko prozesuak, inferentzia azkarra eta ikaskuntza automatikoko ereduaren aplikazioak, guztiak ibilgailuan inolako aldaketarik egin behar izan gabe. Horrela, sistema honek erosotasuna hobetzeko eta gidatzeko esperientzia aberasteko aukera berriak irekitzen ditu automobilgintza sektorean eta haratago.

3.1 Etorkizuna

Proiektu honetan garatutako sistema, etorkizunean hainbat eragingailurekin integra daiteke. Balizko Eragingailuren artean, eserleku bibrazio-sistema bat aipa daiteke non seinale bat jasotzean, bibrazio leunak aktiba daitezkeen, sentsazio propiozeptiboa aldatuz eta ondorez arinduz. Aireztapen sistema aktibo bat ere inplementa daiteke, aire-fluxua eta tenperatura bidaiari-lekuan kontrolatzeko. Argi-banda bat eragingailu gisa erabiltzea beste aukera bat da, adibidez, mezuak bidaltzeko.

Etorkizuneko lanari dagokionez, hainbat norabide aztertu daitezke proiektu honen garapena jarraitzeko, esate baterako, datu-basea handitzea, eragingailuekin egindako sistema integratzea eta ibilgailu errealetan probak egitea.

Erreferentziak

- Devi, D., Kathula, N. R., Kalluri, G., & Bondalapati, L. (2023). Design and implementation of image processing application with zynq soc. *International Journal of Computing and Digital Systems*, 14.
- El-Khozondar, H. J., Mtair, S. Y., Qoffa, K. O., Qasem, O. I., Munyarawi, A. H., Nassar, Y. F., Bayoumi, E. H., & Halim, A. A. E. B. A. E. (2024). A smart energy monitoring system using esp32 microcontroller. *e-Prime - Advances in Electrical Engineering, Electronics and Energy*, 9:100666.
- Gill, S. S., Golec, M., Hu, J., Xu, M., Du, J., Wu, H., Walia, G. K., Murugesan, S. S., Ali, B., Kumar, M., Ye, K., Verma, P., Kumar, S., Cuadrado, F., & Uhlig, S. (2024). Edge ai: A taxonomy, systematic review and future directions. *Cluster Computing*, 28(1).
- Griffin, M. (2012). *Handbook of Human Vibration*. Academic Press.
- ISO 2631-1:1997 (1997). Mechanical vibration and shock – evaluation of human exposure to whole-body vibration – part 1: General requirements. iso 2631-1 international organisation for standardisation. Standard, International Organization for Standardization, Geneva, CH.
- Khan, M. U., Ishtiaq, A., Ali, S., Habib, K., Samer, S., & Hafeez, E. (2021). A review of system on chip (soc) applications in internet of things (iot) and medical.
- Mandumula, S. R., Park, J., Asolkar, R. P., & Somashekar, K. (2023). Multi-sensor object detection system for real-time inferencing in adas. In *2023 IEEE Symposium Series on Computational Intelligence (SSCI)*, 215–220.
- Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2019). Catboost: unbiased boosting with categorical features.
- Samantaray, R. (2023). Adas sensor data handling in the world of autonomous mobility. In *SAE Technical Paper 2023-01-0993*.
- Yadav Kanneboina, R., Satish Kumar, G., Vidyadhar, R. P., Reddy, V. K., & Tammineni, S. (2024). Development of autonomous vehicle system using esp32 microcontroller. In *2024 Asia Pacific Conference on Innovation in Technology (APCIT)*, 1–6.

4 Eskerrak eta Oharrak

Artikulu hau Jon Ander Ruiz Colmenares, Óscar Mata-Carballeira eta Estibalitz Asua Uriarte izenpean badago ere, ikerketa hau Elektrizitatea eta Elektronika saileko *Elektronika Digitaleko Diseinu* taldeari esker egin da.