

MARC MOLINA FIGUERAS

**DETECCIÓ PER VISIÓ ARTIFICIAL D'AVISOS EN ASSAJOS DE
SEGURETAT ACTIVA**

TREBALL DE FI DE GRAU

dirigit pel Dr. Nicolau Cañellas Alberich

Grau d'Enginyeria Electrònica Industrial i Automàtica



UNIVERSITAT ROVIRA I VIRGILI

Tarragona

2021

Índex

1	Objectius.....	1
2	Introducció.....	2
2.1	Pistes de proves.....	3
2.2	Departament ADAS.....	5
3	Antecedents del projecte.....	6
4	Implementació del sistema de visió artificial.....	8
4.1	Estat de l'art.....	8
4.2	Hardware.....	11
4.3	Software (Open CV).....	13
4.4	Estratègia 1: comparació d'histogrames.....	14
4.5	Estratègia 2: detecció de color.....	20
4.6	Avaluació de les estratègies de detecció.....	24
5	Implementació de la comunicació CAN i <i>data logger</i>	33
5.1	Bus CAN.....	33
5.2	Arxius DBC.....	35
5.4	Hardware.....	37
5.5	Software.....	40
6	Assajos de validació a les pistes de proves.....	41
6.1	Instrumentació del vehicle i configuració del <i>data logger</i>	41
6.2	Assaig de validació 1 (dia pluja amb poca lluminositat).....	48
6.3	Assaig de validació 2 (assajos a primera hora matí).....	49
6.4	Assaig de validació 3 (assajos tarda).....	49
6.5	Anàlisis dades assajos.....	50
7	Conclusions.....	54
8	Annexes.....	55
8.1	Annex 1: Comparació d'histogrames.....	55
8.2	Annex 2: Detecció de color.....	56
8.3	Annex 3: Arxiu DBC.....	57
8.4	Annex 4: Esquema CAN HAT.....	58
8.5	Annex 5: Script Final.....	59
9	Referències.....	60

1 Objectius

Aquest projecte de final de grau forma part de les pràctiques realitzades a Applus IDIADA.

L'estada a IDIADA s'ha realitzat al departament ADAS (*Advanced Driver Assistance Systems*), on es duen a terme projectes R+D en sistemes avançats d'ajuda al conductor i assajos per a la validació de la seguretat activa dels prototips abans que el vehicle comercial surti al mercat.

La motivació d'aquest projecte sorgeix de la necessitat del departament ADAS d'implementar un nou sistema d'instrumentació per assajar els vehicles prototips. Aquest sistema ha de ser capaç de detectar els avisos visuals, que es generen en els *dashboard* dels cotxes, per alertar al conductor que actui sobre el fre i evitar una col·lisió. En cas que el conductor no faci cas de l'avís, el cotxe activarà el fre autònom d'emergència.

La detecció dels avisos forma part de la nova normativa que s'implementarà en un futur pròxim per part d'Euro NCAP i NHTSA, on es regularà oficialment l'ús dels avisos d'alerta. A conseqüència de la nova normativa, serà necessària la detecció dels avisos per poder validar que es compleixen els nous requeriments.

A part dels avisos visuals també es generen avisos acústics mitjançant els altaveus del cotxe i avisos hàptics mitjançant vibracions al volant o generant una estirada sobtada del cinturó de seguretat del conductor. Actualment ja hi ha sistemes en la instrumentació dels assajos per a la detecció dels avisos no visuals.

Els requeriments per al sistema han de ser els següents:

- Ha de ser un sistema de baix cost, ja existeixen solucions comercials amb un preu elevat.
- Ha de ser capaç de detectar els avisos visuals en diferents condicions lumíniques.
- Ha de poder utilitzar-se en tots els displays dels diferents cotxes i per a tots els diferents tipus d'avisos visuals que es poden generar.
- Ha de ser fàcil d'utilitzar per part dels enginyers d'assajos.
- Ha de tenir una fàcil integració amb la instrumentació existent.
- Ha de tenir una ràpida instal·lació en els prototips a assajar.
- Ha de comunicar-se amb el *data logger* de la instrumentació mitjançant bus CAN.
- Ha de tenir un temps màxim de detecció de 100 ms.



2 Introducció

Applus IDIADA és una empresa global al servei de la indústria de l'automòbil amb més de 30 anys d'experiència en el desenvolupament de producte, que presta serveis de disseny, enginyeria, assaig i homologació de vehicles. L'èxit d'IDIADA en el desenvolupament de productes es basa en una combinació única d'enginyers altament experimentats, instal·lacions d'assaig i desenvolupament d'última generació i l'impuls constant cap a la innovació.

L'empresa compta amb més de 2.500 professionals i una xarxa internacional de filials i sucursals en 22 països, el que garanteix als seus clients solucions personalitzades de valor afegit.

La missió d'IDIADA és ajudar als seus clients en les seves activitats de desenvolupament de productes proveint amb serveis d'enginyeria, assajos i homologació ajustats a les seves necessitats.

IDIADA (80% propietat d'Applus + 20% del Govern de Catalunya) opera sota un contracte exclusiu des del centre tecnològic de 351 hectàrees a prop de Barcelona (propietat del Govern de Catalunya) des de 1999. El contracte per gestionar el negoci s'estén fins a Setembre 2024 i tot i que és renovable en períodes de cinc anys fins a 2049, l'expectativa actual és que no hi haurà més extensions sinó una licitació per a una nova concessió de 20 anys.

El disseny, la qualitat i la varietat dels traçats, la repetibilitat dels assajos realitzats, l'estratègia de millora constant, el manteniment i la inversió en les últimes tecnologies, es combinen per fer de les pistes de prova d'IDIADA la instal·lació més eficient.

2.1 Pistes de proves

Pistes a Espanya

Actualment Applus + IDIADA gestiona dues pistes de prova: una a Espanya i un altre a la Xina, les dues considerades les pistes de prova independents més completes d'Europa i Àsia, respectivament.

Inaugurat el 1994, el circuit de proves d'Applus + IDIADA és el més complet i independent d'Europa.

Al centre tècnic de 370 hectàrees, es troba el camp de proves, on la seguretat i la confidencialitat són de màxima prioritat. Applus IDIADA ha pres totes les mesures raonables per crear un entorn de treball ideal, que compleixi fins i tot amb els estàndards industrials més estrictes.

Situat prop de la costa mediterrània, el clima favorable permet proves durant tot l'any. Les temperatures d'estiu són ideals per a proves favorables en climes càlids, i els programes de desenvolupament poden continuar independentment del clima durant els mesos d'hivern, és per això que aquesta instal·lació és la primera opció, sense importar el tipus de prova. El resultat és que el complex de l'Albornar és reconegut com una de les millors instal·lacions d'aquest tipus a tot el món.

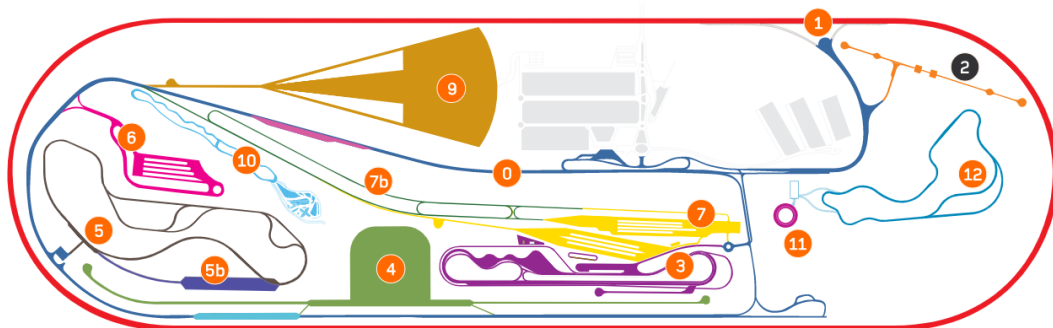


Figura 2-1. Pistes de proves a Espanya

- | | |
|-------------------------|------------------------|
| 1- Pista alta velocitat | 7- Pista confort |
| 2- Pista soroll | 7b- Pista confort B |
| 3- Pista fatiga | 9- Pista dinàmica B |
| 4- Pista dinàmica | 10- Pista off-road |
| 5- Pista dry handling | 11- Pista wet circle |
| 6- Pista desnivells | 12- Pista wet handling |

Pistes a Xina

Des de 2016 Applus + IDIADA està gestionant un nou complex de pistes de proves a la Xina, obert a totes les companyies del sector de l'automòbil.

Applus IDIADA contribueix al disseny de les noves pistes de prova, mantenint el mateix nivell de qualitat i seguint els mateixos estàndards de seguretat i confidencialitat que s'apliquen al centre tècnic d'Espanya.

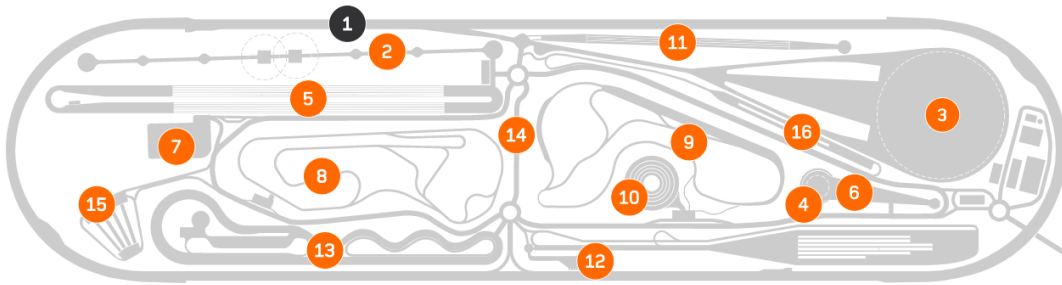


Figura 2-2. Pistes de proves a Xina

- | | |
|-------------------------|----------------------------|
| 1- Pista alta velocitat | 10- Pista wet circle |
| 2- Pista soroll | 11- Pista drift & pull |
| 3- Pista dinàmica | 12- Pista durabilitat |
| 4- Pista frenada | 13- Pista fatiga |
| 5- Pista NVH | 14- Carretera general |
| 6- Pista multi ús | 15- Pista desnivells |
| 7- Pista off-road | 16- Pista frenada B |
| 8- Pista dry handling | 17- Pista frenada en corba |
| 9- Pista wet handling | |

2.2 Departament ADAS

Applus + IDIADA ofereix serveis integrals per a ADAS i Vehicle Autònom durant tot el cicle de desenvolupament, des de la fase de definició conceptual fins a la validació final. Aquests serveis inclouen el suport complet per a les funcions d'implementació, combinant l'especialització en els requisits funcionals, processos de desenvolupament de programari, experiència en integració de maquinària i eines de prova i simulació d'última generació.

Ofereix una experiència integral per a tots els sistemes de mercat i assisteix durant tot el cicle de desenvolupament:

- Benchmarking per a avaluació de vehicles i sistemes.
- Especificació de funcions i requisits de sistema.
 - o Requisits d'usuari, funcionals i de rendiment.
 - o Especificació HW / SW.
 - o Eines de traçabilitat seguint els estàndards del sector de l'automoció.
- Desenvolupament de funcions i sistemes.
 - o Fusió de dades i desenvolupament de control lògic.
 - o Prototip i producció de nivell SW.
- Proves físiques (objectives i subjectives).
- Proves virtuals (vehicle in the loop).
- Integració de sistema.
 - o Aplicació de sistema / calibratge de sistema.
 - o Ajust específic de mercat.
- Vehicle complet, funció i sistema de validació
 - o Pistes de proves combinades i proves de camp operacionals en tot el món.



3 Antecedents del projecte

L'estada a IDIADA es va iniciar el 2 de març de 2020, dues setmanes després es va decretar l'estat d'alarma al país a causa de la crisi sanitària. Es van cancel·lar tots els convenis de pràctiques. No ha estat possible reprendre les pràctiques fins el 13 d'octubre de 2020.

Durant les dues setmanes que va durar l'estada inicial es va fer una primera aproximació al problema i es va proposar desenvolupar un sistema de *Deep Computer Vision*. El sistema es basava en una xarxa neuronal convolucional.

Durant els mesos de confinament i de manera autònoma es va anar desenvolupant aquesta primera aproximació i el resultat va ser un sistema funcional.

Per a la realització de la xarxa neuronal es va utilitzar Tensor Flow i una base de dades de l'institut FÜR NEUROINFORMAIK (Alemanya).

TensorFlow és el sistema d'aprenentatge autònom de Google Brain, alliberat com a programari de codi obert el 9 de novembre de 2015. És l'eina més utilitzada en el món del Deep Learning.

No obstant al reprendre les pràctiques es va decidir descartar aquesta opció per part d'IDIADA i intentar buscar noves estratègies. Aquest projecte mostrarà les noves estratègies proposades al reprendre les pràctiques i el seu desenvolupament.

Les noves estratègies descarten l'ús de xarxes neuronals i es basen en Open CV. S'implementen algoritmes per al modelat de les imatges i amb els models obtinguts s'assignen criteris per a la detecció d'avisos visuals.

En els apartats 4.4 i 4.5 es presenten les dues estratègies de detecció, hi ha una explicació teòrica dels fonaments de les estratègies i es mostren les funcions de la llibreria utilitzades per a la implementació.

En l'apartat 4.6 s'explica la metodologia que es durà a terme per a la validació de les dues estratègies de detecció. Bàsicament el procediment que es seguirà serà la captura de vídeos del *dashboard* d'alguns prototips durant assajos de desenvolupament. Després s'utilitzaran els vídeos capturats per a fer una un anàlisi del funcionament de les estratègies. Comparant *frames* en què hi apareixen avisos visuals amb altres frames en què no.

Exemple d'avís visual

Per tal de posar en context l'objectiu del treball es mostren unes imatges que serveixen per entendre la funcionalitat dels avisos visuals i la seva seqüència de funcionament.

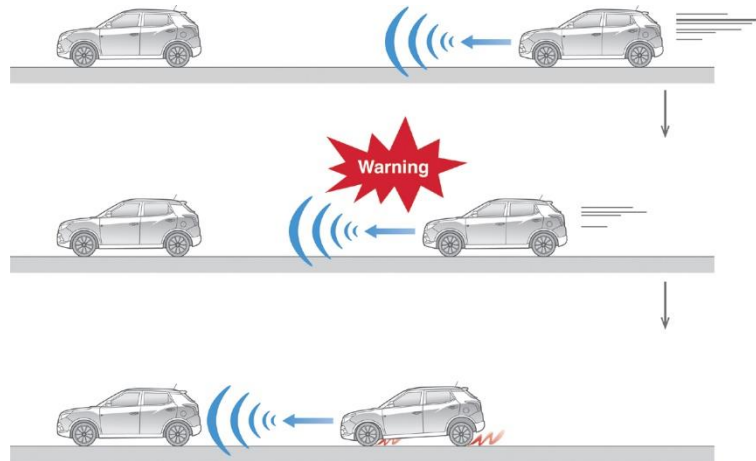


Figura 3-1. Lògica d'activació del sistema.



Figura 3-2. Exemple d'avís visual.

4 Implementació del sistema de visió artificial

4.1 Estat de l'art

En les últimes dues dècades s'ha viscut un auge respecte a la intel·ligència artificial i les seves possibilitats. En aquest apartat es mostraran les seves principals característiques.

El camp de la intel·ligència artificial no és un descobriment nou. A la segona meitat de segle XX es van realitzar diferents investigacions. Es van desenvolupar sistemes capaços de realitzar tasques senzilles i millorar la seva eficàcia a mesura que les duïen a terme. Però aquests projectes es van veure limitats per la tecnologia disponible i la intel·ligència artificial no va mostrar avanços significatius fins a principis de segle XXI.

El 2006 Geoffrey Hinton va publicar un treball on mostrava com entrenar un sistema d'intel·ligència artificial que era capaç de reconèixer dígitos escrits a mà amb una precisió superior al 98%. Aquesta tècnica va ser nombrada com "*Deep Learning*", aquest descobriment va fer despertar de nou l'interès científic en la intel·ligència artificial.

Es pot considerar la publicació de Geoffrey Hinton com un punt d'inflexió per a la disciplina. En els últims anys s'han aconseguit grans avenços en *Deep Learning* i s'han desenvolupat noves estratègies per a crear sistemes d'intel·ligència artificial.

Xarxes neuronals

Les xarxes neuronals són la base fonamental del *Deep Learning*. Originalment inspirades per arquitectura del mateix cervell humà.

Les xarxes neuronals són versàtils, potents i fàcilment escalables. Aquestes propietats fan que el seu ús sigui ideal per afrontar els problemes més complexos d'intel·ligència artificial. El seu ús ja està estès a sistemes útils per al dia a dia. Podem trobar dos exemples d'això en Google Imatges (problema de classificació de milions d'imatges) i Siri d'Apple (reconeixement de veu en temps real).

La idea de neurona artificial va ser introduïda el 1943 pel neurobiòleg Warren McCulloch i el matemàtic Walter Pitts en una publicació titulada "*A Logical Calculus of Ideas Immanent in Nervous Activity*". En aquesta presentació, els autors modelaven un sistema computacional simple que buscava explicar el funcionament biològic de les neurones dels animals.



Es va crear una gran expectació al respecte aquest tipus de tecnologia i es creia que en poc temps els humans havien de ser capaços de crear sistemes artificials amb intel·ligència pròpia. L'eufòria inicial es va dissipar en la dècada de 1960 quan es va comprendre que el problema de crear sistemes artificials intel·ligents era molt més gran del que es creia inicialment.

A partir d'aquest moment tot el desenvolupament associat a les xarxes neuronals entra en un període de pausa. Actualment es viu una segona onada, això és gràcies a diversos factors que fan possible la solució de problemes cada vegada més complexos.

Alguns d'aquests factors diferenciadors respecte a la primera onada viscuda a mitjans del segle passat són:

- La gran quantitat de dades que hi ha per entrenar les xarxes neuronals, cosa que permet solucionar nous problemes.
- L'enorme augment de la potència de computació a partir de la dècada de 1990 ara és possible entrenar grans xarxes neuronals amb un temps raonable.
- Els algoritmes d'entrenament han estat millorats, les diferències no són molt grans respecte als originals, però algunes millores concretes han permès un avanç significatiu.

Xarxes neuronals convolucional

Les xarxes neuronals convolucional es van crear a partir de l'estudi de la part del còrtex encarregada de la visió i han estat utilitzades en reconeixement d'imatges des de la dècada de 1980. En els últims anys gràcies a l'augment de la potència computacional, l'accés a grans bases de dades i els nous algorismes per entrenar xarxes neuronals han permès als sistemes de *Deep Computer Vision* assolir cotes molt elevades de funcionament.

La necessitat de crear les Xarxes Neuronals Convolucional radica que el problema de reconeixement d'imatges no pot ser resolt per una xarxa neuronal ordinària. La quantitat de paràmetres necessaris per al seu funcionament estan fora de la capacitat actual.

El component essencial per a aquest nou tipus de xarxa neuronal és la capa convolucional. En la què les neurones de la primera capa no estan connectades a cada píxel individualment sinó a un rang de píxels. La segona capa convolucional està connectada a un conjunt delimitat de píxels de la primera capa. I així consecutivament.

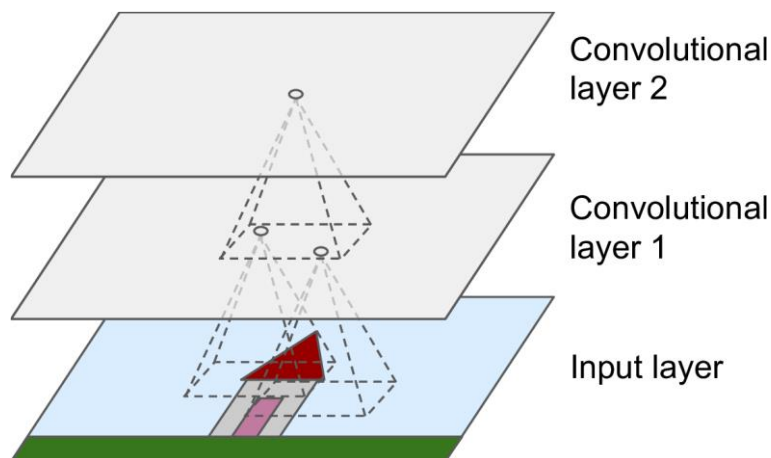


Figura 4-1. Estructura d'una xarxa convolucional.

4.2 Hardware

Raspberry Pi 3 B +

Raspberry Pi és un ordinador de placa reduïda. Està dissenyada amb el propòsit d'oferir una plataforma de desenvolupament informàtic de baix cost.

Per a aquest projecte es utilitzada la versió 3B+. L'elecció d'aquest model va ser suggeriment d'IDIADA. El departament ADAS disposava de diverses unitats i volia avaluar si era possible la seva utilització.



Figura 4-2. Raspberry Pi.

- Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz
- 1GB LPDDR2 SDRAM
- 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE
- Gigabit Ethernet over USB 2.0 (maximum throughput 300 Mbps)
- Extended 40-pin GPIO header
- CSI camera port for connecting a Raspberry Pi camera
- 5V/2.5A DC power input

Logitech C920

La càmera utilitzada per a la captura de vídeo ha estat determinada per la disponibilitat tècnica del departament ADAS, s'utilitzarà la mateixa càmera que utilitzen per a altres funcions.



Figura 4-3. Logitech C920.

- Resolució màxima: 1080p / 30 fps - 720p / 30 fps
- Tipus d'enfocament: automàtic
- Micròfon integrat: estèreo
- Clip universal compatible amb trípodos per a monitors, pantalles LCD o portàtils

4.3 Software (Open CV)

Open CV (Open Source Computer Vision) és una llibreria de visió artificial i *Deep Learning*. Va ser creada amb l'objectiu d'oferir un marc de treball per a la creació de sistemes de visió artificial i per accelerar la implementació dels sistemes de visió artificial en aplicacions comercials.

La llibreria té més de 2500 algoritmes optimitzats, els quals inclouen tota mena d'estratègies. Tots aquests algoritmes poden ser utilitzats per a detectar objectes, classificar accions humanes, fer seguiment d'objectes, extreure models 3D i moltes més aplicacions.

Actualment és utilitzada per més de 47 mil programadors i per grans companyies.

Permet la utilització de C++, Python, Java i MATLAB. La mateixa llibreria va ser escrita en C++ i la seva principal avantatge respecte a altres llibreries de visió artificial és que en general els algoritmes implementats estan dissenyats per tal d'optimitzar al màxim el temps d'execució. És per aquest motiu que s'ha triat aquesta llibreria.

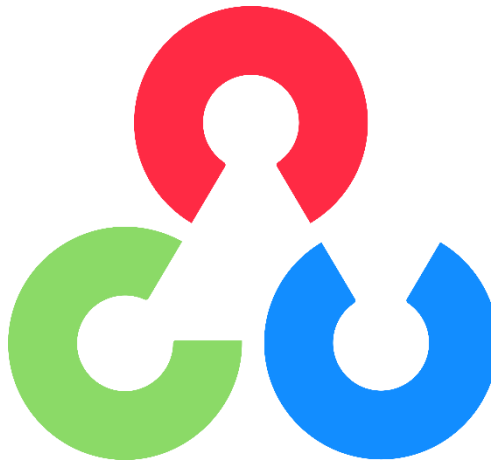


Figura 4-4. Logotip Open CV.

4.4 Estratègia 1: comparació d'histogrames

La primera estratègia proposada per a modelitzar els *frames* capturats es basa a obtenir l'histograma de la imatge en un espai de color determinat i comprar l'histograma obtingut amb un histograma que correspon a una imatge d'avis visual. En funció de la correlació dels dos histogrames es pot classificar el *frame* capturat. Open CV disposa de diversos mètodes per a poder generar i comparar els histogrames (*veure annex 1*).

Histograma

Un histograma és una recopilació de dades agrupades en intervals denominats *bins*. Aquestes agrupacions poden tenir la mateixa mida o no, aquesta mida es denomina *rang*. La representació de la informació continguda en un histograma es representa en forma de gràfic de barres, tot i que si el *rangs* són molts petits, és similar a un gràfic continu.

En l'eix horitzontal es representen els *bins* i en l'eix vertical es representen els valors continguts dintre de cada *rang*. La informació que es pot representar pot ser molt variada i de diverses dimensions, és a dir en el cas de les imatges, podríem realitzar un histograma on es representés només la intensitat de cadascun dels píxels i també es podria realitzar un histograma on es representés un determinat espai de colors.

Com a exemple es pot agafar una imatge de baixa resolució i dividir-la en els seus píxels.

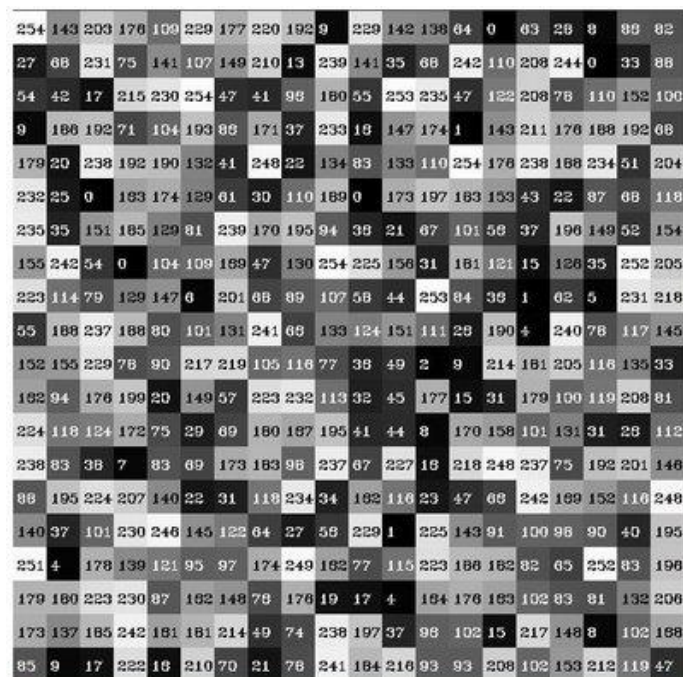


Figura 4-5. Imatge dividida en els seus píxels. Veure [1].

La imatge representada mostra un valor associat a la intensitat de cadascun dels seus píxels, per tant per tal de poder contenir aquesta informació en un histograma és necessari establir el rang de cada *bin* i número total de *bins*.

- Rang total intensitats dels píxels $[0, 255]$
- Bin 1 \rightarrow Rang $[0, 15]$
- Bin 2 \rightarrow Rang $[16, 31]$
- Bin 16 \rightarrow Rang $[240, 255]$

$$[0, 255] = [0, 15] \cup [16, 31] \cup \dots \cup [240, 255]$$

Que és equivalent a escriure:

$$\text{Rang total} = \text{bin 1} \cup \text{bin 2} \cup \dots \cup \text{bin 16}$$

Així doncs per a representar de forma gràfica l'histograma es contenen el número de píxels continguts en cada rang i és el valor que s'associa a cada *bin*. Una possible representació seria la següent.

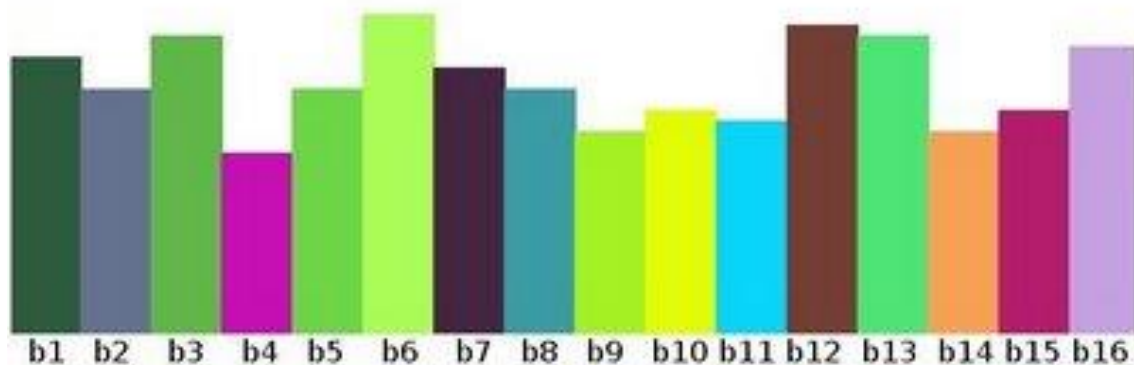


Figura 4-6. Exemple senzill d'histograma. Veure [2].

Espai de colors

Un espai de color és un sistema matemàtic per a la interpretació d'un color. Cada espai de color té una organització específica on la representació d'un color determinat correspon a la combinació de certes coordenades dels eixos que formen l'espai de color.

Els dos principals espais de colors són RGB i HSV. En el primer es descriuen els colors com una combinació de Blau, Verd i Vermell i el segon en funció del Matís, Saturació i Valor que són els noms que reben els eixos de referència del citat espai.

Per a la implementació de l'estratègia, l'espai de color és la base per a la modelització dels *frames*. A partir del càlcul d'histogrames de tres dimensions es conté la informació de cada imatge.

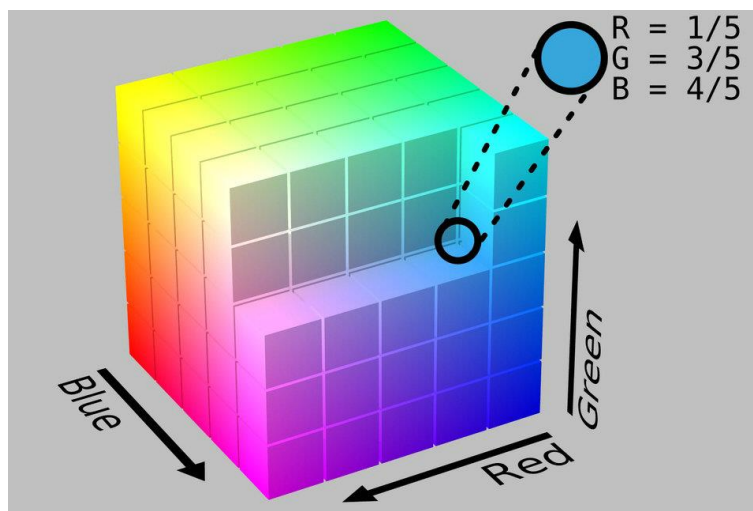


Figura 4-7. Espai de color RGB.

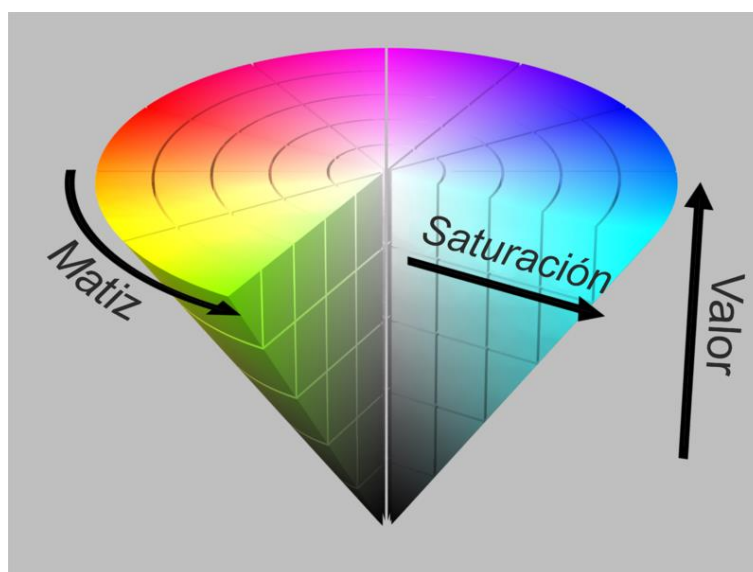


Figura 4-8. Espai de color HSV.

El diagrama de flux de l'algoritme a implementar és el següent:

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

Figura 4-9. Algoritme estratègia 1.

Les funcions d'Open CV que permeten implementar l'algoritme:

`split ()`: És una funció prèvia al càlcul de l'histograma, s'utilitza per a una qüestió tècnica de la llibreria. És necessari transformar el format en què és capturat el *frame*. Es convertirà el vector multidimensional que conté la informació de la imatge d'entrada en diversos vectors d'una sola dimensió. L'únic paràmetre necessari és la mateixa imatge.

`calcHist ()`: Una vegada realitzada la transformació del format ja es pot procedir a fer el càlcul de l'histograma. La llibreria disposa d'una funció específica per aquesta operació. És mostra un exemple de crida a la funció i l'explicació dels paràmetres necessaris.

- `Hist = cv2.calHist (images, channels, mask, histSize, ranges [, hist [, accumulate]])`
- `Images` fa referència a la imatge d'entrada prèviament modificada mitjançant la funció `split ()`.
- `Channels` fa referència a un número enter per indicar les dimensions del histograma a calcular.
- `Mask` és un paràmetre opcional que s'utilitza quan es vol calcular només l'histograma d'una part concreta de la imatge. Si es vol fer ús d'aquesta funcionalitat és necessari la creació de la màscara mitjançant altres funcions de la llibreria o bé de forma manual indicant els píxels que formen part de l'àrea d'interès.
- `HistSize` fa referència a un número enter per indicar la quantitat de *bins* del histograma.
- `Ranges` estableix els rangs de cada *bin* de l'histograma.
- `Accumulate` serveix per a poder obtenir un histograma on s'han utilitzat més d'una imatge d'entrada.



`compareHist ()`: És la funció de la llibreria que permet realitzar la comparació de dos histogrames per tal de poder establir la similitud entre dues imatges. La llibreria ofereix quatre mètodes diferents per a poder utilitzar la funció, els quals difereixen entre si per la mètrica que utilitzen per a d'establir les similituds entre els dos histogrames. És mostra un exemple de crida a la funció i l'explicació dels paràmetres necessaris.

- `returnVal = cv2.compareHist (H1, H2, method)`
- H1 fa referència al primer histograma que es vol utilitzar en la comparació.
- H2 fa referència al segon histograma que es vol utilitzar en la comparació.
- Method és un número enter per indicar la mètrica que és vol utilitzar per a comparar els dos histogrames.

Com a limitació de la funció, s'ha dir que funciona correctament fent comparacions d'histogrames de dimensió 3 o inferior. Per a dimensions superiors no és adequat la seva utilització. Per al que fa a les diferents mètriques per a realitzar la comparació, no s'entrarà en detalls matemàtics, ja que el treball té una orientació pràctica i s'utilitzaran totes les opcions disponibles en l'avaluació de l'algoritme per a d'establir quina és la més adequada.



4.5 Estratègia 2: detecció de color

La segona estratègia proposada farà ús del concepte d'espai de color exposat en l'apartat anterior. La idea d'aquesta estratègia és tractar de detectar un color específic en la imatge per tal de poder fer la detecció adequada. L'algoritme utilitzat és més simple i no fa ús de tantes funcions com l'anterior, fet que ha de millorar el temps d'execució del programa el qual és un paràmetre important en la detecció.

Per tal de realitzar la detecció del color desitjat s'utilitzarà una màscara que contindrà només certs valors delimitats per unes cotes en l'espai de color i descartarà tots els altres colors de la imatge.

Seguidament es realitzaran certes alteracions a la imatge per tal de poder disminuir al màxim el soroll que s'hagi filtrat, a la vegada que s'intentarà optimitzar la detecció del color especificat per tal d'obtenir la màxima fiabilitat i evitar falsos positius en els assajos.

Finalment s'aplicarà una segona màscara a la imatge a la qual se li han realitzat les alteracions i així poder fer un recompte del total píxels que es detecten amb el color desitjat i poder determinar si s'ha activat o no l'avís (*veure annex 2*).



El diagrama de flux de l'algoritme a implementar és el següent:

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

Figura 4-10. Algoritme estratègia 2.

Les funcions d'Open CV que permeten implementar l'algoritme:

`inRange ()`: Aquesta funció permet comprovar quins valors d'un vector d'entrada que estan dintre dels límits delimitats per certes cotes establertes en un espai de color. Retorna un vector on es delimiten els píxels que estan dintre del rang i els que no.

- `returnVal = cv2.inRange (src, lowerb, upperb)`
- `Src` fa referència a la imatge d'entrada que es vol analitzar.
- `Lowerb` fa referència a la cota inferior.
- `Upperb` fa referència a la cota superior.

`Bitwise_and ()`: Aquesta funció el que permet és fer una operació lògica AND entre dos vectors. Partint del vector que retorna la funció anterior és útil per a poder crear una màscara que només contingui els píxels que ha retornat la funció `inRage ()`.

- `returnVal = cv2.bitwise_and (src1, src2, mask)`
- `Src1` fa referència a la primera imatge d'entrada per a realitzar l'operació
- `Src 2` fa referència a la segona imatge d'entrada per a realitzar l'operació.
- `Mask` fe referència a la mascara que es vol aplicar a per a realitzar l'operació.



Erode (): Aquesta funció permet realitzar una transformació morfològica de la imatge. En concret realitza escaneig de la imatge amb un filtre denominat *kernel*, el qual té una mida específica. La funció que realitza és la d'eliminar tots els píxels de diferent valor que trobi durant l'escaneig i que tinguin una mida inferior al *kernel*. Per utilitzar aquesta funció, és necessari prèviament modificar la imatge per tal que tingui un format binari, és a dir blanc i negre. En el nostre cas concret, ja es realitza automàticament, en el retorn de la funció `bitwise_and ()`.

- `returnVal = cv2.erode (src, kernel, iterations)`
- `Src` fa referència a la imatge d'entrada.
- `Kernel` és un vector per especificar el tipus de filtre a utilitzar, es pot utilitzar un filtre predefinit.
- `Iterations` és un enter que fa referència al número de vegades repetides que es vol fer l'operació.

Dilatation (): és una altra funció de la llibreria que permet fer transformacions morfològiques, bàsicament realitza la operació contrària a la funció `erode ()`. Quan s'utilitzen conjuntament, l'objectiu és poder eliminar el soroll d'una imatge mitjançant `erode ()` i després poder restablir certs píxels d'interès amb la funció `dilatation ()`.

- `returnVal = cv2.dilate (src, kernel iterations)`
- Els paràmetres per a la utilització de la funció són els mateixos que en la funció `erode()`.

4.6 Avaluació de les estratègies de detecció

Per tal de poder analitzar la viabilitat de les dues estratègies proposades, es farà un anàlisis amb imatges reals dels assajos d'IDIADA. La recopilació de les dades necessàries es realitzarà mitjançant la gravació de vídeos durant els assajos d'un vehicle prototip.

El prototip ha de realitzar diverses proves per al seu procés de desenvolupament, s'equiparà amb el hardware utilitzat en aquest projecte es capturarà el *dashboard* durant els assajos. Els assajos planificats formen part d'un projecte comercial real d'IDIADA i engloben gran part de les condicions climàtiques i lumíniques que poden afectar el funcionament del sistema.

Una vegada recopilats els vídeos durant els assajos, es procedirà a validar les dues estratègies i veure la viabilitat. Com ja es tenen les gravacions tot aquest procés es pot realitzar de forma remota i sense necessitat d'utilitzar les pistes de proves, ja que el cost és molt elevat.

La validació en si es durà a terme mitjançant primer un anàlisis de certs *frames* concrets i després analitzant tota una gravació sencera per tal de poder comprovar el seu funcionament en una simulació tan realista com sigui possible.

Una vegada finalitzat el procés de validació i aprofitant que el mateix prototip romandrà a IDIADA al finalitzar les proves de desenvolupament. Es muntarà el sistema final i es faran les proves finals de validació a les pistes de proves. Aquest procés queda reflectint en l'apartat 6.1 de la memòria, on es mostra tot el procés d'implementació de la nova instrumentació en el vehicle, les proves realitzades amb el sistema en funcionament i el posterior anàlisi de les dades obtingudes.

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

Figura 4-11. Exemple frame capturat amb avís.

Avaluació estratègia 1: comparació d'histogrames

Com ja s'ha comentat es parteix de les dades recopilades i el primer pas és l'elecció d'una imatge de *template* per tal de poder calcular l'histograma i fer les posteriors comparacions dels diferents *frames* capturats.

L'elecció del *teemplate* és clau, la imatge seleccionada ha de ser de gran qualitat i lliure de soroll per tal de ser tan genèrica com sigui possible. L'elecció d'aquest *template* s'ha de fer cada vegada que hi hagi model nou de *dashboard* i està pensat per anar creant un base de dades dels diferents models.

No és un procés tan laboriós com pot semblar, ja que abans de començar els assajos es fan un conjunt de comprovacions dinàmiques a tots els vehicles i seria el moment adequat per a captura el *template*. També s'ha de comentar que gran part dels vehicles que formen part d'un mateix grup automobilístic comparteixen tecnologia i els símbols dels avisos als *dashboards* són idèntics. Per posar un exemple d'això es pot observar com en un avís de SEAT que pertany al grup VAG i un avís d'SKODA que pertany al mateix grup són iguals. Fet que facilita encara més el procés de recopilació de *templates* i redueix la mida de la base de dades que s'ha de generar per a poder utilitzar aquesta estratègia de detecció.

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

Figura 4-12. Avís visual SEAT i SKODA.

Per a tal de poder fer viable l'ús d'aquesta estratègia és necessària la creació d'una màscara que delimitarà l'àrea d'interès dels *frames* capturats. La mascara és necessària per a dos motius, al reduir la imatge a una zona d'interès més petita, el temps d'execució per al càlcul de l'histograma és redueix i a la vegada s'eliminen totes les altres fons d'interferències de color (altres símbols del *dashboard*).

En les següents imatges es mostren diferents histogrames obtinguts a partir del *frame* de la figura 4-12. Les gràfiques mostren en l'eix horitzontal el *bins* i en el vertical el número de píxels en cada *bin*.

No es vol fer un anàlisi detallat de cada gràfica, sinó mostrar les diferències entre un *frame* amb avís i un altre sense. I també com afecta la aplicació d'una màscara per delimitar l'àrea d'interès.

Tots els histogrames s'han calculat amb imatges de la mateixa mida, ja que sinó no es pot realitzar la comparació.

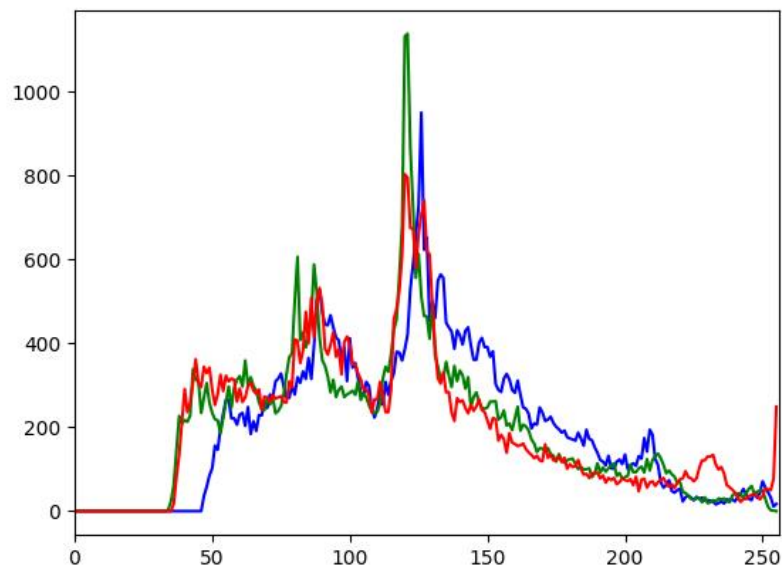


Figura 4-13. Histograma del figura 4-12 sense avís visual. Veure [3]

La figura 4-13 correspon al histograma sense avís visual, es a dir una imatge on tan sols apareix el fons en negre sense el símbol en vermell. Es pot veure com la majoria de píxels es concentren en els *bins* del centre. Correspon a la barreja de colors per fer el fons negre.

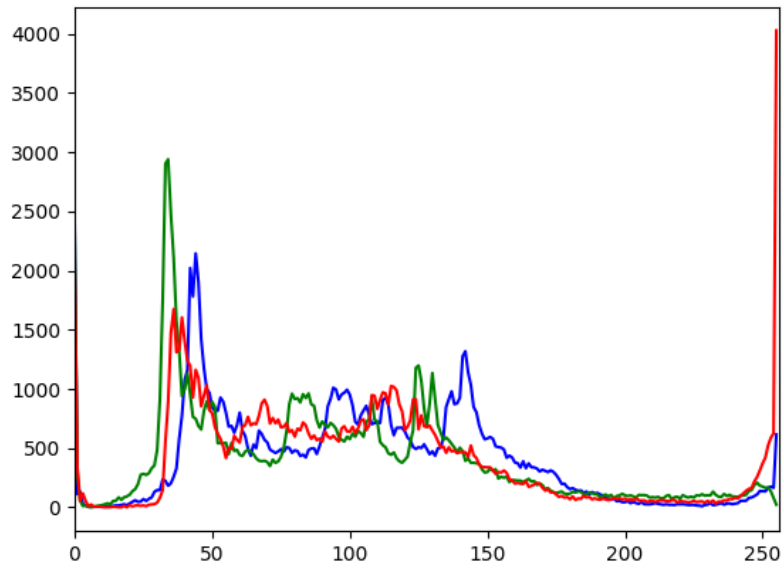


Figura 4-14. Histograma del figura 4-12 amb avís visual.

La figura 4-14 correspon al histograma amb avís visual i sense màscara. Es pot veure com en els *bins* superiors hi ha un pic de gairebé 4000 píxels vermells que correspon al avís visual. Tot i això, en la resta de *bins* encara s'observa una quantitat important de píxels. Això farà que la correlació amb l'histograma d'un *template* no sigui suficientment elevada i pot provocar falsos positius.

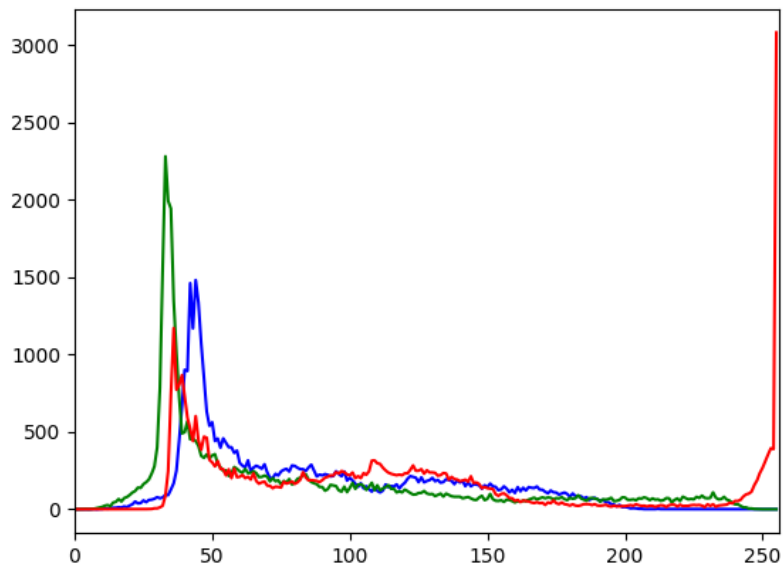


Figura 4-15. Histograma del figura 4-12 amb avís visual i màscara.

La figura 4-15 correspon al histograma amb avís visual i amb màscara per delimitar l'àrea d'interès en el símbol en vermell. Es pot observar com el pic de píxels vermells apareix en els mateixos *bins* que en histograma sense màscara, alhora que en tot l'altre rang de *bins* ha disminuït molt considerablement el número de píxels.

Això permet fer la comparació amb un *template* ideal de tal manera que la correlació entre els dos histogrames sigui elevada.

Es mostra una taula amb resultats obtinguts en la comparació d'histogrames utilitzant la mètrica 4 d'Open CV i mitjançant la descripció primer en un espai de color HSV i després en RGB.

La taula es un exemple concret del procediment que s'ha seguit per a comprovar la eficàcia de la estratègia. Comparant la figura 4-16 com a *template* i la figura 4-12 amb una màscara aplicada.

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

Figura 4-16. Template utilitzat en la taula d'exemple.

HSV	RGB
Mètrica 4	Mètrica 4
0.83 (83 %)	0,9 (90 %)

Figura 4-17. Exemple de concordança entre histogrames. Veure [4]

Per a fer les comparacions s'ha utilitzat la mètrica 4 de Open CV, la qual retorna un resultat de les correlacions dels histogrames en tant per un. Així doncs en aquest exemple concret es veu com utilitzant aquesta estratègia s'obtenen percentatges elevats de correlació entre els histogrames. Sent una mica millor si es fa servir una descripció en espai RGB, que en totes les proves realitzades s'ha comprovat que és una descripció millor, ja que es veu menys afectada per possibles reflexos de la llum de l'ambient.

Avaluació estratègia 2: detecció de color

Per a l'avaluació de la segona estratègia de detecció es partiran de dades similars a les utilitzades en l'avaluació de la primera estratègia.

Per a la implementació d'aquest sistema de detecció el procediment és més senzill, ja que no s'ha de generar cap zona d'interès de forma manual. El que si és necessari com en la primera estratègia és una recopilació de vídeo durant les comprovacions prèvies.

L'estratègia consisteix en agafar un *frame* on aparegui un avís visual i mitjançant diverses aplicacions comercials establir el codi de color en un espai RGB del avís. Un exemple d'una aplicació comercial poden ser Pixel Picker o Color Picker entre moltes altres que és troben disponibles en format APP tant per a descarregar com per utilitzar directament des de un navegador Web. Aquest pas és necessari per a establir les cotes inferior i superior en l'espai de color i així poder filtrar només el color corresponent al avís visual.

Una vegada feta l'elecció de les cotes de l'espai de color es pot aplicar la primera màscara al *frame* capturat. En alguns exemples es pot veure com apareix l'avís visual juntament amb cert soroll procedent d'altres fons.

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

Figura 4-18. Frame sense avís amb la primera màscara aplicada. Veure [5]

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

Figura 4-19. Frame sense avís amb la primera màscara aplicada..

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

Figura 4-20. Frame amb avís amb la primera màscara aplicada.

En les figures superiors 4-18, 4-19 i 4-20 es poden veure tres exemples del resultat del primer filtre. En la part esquerra de la imatge és veu el *frame* capturat per la càmera i a la part dreta la imatge amb el filtre aplicat. En la figura 4-18 i 4-19 es pot veure cert soroll no desitjat en la part inferior de la imatge.

Per tal de poder eliminar el soroll i millorar la fiabilitat del sistema s'apliquen les transformacions morfològiques a la imatge. Mitjançant les funcions *erode* i *dilatation*.

Finalment es torna a aplicar una segona màscara per tal d'obtenir una detecció del color el més eficaç possible. Tot el procediment permet augmentar el valor de píxels que s'estableix per tal d'establir que el *frame* capturat correspon a un avís de perill. Sent per tant una forma d'eliminar gairebé per complet falsos pastius durant els assajos.

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

Figura 4-21. Frame amb avís resultat final.

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

Figura 4-22. Frame amb avís resultat final.

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

Figura 4-23. Frame sense avís resultat final.

En les figures 4-21, 4-22 i 4-23 es poden veure les imatges una vegada realitzats els dos filtres i les transformacions morfològiques. S'ha de aclarir que aquestes imatges es mostren a mode de aclariment i quan el programa s'està executant no es veu res per pantalla per a optimitzar el temps d'execució.

Les figures 4-21 i 4-22 corresponen a un avís de col·lisió amb cotxe i col·lisió amb vianant respectivament. I es veu com la forma de les imatges s'ha modificat a causa de les transformacions morfològiques fet que no és important ja que no es fa un anàlisi de la forma. A canvi s'ha eliminat tot el soroll que es podia veure anteriorment i hem augmentat els píxels detectats al voltant de un 50 %.

Aplicant només el primer filtre tenim 23 píxels detectats en la figura 4-19 amb el segon filtre 0 píxels detectats en la figura 4-23. Ja que eren soroll.

Aplicant només el primer filtre tenim 351 píxels detectats en la figura 4-19 amb el segon filtre 692 píxels detectats en la figura 4-23.

Aplicant només el primer filtre tenim 792 píxels detectats en la figura 4-20 amb el segon 1603 detectats píxels en la figura 4-21.

Elecció final de la estratègia a implementar

Una vegada realitzada les avaluacions de les dues estratègies és presenten els resultats al departament ADAS.

Les dues estratègies són vàlides per a la detecció dels avisos visuals. De una manera senzilla s'ha obtingut una concordança elevada en la comparació d'histogrames i a la vegada la segona estratègia permet implementar una metodologia encara més senzilla i també eficaç.

En al comparació d'histogrames s'han obtingut concordançes del voltant del 80 i 90 % aplicant la màscara per delimitar l'àrea d'interès. I en la detecció de color s'ha pogut filtrar només el color del avís, eliminant el soroll per complert amb prou marge per classificar quan hi ha avís o no sense generar falsos positius.

Finalment des de el departament ADAS es decideix implementar la segona estratègia pels següents motius:

- El temps d'execució és menor respecte a la comparació d'histogrames, 50 % més ràpid.
- Presenta una millor resposta respecte reflexos de la llum ambient, és important per assajos a primera hora del matí o a última hora de la tarda.
- És més fàcil d'utilitzar, només cal col·locar la càmera i executar el programa. En la comparació d'histogrames, s'ha de fer una selecció del àrea d'interès mitjançant el ratolí i si es mou la càmera duran els assajos s'ha de tornar a seleccionar.

5 Implementació de la comunicació CAN i *data logger*

5.1 Bus CAN

CAN bus (*Controller Area Network*) és un bus de camp principalment utilitzat en automoció. Permet la comunicació entre diferents nodes, que en el cas dels automòbils es coneixen com a ECUs. Cadascun d'aquest node genera informació que ha de ser compartida amb altres nodes.

El CAN bus permet la comunicació dels nodes mitjançant una capa física de només dos cables, CAN High i CAN Low. Els missatges enviats per cadascun dels nodes són rebuts per tots els altres nodes, que en funció del tipus de missatge en faran la lectura o l'ignoraran.

Els principals avantatges del CAN bus són:

- Simplicitat i baix cost, permet la comunicació amb una capa física molt simple que permet reduir cablejat.
- Robust respecte al soroll electromagnètic, és adequat per als sistemes crítics.
- Eficient, els missatges estan organitzats en funció de la prioritat de cada trama, un missatge amb prioritat elevada rep accés ràpid al bus sense causar interrupcions als altres nodes.

Historia del CAN bus:

- Abans del CAN bus: Les ECUs es comunicaven mitjançant una sistema punt a punt.
- 1986: Bosch crea CAN bus.
- 1991: Bosch desenvolupa CAN bus 2.0 (CAN 2.0 A: 11 bits i CAN 2.0 B: 29 bits).
- 1993: CAN bus es acceptat com a estàndard internacional (ISO 11898).
- 2003: ISO 11898 es obligatòria per a tots el vehicles de producció.
- 2012: Bosch desenvolupa CAN FD (flexible data rate).
- 2015: CAN FD es estandarditzat (ISO 11898-1).

Avui dia el CAN bus és el sistema de comunicació estàndard per als sistemes crítics de cotxes, camions, avions i maquinària pesada.

La comunicació via CAN bus es realitza mitjançant trames de dades. En funció del tipus de protocol utilitzat el format de la trama varia lleugerament. CAN 2.0 A utilitza un format de trama amb un identificador d'11 bits i CAN 2.0 B utilitza un format de trama amb un identificador de 29 bits. Tots els altres elements de la trama són iguals per als dos protocols.

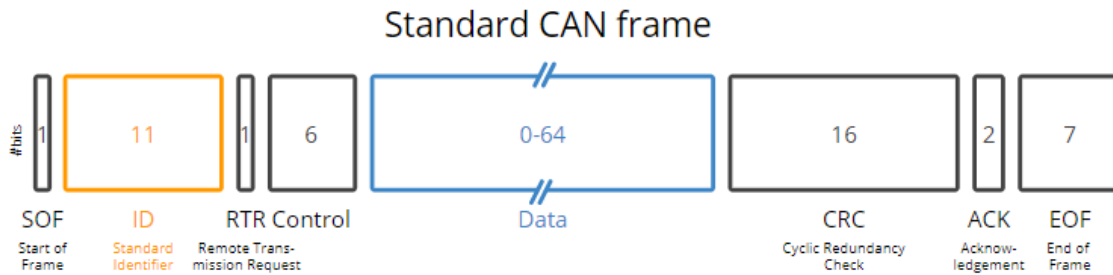


Figura 5-1. Format d'un missatge CAN. Veure [6].

- SOF: Inici de trama correspon a un '0 dominant', indica als nodes on s'inicia la trama.
- ID: És l'identificador del node transmissor. Com més petit es el valor més gran és la prioritat.
- RTR: Indica si un node esta transmeten informació o si ve sol·licita informació d'un altre node.
- Control: Conté un bit de control per a identificar si es fa servir un protocol CAN 2.0 A o CAN 2.0 B, també conte 4 bits que indiquen la longitud de les dades (0 a 8 bytes de dades).
- Data: Conté la informació codificada que es vol transmetre.
- ACK: Conté 2 bits que indiquen si el node destinatari ha rebut la trama correctament.
- EOF: Indica el final de la trama.

5.2 Arxius DBC

Un arxiu DBC és un arxiu de text que conté la informació necessària per descodificar la informació transmesa per un bus CAN i transforma-la en informació intel·ligible per a les persones.

Al rebre una trama de bus CAN es pot observar la informació mitjançant diversos softwares i el que es pot observar és el següent:

CAN ID	Data bytes
0CF00400	FF FF FF 68 13 FF FF FF

Figura 5-2. Exemple missatge CAN raw data.

La funció d'un arxiu DBC és establir les normes per tal de descodificar la trama d'informació en paràmetres i senyals. Una possible descodificació del missatge superior pot ser:

Message	Signal	Value	Unit
EEC1	EngineSpeed	621	rpm

Figura 5-3. Exemple missatge CAN amb descodificació de la DBC.

Un exemple d'arxiu DBC:

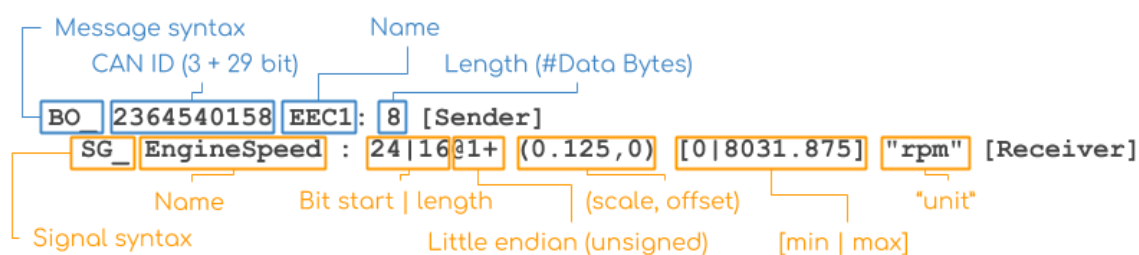


Figura 5-4. Exemple senzill arxiu DBC. Veure [7].

- En blau s'indica la sintaxi per a la descodificació del missatge.
- En taronja s'indica la sintaxi per a la descodificació dels senyals

Veure annex 3 per al arxiu DBC que s'ha creat.

5.3 Arxius MDF4

MDF (*Measurement Data Format*) és un format estàndard per a enregistrar CAN i CAN FD. MDF va ser creat als anys 90 en una col·laboració de Vector i Bosch, al 2002 va passar a ser de domini públic. L'última versió denominada MDF4 va ser llançada al mercat el 2014.

Els principals avantatges del format MDF són:

- Obert i estandarditzat, és un format que està suportat per la majoria de *data loggers* de bus CAN.
- De ràpida escriptura i lectura, característica essencial per quan s'utilitza una alta velocitat de transferència de dades.
- Suporta compressió per tant millora la portabilitat i transmissió dels arxius.
- Hi ha APIs que permeten la visualització, modificació i exportació a altres formats dels arxius MDF.

En la següent figura es pot observar l'estructura que tenen tots els arxius MDF 4 i que es mostra en les APIs quan es vol accedir a alguna dada.

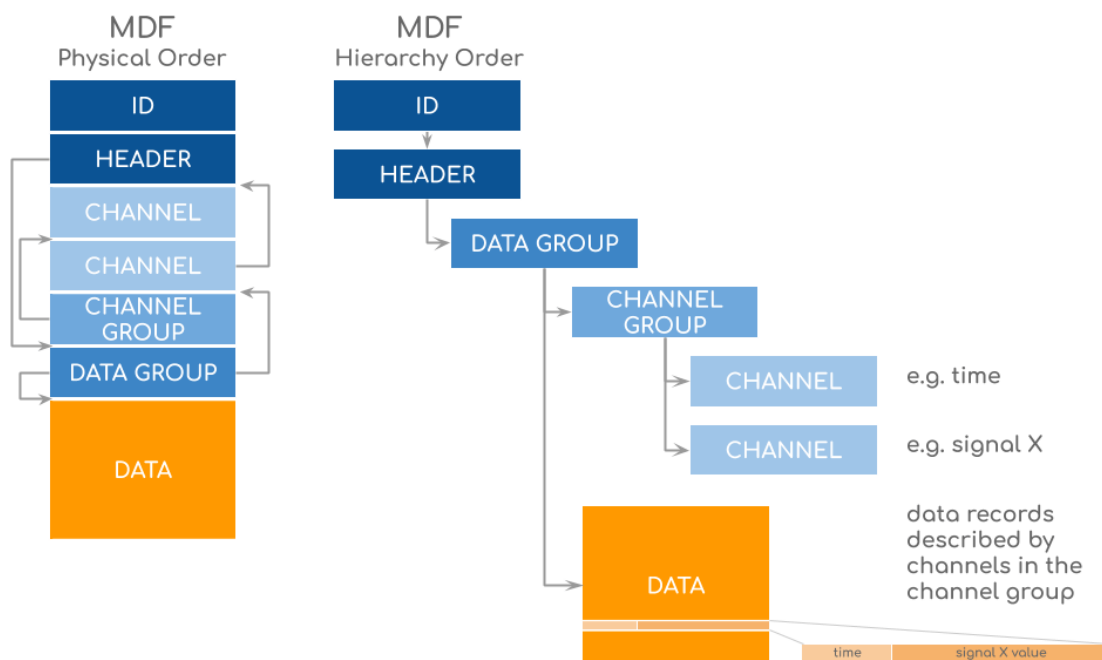


Figura 5-4. Format arxius MF4. Veure [8].

5.4 Hardware

RS485 CAN HAT

Es tracta d'una placa d'extensió per a Raspberry Pi. La qual permet la comunicació CAN i amb protocol RS 485 (*veure annex 4*). Les seves principals característiques són:

- Compatibilitat amb tots els models de Raspberry Pi.
- Comunicació CAN mitjançant el controlador MCP 2515 i comunicació amb Raspberry Pi per SPI.
- Comunicació protocol RS 485 mitjançant el transductor SP 3485 i comunicació amb Raspberry Pi per UART.
- Té disponible una llibreria de Python per facilitar la implementació de les comunicacions.
- La comunicació física entre el HAT i Raspberry es fa mitjançant els pins d'entrada i sortida.



Figura 5-5. CAN HAT.

Per a gestió de la comunicació CAN la placa d'expansió fa ús de dos integrats. MCP 2515 que és el controlador CAN i permet la comunicació SPI amb Raspberry Pi i SN 65 HVD 230 que fa les funcions de transductor per generar el senyal elèctrica per a la capa física del bus.

MCP 2515

És un controlador de CAN desenvolupat per a simplificar la implementació de comunicació CAN. Està format per tres blocs.

- El mòdul CAN gestiona totes les funcions de recepció i transmissió de missatges al bus CAN. Els missatges carregen primer als buffers i la transmissió s'inicia mitjançant una gestió dels registres de control per SPI. Per a la recepció de missatges del bus, es comprova si hi ha errors al missatge i s'apliquen els filtres definits per tal d'acceptar o no el missatge.
- El mòdul de control lògic controla la configuració del MCP 2515 i permet la interacció dels diferents blocs.
- El mòdul SPI permet la lectura i escriptura dels registres per part d'un dispositiu extern.

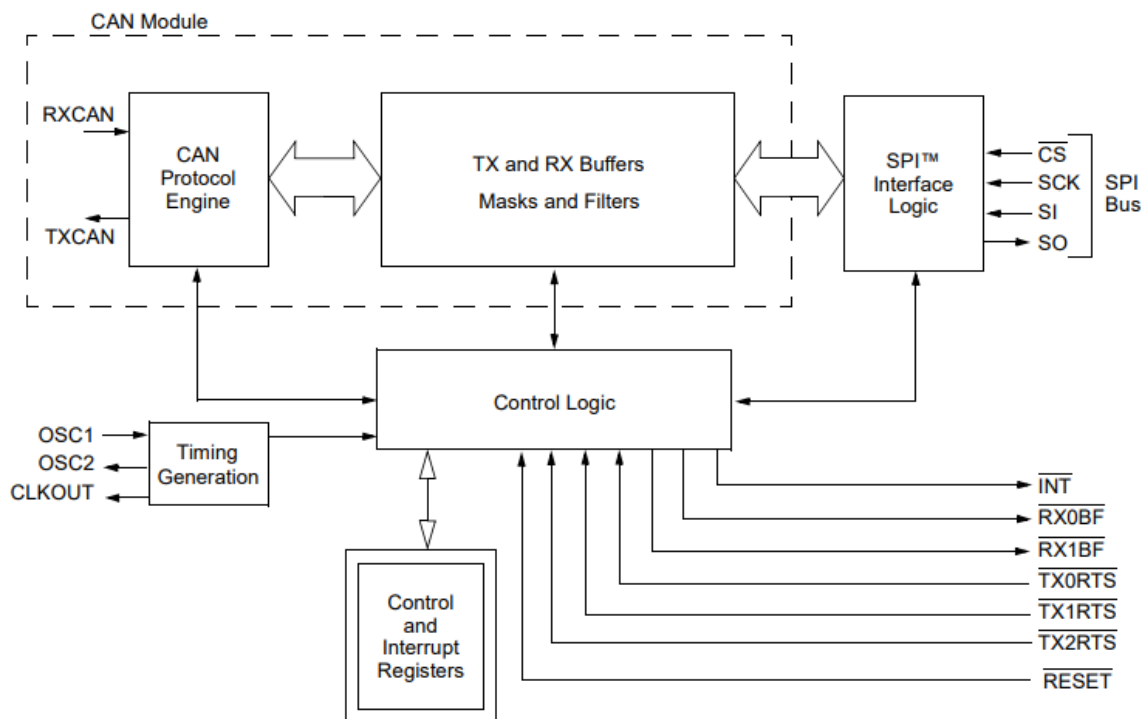


Figura 5-6. Funcionament MCP 2515. Veure [9].

SN 65 HVD 230

El transductor està dissenyat per tal d'implementar la ISO 11898 per a la comunicació CAN. Permet la transmissió de un senyal diferencial fins a velocitats d'1 Mbps. Actua com una porta d'accés a la capa física de bus.

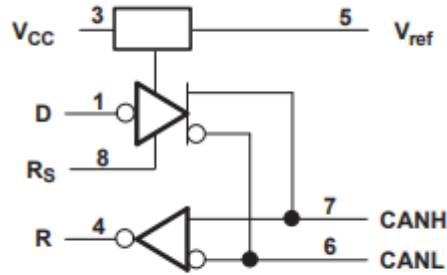


Figura 5-7. Estructura SN 65 HVD 230.

VECTOR VN 1640

Permet la lectura de diferents protocols utilitzats en l'automoció, tals com CAN i LIN.



Figura 5-8. Data Logger..

5.5 Software

Python CAN API

És una API creada per Dynamics Control. Permet implementar sistemes de comunicació CAN amb poques línies de codi.

Suporta diverses interfases, Socket CAN va ser creada per Rose Lu el 2011. Aquest proporciona una eina per a controlar, gestionar i monitoritzar dispositius CAN.

Altres interfases de l'API són:

- CAN over serial
- USB 2 CAN
- NI-CAN
- CANalyst - II

Llibreria BCM 2835

BMC 2835 i versions posteriors són els processadors utilitzats en Raspberry Pi. La llibreria permet gestionar els perifèrics connectats al processador. En aquest treball s'utilitza juntament amb l'API de CAN per tal de poder realitzar la comunicació SPI entre el processador i el controlador MCP 2515.

6 Assajos de validació a les pistes de proves

6.1 Instrumentació del vehicle i configuració del *data logger*

En la validació final del sistema es realitzarà tot el procediment que es seguiria normalment en una prova de desenvolupament d'un prototip, a més en ser la primera vegada que s'implementa es faran certes accions de configuració, tant de la comunicació CAN en el HAT com per configurar correctament el hardware encarregat d'enregistrar les dades que va rebent dels diferents nodes CAN tant del propi vehicle com de la instrumentació implementada.

Com ja s'ha mencionat en l'apartat 4.6, tant el procés de recopilació de les dades per tal a fer les avaluacions inicials de les estratègies proposades com les posteriors proves en pistes es faran mitjançant un prototip que està realitzant les seves proves desenvolupament.

Això es degut a que el fabricant ens ha permès accedir al CAN del vehicle i a més ens ha proporcionat els arxius DBC per tal de poder convertir els senyals obtingudes a gràfiques intel·ligibles. Aquest fet ens permetrà comprovar si les deteccions que fa el detector són correctes o no i el temps de detecció que té. Ja que podem comprovar en quin moment l'ECU envia el senyal per activar l'avís visual i comparar-lo amb el CAN que es genera a partir de detector.

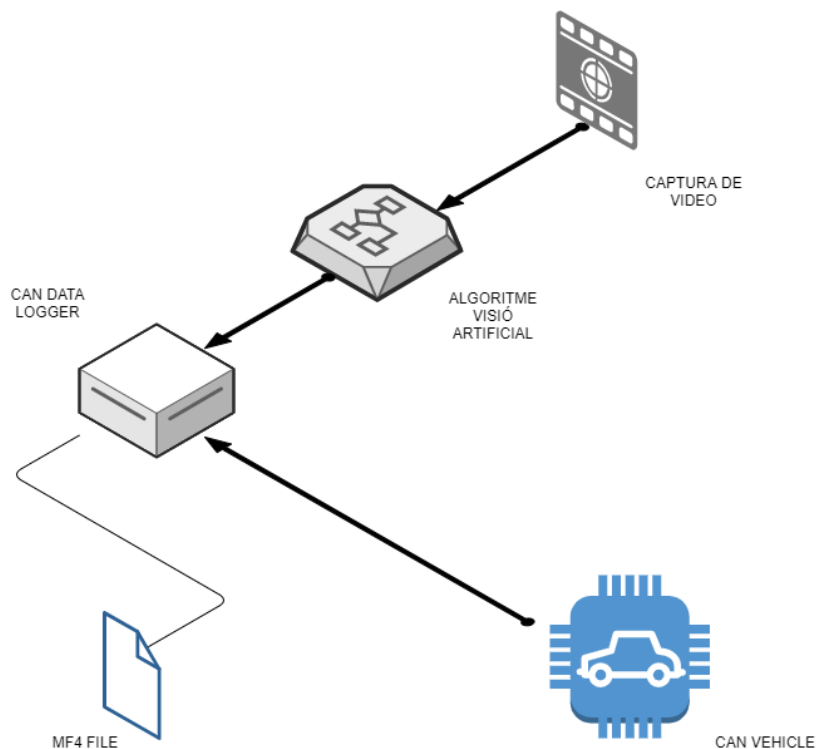


Figura 6-1. Organització CAN.

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

Figura 6-2. Vehicle instrumentat per fer els assajos.

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

Figura 6-3. Raspberry Pi amb CAN HAT i connectors RS 485.

Configuració CAN HAT

Per tal de poder establir la comunicació CAN cal configurar el HAT i instal·lar les llibreries necessàries.

- Llibreria BCM 2835
- Llibreria wiring Pi

Tota la configuració és pot realitzar mitjançant el terminal de Linux. El primer pas que s'ha de fer és accedir al propi arxiu de configuració de la BIOS. Per poder accedir aquest arxiu s'ha de escriure al terminal el següent comandament:

- `sudo vi /boot/config.txt`

Al introduir la aquesta comandament apareix una pestanya amb diverses configuracions de la BIOS, és important no modificar res, ja que sinó Raspberry deixarà de funcionar correctament. Una vegada oberta la pestanya s'ha d'escriure el següent:

- `dtparam=spi=on`
- `dtoverlay=mcp2515can0,oscillator=8000000,interrupt=25,spifrequency=1000000`

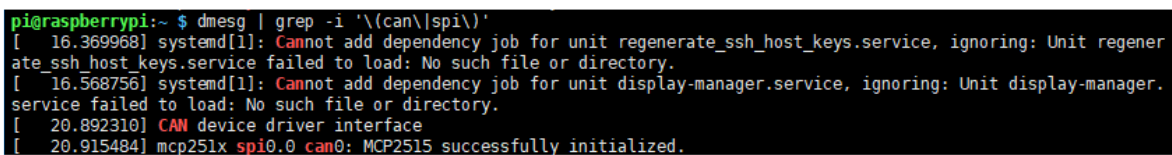
Amb aquestes dues línies es s'activa la comunicació SPI de Raspberry i es configura la freqüència de comunicació.

Per tal de comprovar si les modificacions realitzades funcionen correctament, s'introdueix el següent comandament:

- `dmesg | grep -i '\(can\|spi\)'`

Si les modificacions han estat correctes apareixerà la següent informació per pantalla.

```
pi@raspberrypi:~$ dmesg | grep -i '\(can\|spi\)'
```

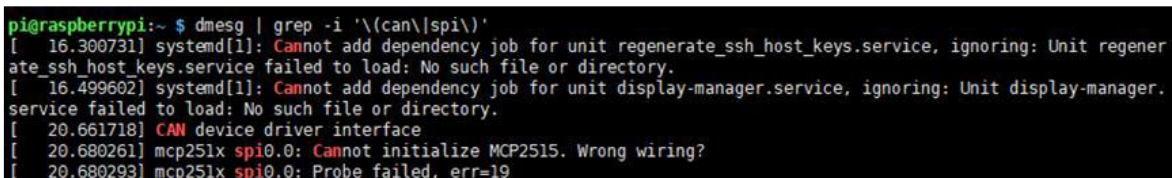


[16.369968] systemd[1]: Cannot add dependency job for unit regenerate_ssh_host_keys.service, ignoring: Unit regenerate_ssh_host_keys.service failed to load: No such file or directory.
[16.568756] systemd[1]: Cannot add dependency job for unit display-manager.service, ignoring: Unit display-manager.service failed to load: No such file or directory.
[20.892310] CAN device driver interface
[20.915484] mcp251x spi0.0 can0: MCP2515 successfully initialized.

Figura 6-4. Missatge instal·lació correcte.

Sinó apareixerà un error indicant un codi.

```
pi@raspberrypi:~$ dmesg | grep -i '\(can\|spi\)'
```



[16.300731] systemd[1]: Cannot add dependency job for unit regenerate_ssh_host_keys.service, ignoring: Unit regenerate_ssh_host_keys.service failed to load: No such file or directory.
[16.499602] systemd[1]: Cannot add dependency job for unit display-manager.service, ignoring: Unit display-manager.service failed to load: No such file or directory.
[20.661718] CAN device driver interface
[20.680261] mcp251x spi0.0: Cannot initialize MCP2515. Wrong wiring?
[20.680293] mcp251x spi0.0: Probe failed, err=19

Figura 6-5. Missatge instal·lació errònia.

Configuració VECTOR VN 1640

Per a fer la recopilació dels CAN es necessari configurar el *data logger*, aquest té un software associat que permet crear un projecte i fer totes les modificacions pertinents. El software en qüestió es denomina CANape.

A continuació es mostren els passos en la configuració:

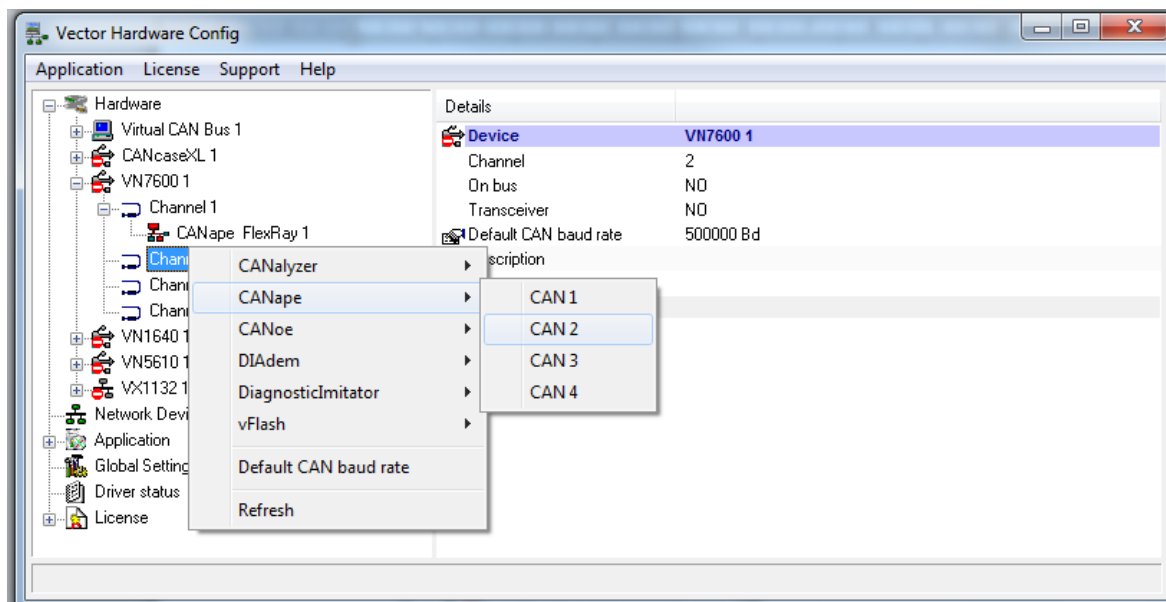


Figura 6-6. Configuració port connexió.

- Es configura el port on es connectarà el CAN, el dispositiu en té 4 i s'ha d'especificar quin s'utilitzarà.

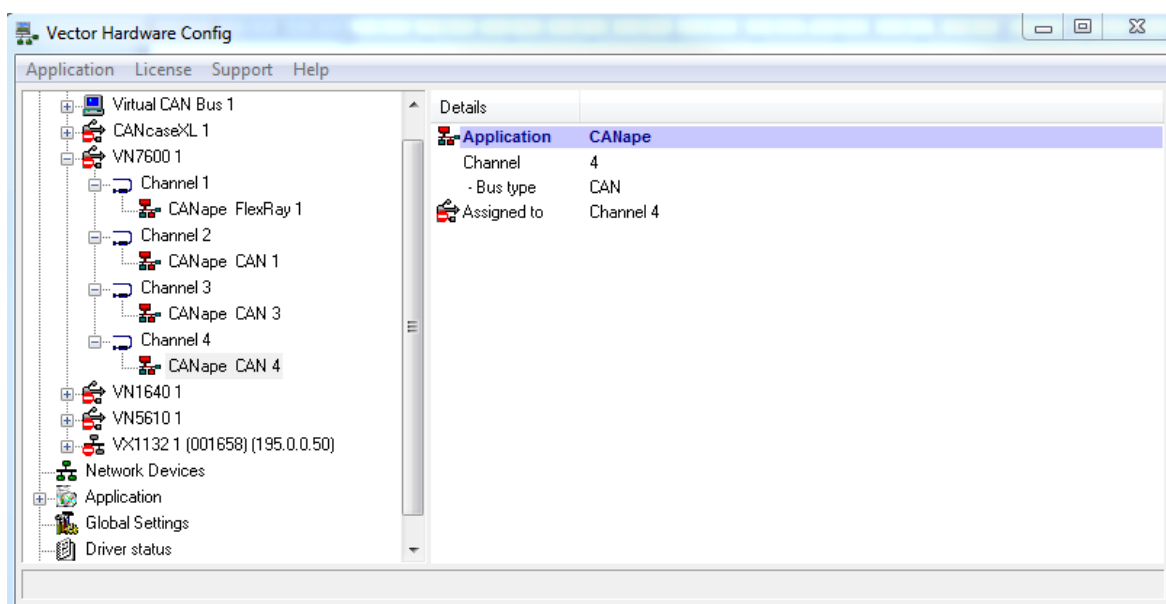


Figura 6-7. Configuració port connexió.

- Si s'ha realitzat correctament, es pot veure la imatge superior.

- El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

Figura 6-8. Creació projecte.

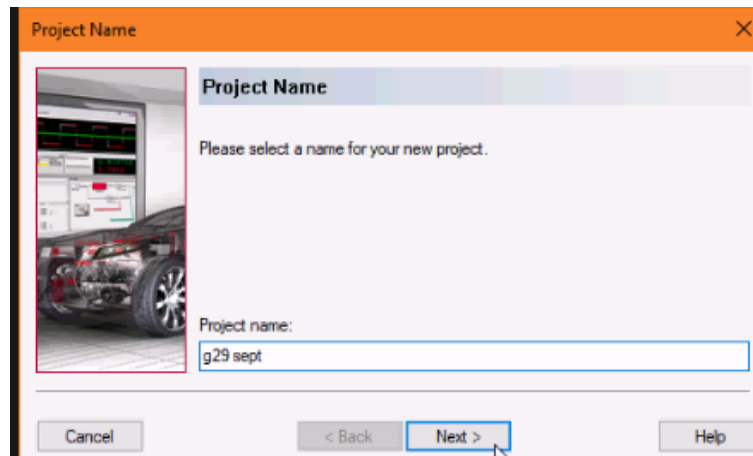


Figura 6-9. Creació projecte.

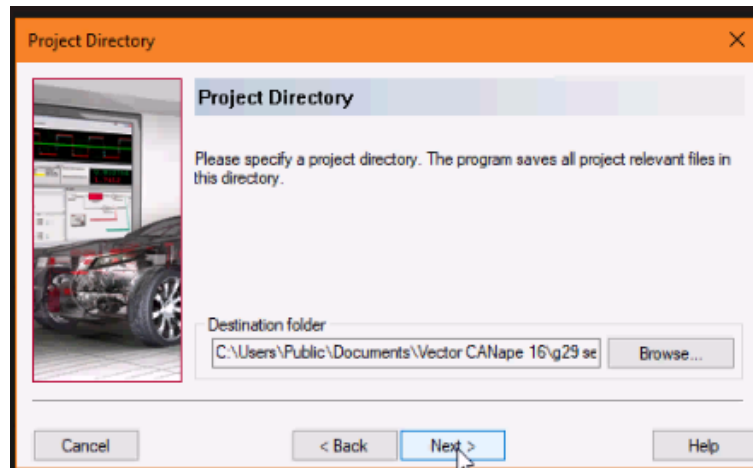


Figura 6-10. Creació projecte.

- Per a poder enregistrar les dades s'ha de crear un projecte nou. S'estableix el nom i la ruta.

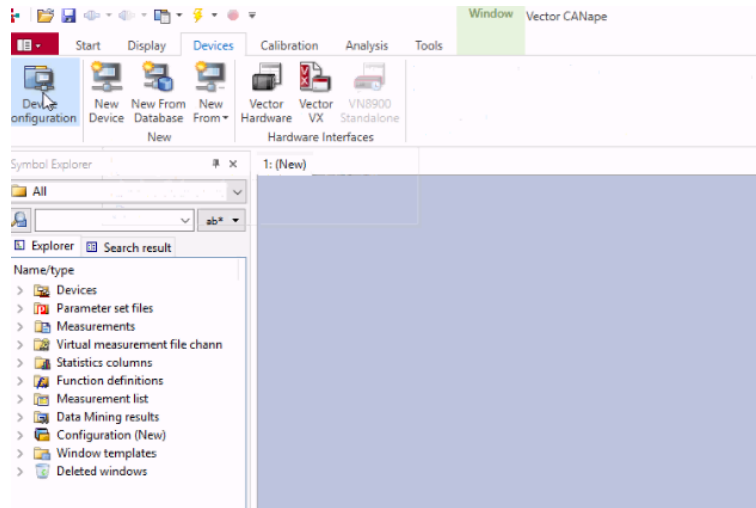


Figura 6-11. Creació projecte.

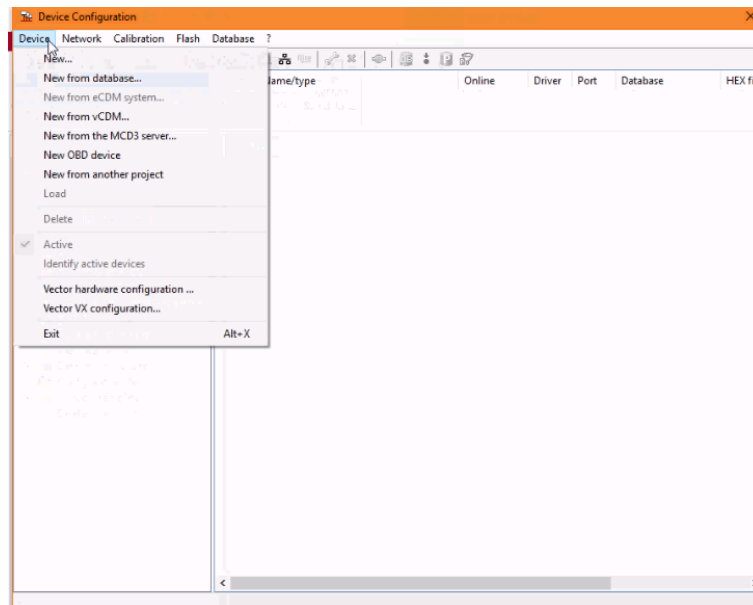


Figura 6-12. Creació projecte.

- Una vegada crear el projecte i es pot accedir a la configuració del procés d'enregistrament.

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

Figura 6-13. Creació projecte.

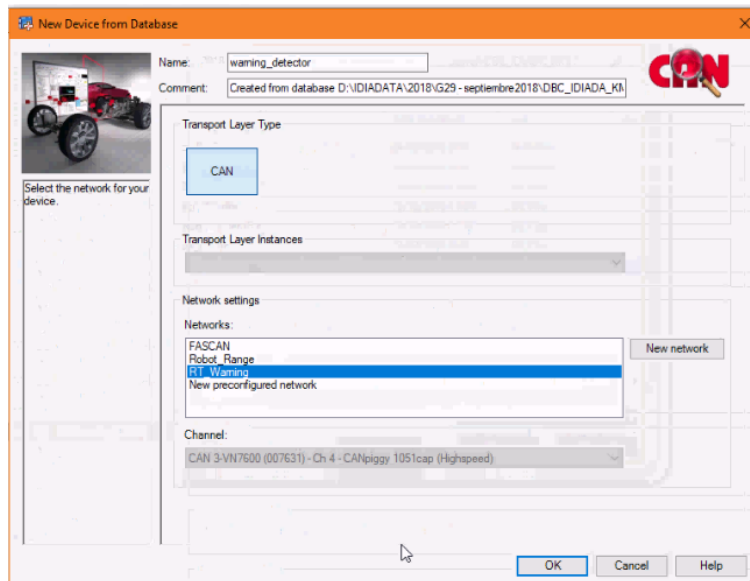


Figura 6-14. Creació projecte.

- En la configuració és carrega l'arxiu DBC per tal de poder generar el posterior arxiu MFD 4 de les dades rebudes.



6.2 Assaig de validació 1 (dia pluja amb poca lluminositat)

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

Taula 6-1. Resultats assaig 1.



6.3 Assaig de validació 2 (assajos a primera hora matí)

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

Taula 6-2. Resultats assaig 2.

6.4 Assaig de validació 3 (assajos tarda)

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

6.5 Anàlisi dades assajos

S'han realitzat assajos en tres dies diferents per tal de poder analitzar el sistema de visió artificial en diverses condicions lumíniques. En quan al anàlisi dels resultats es pot dir que ha funcionat correctament. No hi hagut cap fals positiu en cap intent i en tots ha detectat el senyal.

L'únic inconvenient que ha sorgit és que quan el sistema es calenta el temps d'execució augmenta en 150 ms aproximadament. És un problema que es pot observar en els resultats del primer dia d'assajos. En la realització dels assajos al adonar-nos que els temps d'execució augmentava quan feia estona que estava funcionant es va utilitzar el climatitzador del cotxe per refredar Raspberry Pi i els temps d'execució van tornar a valors normals. Es té previst fer una carcassa amb impressió 3D on s'implementarà refrigeració per evitar aquest problema.

Pel que fa als temps de detecció s'ha de realitzar un aclariment. El requeriment era fer la detecció en 100 ms i els resultats de les taules donen a entendre que no s'ha complert l'objectiu.

S'ha de dir que els temps de les taules es l'obtingut de la diferència entre la senyal generada per el sistema de visió artificial i la senyal de l'ECU del vehicle. Segons dades que ha facilitat el fabricant del vehicle entre la senyal de l'ECU i la visualització del avís al *dashboard* hi ha un retard de 150 ms. Per tant en els temps de les taules s'ha de restar aquest retard i la detecció s'executa en els 100 ms que es requerien.

En la figura 6-15 es pot veure una de les gràfiques dels arxius MF4 generats. Hi apareixen dues senyals, la senyal blava és la senyal de la ECU del vehicle i la senyal taronja és la generada per el sistema de visió artificial.

La senyal de l'ECU és una senyal continua que es manté constant durant tot el temps en que apareix l'avís. En la senyal del sistema de visió artificial es pot observar un tren de polsos, això es degut a que la senyal en el *dashboard* s'encén i s'apaga periòdicament per tal de cridar més l'atenció del conductor.

S'ha realitzat la comparativa directament amb la senyal de la ECU perquè és la que ha donat permís el fabricant per accedir i ens ha facilitat la DBC. Els fabricants són molt reticents a deixar accedir a senyals internes del cotxe.

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

Figura 6-15. Senyals ECU vehicle i sistema de visió artificial superposats.

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

Figura 6-16. Senyal ECU vehicle.

En la figura 6-16 es pot veure la senyal de la ECU del vehicle quan s'activa i s'indica l'instant de temps en que hi ha el flanc de pujada.

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

Figura 6-17. Senyal sistema visió artificial.

En la figura 6-17 es pot veure la senyal del sistema de visió artificial quan s'activa i s'indica l'instant de temps del primer flanc de pujada. La diferencia entre les senyals de la figura 6-16 i 6-17 és de 250 ms. Sent per tant el temps que es veu reflectit en les taules d'assajos.

7 Conclusions

En aquest treball s'ha realitzat un sistema de visió artificial destinat a formar part de la instrumentació que utilitza el departament ADAS d'IDIADA.

S'ha plantejat dues estratègies diferents per tal de poder complir els requeriments i s'han recopilat les dades necessàries directament en la pista de proves. Al avaluar les estratègies proposades s'ha descartat la estratègia basada en comparació d'histogrames i s'ha triat l'estratègia basada en detecció de color, ja que no només era més eficient sinó també més fàcil d'implementar i és pot utilitzar per a altres tipus de deteccions.

Una vegada triada la estratègia final de detecció, s'ha desenvolupat la comunicació del sistema amb el *data logger* que utilitzen en els assajos. S'ha creat un arxiu DBC per poder extreure les dades i poder visualitzar-les. I finalment s'ha pogut validar el sistema.

Pel que fa al rendiment del sistema és pot dir que funciona correctament, ja que compleix tots els requeriments inicials del projecte. Els aspectes més destacats són la facilitat d'ús, el baix cost, la velocitat de detecció i que no genera falsos positius.

Com ha projecte de final de grau m'ha servit per tal de aprendre nous conceptes i el funcionament de sistemes que desconeixia. Com a exemple d'això, es pot dir que s'ha après els conceptes bàsics de programació en Python, ha servit com una introducció a la visió artificial i a com fer el desenvolupament d'un sistema de baix cost, amb els dispositius disponibles en el mercat, tot complint els requeriments establerts i sobreposant-se a les limitacions que presenten els elements disponibles per pressupost.

En definitiva ha estat un procés d'aprenentatge molt profitós. Finalment fora de l'abast d'aquest treball queda la realització mitjançant impressió 3D d'una carcassa de protecció per al sistema i la inclusió d'una refrigeració addicional per tal d'assegurar el rendiment de Raspberry Pi.



8 Annexes

8.1 Annex 1: Comparació d'histogrames

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.



8.2 Annex 2: Detecció de color

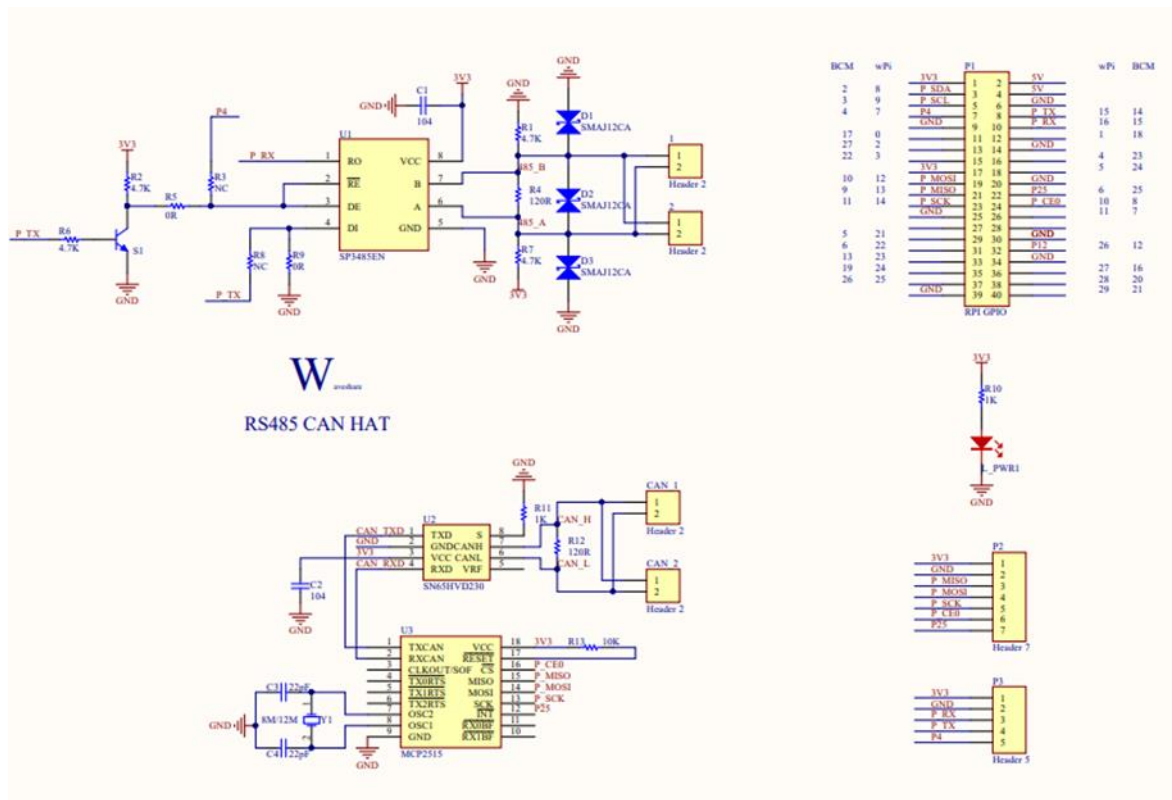
El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.



8.3 Annex 3: Arxiu DBC

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

8.4 Annex 4: Esquema CAN HAT





8.5 Annex 5: Script Final

El contingut d'aquesta figura s'ha retirat per l'acord de confidencialitat.

9 Referències

- [1] Open CV. “Histograms : Find, Plot, Analyze”. Internet Publication. https://docs.opencv.org/master/d1/db7/tutorial_py_histogram_begins.html. [Consulta online] 20-10-2020.
- [2] Cambridge in Colours. “Camera Histograms : Tones & Contrast”. Internet Publication. <https://www.cambridgeincolour.com/tutorials/histograms1.htm>. [Consulta online] 20-10-2020.
- [3] Open CV. “Histogram Calculation”. Internet Publication. https://docs.opencv.org/3.4/d8/dbc/tutorial_histogram_calculation.html. [Consulta online] 23-10-2020.
- [4] Open CV. “Histogram Comparison”. Internet Publication. https://docs.opencv.org/3.4/d8/dc8/tutorial_histogram_comparison.html. [Consulta online] 23-10-2020.
- [5] Open CV. “Thresholding Operations”. Internet Publication. https://docs.opencv.org/3.4/da/d97/tutorial_threshold_inRange.html- [Consulta online] 10 11-2020.
- [6] CSS Electronics. “CAN Bus Explained”. Internet Publication. <https://www.csselectronics.com/screen/page/simple-intro-to-can-bus> [Consulta online] 15-12-2020.
- [7] CSS Electronics. “CAN DBC Explained”. Internet Publication. <https://www.csselectronics.com/screen/page/can-dbc-file-database-intro> [Consulta online] 15-12-2020.
- [8] CSS Electronics. “MF4 Explained”. Internet Publication. <https://www.csselectronics.com/screen/page/mdf4-measurement-data-format> [Consulta online] 15-12-2020.
- [9] Waveshare. “RS 485 CAN HAT WIKI”. Internet Publication. https://www.waveshare.com/wiki/RS485_CAN_HAT. [Consulta online] 10-10-2020.