



AJUSTE AUTOMÁTICO DE ESPEJOS RETROVISORES POR POSICIONAMIENTO OCULAR DEL CONDUCTOR

48087 Mira, María Belén
51440 Hurovitz, Sasha Ivan

8 de diciembre de 2017

1. Agradecimientos

A nuestras familias, sin su apoyo nada sería posible.

A nuestros amigos y compañeros que nos ayudaron en todo lo que pudieron.

A la universidad por proveernos de todos los elementos necesarios para llevar adelante este trabajo.

A Jorge, Gaby y Nelson por ser uno más en el laboratorio.

2. Índice

2.1. Contenidos

Índice

1. Agradecimientos	1
2. Índice	2
2.1. Contenidos	2
2.2. Lista de Figuras	5
2.3. Lista de Tablas	6
3. Acrónimos y Definiciones	7
3.1. Acrónimos	7
3.2. Definiciones	7
4. Resumen	9
5. Introducción	10
5.1. Antecedentes	10
5.2. Contexto del proyecto	10
6. Objetivos	11
6.1. Finalidad del Proyecto	11
6.2. Planteamiento del Problema a Resolver	11
6.3. Alcance	11
7. Definición de Producto	12
7.1. Requerimientos de Cliente	12
7.1.1. Encuesta	12
7.1.2. Relevamiento de Datos	12
7.1.3. Requerimientos finales para trazabilidad	13
7.1.4. Casa de Calidad	14
7.2. Diagrama Funcional de Interfaces	16
7.3. Especificaciones de Diseño	16
7.3.1. Especificaciones Funcionales	16
7.3.2. Especificaciones de Interfaz	17
7.3.3. Especificaciones de Performance	18
7.3.4. Especificaciones de Implementación	19
7.3.5. Especificaciones de Servicio (RAMS)	20
8. Plan de Validación	22
8.1. Diseño de Bancos de Pruebas	22
8.2. Especificaciones de Pruebas	25
8.3. Diseño y Especificaciones de Simulaciones	29
8.4. Matriz de Trazabilidad de Validación	30
8.5. Plan de Tests	31
9. Análisis de Factibilidad	32
9.1. Factibilidad tecnológica	32
9.1.1. Explicación teórica	32
9.1.2. Aplicación al proyecto	32
9.1.3. Propuesta de alternativas de diseño	34
9.1.3.1. Obtención de la imagen	34
9.1.3.2. Obtención de la posición del espejo	34
9.1.3.3. Microcontrolador	35
9.1.3.4. Procesamiento de imagen	35
9.1.3.5. Correcciones por experimentación	36
9.1.4. DFMEA	37
9.2. Factibilidad de tiempos	39
9.2.1. Planificación (PERT y simulación de Montecarlo)	39

9.2.2.	Simulación de Monte Carlo	40
9.2.3.	Programación (Gantt)	40
9.2.4.	PERT	41
9.3.	Factibilidad económica	42
9.3.1.	Mercado	42
9.3.2.	Costo estimado	42
9.3.3.	Modelos de negocio	43
9.3.3.1.	Comentarios adicionales	44
9.4.	Factibilidad legal y responsabilidad civil	44
9.4.1.	Seguridad eléctrica	45
9.4.2.	Normas aplicables sobre la compatibilidad electromagnéticas	45
9.4.3.	Patentes	46
10.	Ingeniería de detalle	47
10.1.	Hardware	47
10.1.1.	Diagrama de bloques	47
10.1.2.	Descripción detallada de cada bloque	47
10.1.3.	Detalles de selección y cálculo de los elementos circuitales de cada bloque (del final)	48
10.1.4.	Plan de pruebas de cada módulo (del prototipo)	49
10.2.	Software	50
10.2.1.	Diagrama de estados y flujogramas	50
10.2.2.	Análisis de complejidad	51
10.2.2.1.	Reconocimiento del rostro del conductor	53
10.2.2.2.	Motivación	53
10.2.2.3.	Algoritmo	53
10.2.2.3.1.	Rasgos e imagen integral	53
10.2.2.3.2.	Selección de rasgos con Adaboost	55
10.2.2.3.3.	Cascada atencional[1]	56
10.2.2.3.4.	Experimentos y datos	58
10.2.2.3.5.	Pos-Procesamiento	58
10.2.2.3.6.	Fase final	59
10.2.2.4.	Tiempos.	60
10.2.2.5.	Versatilidad.	61
10.2.2.6.	Robustez.	61
10.2.2.7.	Conclusión	61
10.2.2.8.	Identificación de la profundidad	61
10.2.3.	Descripción de subrutinas	62
10.2.3.1.	Toma de la imagen	62
10.2.3.2.	Reconocimiento de rostro	62
10.2.3.3.	Reconocimiento de ojos	63
10.2.3.4.	Cálculo de diferencia de posición respecto una referencia	63
10.2.3.5.	Lazo de control del espejo	63
10.2.4.	Plan de prueba de módulos y de depuración de Software	63
10.2.4.1.	Detector de rostros	63
10.2.4.1.1.	Objetivo	63
10.2.4.1.2.	Pre-requisitos	63
10.2.4.1.3.	Entrada	63
10.2.4.1.4.	Procedimiento	63
10.2.4.1.5.	Conclusión	64
10.2.4.2.	Detector de ojos	64
10.2.4.2.1.	Objetivo	64
10.2.4.2.2.	Pre-requisitos	64
10.2.4.2.3.	Entrada	64
10.2.4.2.4.	Procedimiento	64
10.2.4.2.5.	Conclusión	64
10.2.4.3.	Medición de movimiento	65
10.2.4.3.1.	Objetivo	65
10.2.4.3.2.	Pre-requisitos	65
10.2.4.3.3.	Entrada	65
10.2.4.3.4.	Procedimiento	65
10.2.4.3.5.	Conclusión	65

10.2.4.4.	Control sobre los motores	65
10.2.4.4.1.	Objetivo	65
10.2.4.4.2.	Pre-requisitos	65
10.2.4.4.3.	Entrada	65
10.2.4.4.4.	Procedimiento	65
10.2.4.4.5.	Conclusión	65
11.	Construcción del prototipo	66
11.1.	Definición de los módulos	66
11.1.1.	Alimentación	66
11.1.2.	Sensores para detectar movimiento de espejo	66
11.1.3.	Unidad de procesamiento	66
11.1.4.	Driver motores	66
11.1.5.	Procesamiento óptico	67
11.2.	Diseño de los circuitos impresos	67
11.2.0.1.	Sustrato	67
11.2.0.2.	IPC-A-600	68
11.2.0.3.	IPC-A-601	68
11.2.0.4.	Acabado en superficies	68
11.2.0.5.	Protecciones	68
11.2.0.6.	Conclusión	68
11.3.	Detalles de construcción y precauciones especiales de montaje	68
11.3.1.	IMU	68
11.3.1.1.	Tipo de conexionado	69
11.3.2.	Cámara	69
11.3.3.	Sistema	69
12.	Validación del prototipo	70
12.1.	Resultados	70
12.2.	Evaluación	70
12.3.	Consideraciones finales hacia el producto final	71
12.4.	Estudios de confiabilidad de hardware y software	71
12.4.1.	Software	71
12.4.1.1.	Modelo de predicción	71
12.4.1.2.	Modelo de estimación	72
12.4.2.	Hardware	73
13.	Conclusiones	75
13.1.	Cumplimiento de Objetivos	75
13.2.	Lecciones aprendidas	75
13.3.	Recomendaciones para futuros diseños	75
14.	Anexos Técnicos	77
14.1.	Esquemáticos	77
14.2.	Planos de PCB	79
14.3.	Listado de Partes y Componentes (BOM)	80
14.4.	Códigos de Software	81
14.4.1.	Lectura de IMU	81
14.4.2.	Inicialización	89
14.4.3.	Homing	91
14.4.4.	Movimiento	92
14.4.5.	Toma de imagen	92
14.4.6.	Reconocimiento de centro de ojos	93
14.4.7.	Cálculo de desplazamiento	94
14.4.8.	Calibración	94
14.4.9.	Detección de luminosidad	94
14.5.	Hojas de Datos de Componentes	95
14.5.1.	Microcontrolador	95
14.5.2.	IMU	95
14.5.3.	Driver	95
14.5.4.	Fuente	95

14.6. Bitácora	96
14.6.1. Bitácora de software	97
14.6.1.1. Semana 1:	97
14.6.1.2. Semana 2:	98
14.6.1.3. Semana 3:	98
14.6.1.4. Semana 4:	98
14.6.1.5. Semana 5:	98

2.2. Lista de Figuras

Índice de figuras

1. Preguntas 1 y 2	12
2. Preguntas 3 y 5	13
3. Preguntas 7 y 8	13
4. Casa de calidad	15
5. Diagrama Funcional de Interfaces	16
6. Diagrama de conexionado Banco de prueba 1	24
7. Diagrama de conexionado Banco de prueba 2	24
8. Diagrama de conexionado Banco de prueba 3 y 4	24
9. Diagrama de dependencias de Validación	31
10. Diagrama de reflexión	32
11. Diagrama óptico	33
12. DFMEA de diseño modular	38
13. Histograma de duración en días	40
14. Diagrama de Gantt	41
15. Diagrama PERT	41
16. Participación en 2014	42
17. Tabla de valores de referencia	44
18. Flujo de fondos	44
19. Esquema de espejos retrovisores exteriores	45
20. Diagrama en bloques	47
21. Lectura I2C	49
22. Distribución de <i>pins</i> de L293D	49
23. Lectura IMU	50
24. Diagrama de estados de software	51
25. Plano general de software	52
26. Método Viola-Jones	52
27. AdaptiveBoosting	53
28. Rasgos pseudo-Haar. El fondo de la plantilla (b) es gris para resaltar el patrón. Solo los pixels blancos y negros son utilizados.	54
29. Cinco patrones pseudo-Haar. El tamaño y la posición de los patrones puede variar, siempre manteniendo el mismo tamaño en cuanto a sus rectángulos negros y blancos.	54
30. Ejemplos supervisados. (a) Ejemplos positivos (b) Ejemplos negativos	55
31. Diferente desempeño por cantidad de capas	59
32. Detección de ojos	60
33. Resultado final	60
34. Histórico de errores	73
35. Esquemático IMU	77
36. Esquemático Driver	77
37. Esquemático Fuente Switching	78
38. Esquemático uSD	78
39. PCB IMU	79
40. PCB Placa controladora	79

2.3. Lista de Tablas

Índice de cuadros

1.	Acrónimos y definiciones	7
2.	Requerimientos	14
3.	Leyenda de uso en especificaciones	16
4.	Especificaciones Funcionales	17
5.	Especificaciones de Interfaz VIN	17
6.	Especificaciones de Interfaz VOUT	17
7.	Especificaciones de Interfaz MEC	18
8.	Especificaciones de Performance	18
9.	Especificaciones de Operación	19
10.	Especificaciones de Almacenamiento y Transporte	19
11.	Especificaciones de Compatibilidad Electromagnética	20
12.	Especificaciones Dimensionales y de Peso	20
13.	Especificaciones de costos	20
14.	Especificaciones de Confiabilidad	20
15.	Especificaciones de Mantenibilidad	21
16.	Especificaciones de Seguridad	21
17.	Bancos de prueba	23
18.	Medidas de habitáculos a simular	24
19.	Tests de Interfaz VIN-VOUT	26
20.	Tests de Mantenibilidad	26
21.	Tests de funcionalidad	27
22.	Tests de operación	28
23.	Especificaciones de simulaciones	29
24.	Matriz de Trazabilidad para Validación	31
25.	Actividades	39
26.	Costos prototipo	43
27.	Prueba de resolución de cámara	48
28.	Rostros: LBP vs Haar	63
29.	Ojos: LBP vs Haar	64
30.	Tabla de consumos	66
31.	Cotización	68
32.	Tests de Interfaz VIN-VOUT	70
33.	Tests de Mantenibilidad	70
34.	Tests de funcionalidad	70
35.	Tabla de errores	72
36.	Confiabilidad Capacitores	74
37.	Confiabilidad resistores	74
38.	Confiabilidad otros componentes	74
39.	BOM IMU	80
40.	BOM placa controladora	80

3. Acrónimos y Definiciones

3.1. Acrónimos

dc	Corriente continua
mA	miliampere
uP	Microprocesador
uC	Microcontrolador

Cuadro 1: Acrónimos y definiciones

3.2. Definiciones

DSP Un procesador digital de señales o DSP (sigla en inglés de digital signal processor) es un sistema basado en un procesador o microprocesador que posee un conjunto de instrucciones, un hardware y un software optimizados para aplicaciones que requieran operaciones numéricas a muy alta velocidad. Debido a esto es especialmente útil para el procesamiento y representación de señales analógicas en tiempo real: en un sistema que trabaje de esta forma (tiempo real) se reciben muestras (samples en inglés), normalmente provenientes de un conversor analógico/digital (ADC).

Driver Circuito electrónico o componente utilizado para controlar otro circuito o componente, como transistores de alta potencia, displays de cristal líquido, entre otros. Generalmente son usados para regular el flujo de corriente hacia un circuito o controlar otros factores. Un claro ejemplo es el de los amplificadores para altoparlantes o un circuito de tensión constante que mantiene los componentes anexados operando pese a un gran rango de tensiones de entrada, como es el caso de este proyecto.

FPGA Una FPGA (del inglés Field Programmable Gate Array) es un dispositivo programable que contiene bloques de lógica cuya interconexión y funcionalidad puede ser configurada ‘in situ’ mediante un lenguaje de descripción especializado. La lógica programable puede reproducir desde funciones tan sencillas como las llevadas a cabo por una puerta lógica o un sistema combinacional hasta complejos sistemas en un chip.

Microcontrolador Un microcontrolador es un circuito integrado que en su interior contiene una unidad central de procesamiento (CPU), unidades de memoria (RAM y ROM), puertos de entrada y salida y periféricos. Estas partes están interconectadas dentro del microcontrolador, y en conjunto forman lo que se le conoce como microcomputadora. Se puede decir con toda propiedad que un microcontrolador es una microcomputadora completa encapsulada en un circuito integrado.

Microprocesador El microprocesador es en principio un componente electrónico, más específicamente decimos que es un circuito integrado, es decir, es una pieza electrónica que contiene miles o millones de transistores a la vez.

Motor en contacto Cuando se coloca la llave de encendido en la primera posición. En este modo, funciona toda la parte eléctrica del vehículo sin que el motor esté en marcha.

Puente H Un puente H es un circuito electrónico que permite a un motor eléctrico DC girar en ambos sentidos, avance y retroceso. Son ampliamente usados en robótica y como convertidores de potencia. Los puentes H están disponibles como circuitos integrados, pero también pueden construirse a partir de componentes discretos.

Raspberry Pi Serie de pequeñas computadoras de una sola placa que promueven la enseñanza de ciencias básicas de la computación. Varias generaciones han sido periódicamente publicadas desde su primer lanzamiento en 2012. La Raspberry Pi (RPi) cuenta con su propio procesador, memoria, unidad de procesamiento gráfico y la capacidad de conectarle diversos periféricos.

VHDL VHDL es un lenguaje definido por el IEEE (Institute of Electrical and Electronics Engineers) (ANSI/IEEE 1076-1993) usado por ingenieros y científicos para describir circuitos digitales o modelar fenómenos científicos respectivamente. VHDL es el acrónimo que representa la combinación de VHSIC y HDL, donde VHSIC es el acrónimo de Very High Speed Integrated Circuit y HDL es a su vez el acrónimo de Hardware Description Language. Para el modelado físico existe la derivación del lenguaje VHDL-AMS. Originalmente, el lenguaje VHDL fue desarrollado por el departamento de defensa de los Estados Unidos a inicios de los años 80's basado en ADA, con el fin de realizar simulación de circuitos eléctricos digitales;

sin embargo, posteriormente se desarrollaron las herramientas de síntesis e implementación en hardware a partir de los archivos .VHD. Aunque puede ser usado de forma general para describir cualquier circuito digital se usa principalmente para programar PLD (Programmable Logic Device - Dispositivo Lógico Programable), FPGA (Field Programmable Gate Array), ASIC y similares.

IMU Acrónimo de Unidad de Medición Inercial, por sus siglas en inglés, es un dispositivo electrónico que mide y reporta las fuerzas, momento angular y, a veces, campo magnético de un cuerpo usando una combinación de acelerómetros, giróscopos y, a veces, magnetómetros. Detectando la tasa de aceleración usando uno o más acelerómetros; los cambios en atributos rotacionales tales como cabeceo, alabeo y guiñada usando uno o más giróscopos y cambios en la orientación magnética respecto a una referencia utilizando un magnetómetro, la IMU ofrece hasta 9 grados de control (3 por subsistema).

GPIO Se trata de pines genéricos integrados a un circuito cuyo comportamiento (entrada o salida) es controlado por el usuario en tiempo de ejecución. El limitante que poseen es que su valor está acotado por la alimentación del circuito y la corriente que pueden suministrar es de pocos mA (16mA para el caso en cuestión).

Drift IMU Debido a que el sistema de guía de la IMU se encuentra continuamente integrado aceleración con respecto al tiempo para calcular velocidad y posición, cualquier error de medición, pese a ser bajo, es acumulado sobre el tiempo. A ello se lo llama drift.

Filtro de Kalman Algoritmo que utiliza una serie de mediciones observadas durante un tiempo, las cuales contienen ruido estadístico y otras imprecisiones, y produce una estimación de variables desconocidas que tienden a ser más precisas que aquellas basadas en una sola medición. El algoritmo funciona en un proceso de dos pasos: En el primer paso (de predicción) el filtro de Kalman produce una estimación de las variables de estado actuales junto con sus incertezas. Una vez observada la siguiente medición (necesariamente corrupta con alguna cantidad de error, incluido ruido aleatorio), estas estimaciones son actualizadas usando un promedio ponderado, con más peso dado a las estimaciones con mayor certeza. El algoritmo es recursivo. Puede ejecutarse en tiempo real, utilizando sólo las mediciones de entrada actuales y el estado previamente calculado y su matriz de incertidumbre; No se requiere ninguna información adicional del pasado.

BGA Ball Grid Array. Es un tipo de empaque utilizado para circuitos integrados. Se utilizan para montar permanentemente dispositivos como microprocesadores. Un BGA puede proporcionar más pines de conexión que se pueden poner en un empaque dual line o plano. Se puede utilizar toda la superficie inferior del dispositivo, en lugar del perímetro. Suelen tener capas de pines lo que dificulta su conexión con otras partes.

VCC Acrónimo para tensión de alimentación. Simboliza el voltaje que se le da a un circuito para su funcionamiento.

GND Acrónimo para tierra. Simboliza la tierra de un circuito eléctrico.

GPU Unidad de procesamiento gráfico o GPU (Graphics Processor Unit) es un coprocesador dedicado al procesamiento de gráficos, para aligerar la carga de trabajo del procesador central. De esta forma, mientras gran parte de lo relacionado con los gráficos se procesa en la GPU, la unidad central de procesamiento (CPU) puede dedicarse a otro tipo de cálculos.

DUT Device Under Test. Hace referencia al dispositivo que se encuentra bajo observación y análisis.

ADABOOST Adaptive Boosting. Se trata de un meta-algoritmo de *machine learning*, que puede ser usado en conjunto con otros algoritmos para incrementar el desempeño de estos últimos. Las salidas de los algoritmos de aprendizaje son combinadas en una suma ponderada que representa la salida final del sistema.

4. Resumen

Con el correr del tiempo, en la industria automotriz se ha ido propagando una tendencia hacia la mejora del rendimiento tanto del conductor como la del vehículo dada por el creciente avance e incorporación de nuevas tecnologías.

Esta ola propicia, también, el desarrollo de nuevas formas para garantizar el máximo confort al conducir y acrecentar la seguridad de ello.

La motivación de este trabajo nace de la incomodidad de tener que modificar reiteradas veces, de forma manual, la posición de los espejos laterales debido a que otro usuario los ha acomodado a su gusto y poder garantizar de la forma más confiable posible la posición más segura acorde a estudios y normas preexistentes.

El proyecto se aboca a poder acomodar los espejos laterales del automóvil, de forma automática, con un método no invasivo. Se trata de detectar los ojos del conductor y la profundidad de este, por medio de una cámara de pequeño tamaño ubicada en el parante del vehículo, o tomar control de la posición de ellos a través de una IMU colocada en la parte trasera del espejos dentro de su receptáculo.

En el plano económico, se ha establecido el sistema por un costo menor a \$1500 (valor a 2016). Tecnológicamente, se ha dotado de un lazo de control a la mecánica de los espejos y se ha incorporado una computadora de abordo en caso de no poseerla, bajo el precio mencionado.

Esto último dicho abre el abanico de negocios a dos posibilidades: Licenciar el software y venderlo a las automotrices, o vender el sistema completo a cualquier particular que cumpla con los requerimientos de espejos eléctricos.

Se encontró una limitación insorteable, por tratarse de un prototipo y no de un producto final íntegro, pero que puede ser solventada en futuras versiones, la cual es la dependencia de la localización de los ojos con la cantidad de luz disponible. Esto quiere decir que, por debajo de un determinado nivel de luminosidad la eficacia del sistema disminuye considerablemente.

5. Introducción

5.1. Antecedentes

A comienzos de la historia del vehículo, tener un espejo lateral se consideraba un lujo y el mismo se ofrecía como una opción a los compradores. Inicialmente, solo se utilizaba un espejo central, ya que las calles eran en su mayoría de un carril, haciendo innecesario un espejo lateral para ver a los costados. Recién hacia 1940 se comenzó a ofrecer un solo espejo lateral, del lado del acompañante, como opcional. Pasado el tiempo, y con la evolución de las medidas de seguridad vehicular, la presencia de tres espejos se presenta como equipamiento estándar. Los mecanismos para su posicionamiento también sufrieron una evolución. Las primeras versiones se acomodaban manualmente, actuando directamente sobre los mismos. Luego se llevó el control al interior del habitáculo accionado una pequeña palanca para cada espejo (una en cada puerta), y hoy en día, se acomodan ambos espejos eléctricamente desde un comando ubicado del lado del conductor. Estas dos últimas modalidades conviven en la actualidad, aunque el método manual solo se encuentra en modelos muy básicos. Ya en modelos más caros de tope de gama de un número de marcas muy reducido, existe la posibilidad de guardar en memoria la posición del asiento y la posición de los espejos, previo posicionamiento por parte del usuario.

5.2. Contexto del proyecto

La idea del proyecto surge de querer combinar las técnicas y tecnologías aprendidas durante estos años con un área que nos apasiona como es el mundo automotriz. Al mismo tiempo, nos interesa poder lograr que más gente pueda acceder a funciones *premium* sin necesidad de pagar por todo el paquete de funciones exclusivas.

Hoy en día, los vehículos de alta gama vienen en su mayoría con un sistema integral incorporado a la consola central del vehículo. Este sistema controla todas funciones de confort del vehículo. Estas funciones incluyen el aire acondicionado, la radio, el GPS y hasta la amortiguación. Muchos de estos sistemas tienen la posibilidad de guardar diversos perfiles para las distintas personas que utilizan el vehículo. No solo guardan las preferencias de la radio y el aire acondicionado, sino también incluyen la posibilidad de guardar la posición del asiento, en aquellos casos cuyo acomodamiento se hace eléctricamente, y la posición de los espejos, la cual, naturalmente, acompaña la postura del conductor.

La meta es intentar traer ésta última función a vehículos de variada gama, que no necesariamente incluyan grandes prestaciones. No es siquiera necesario poseer asientos eléctricos (para poner en contexto). El único requerimiento es que el automóvil posea espejos eléctricos. En última instancia, esto no es un impedimento dado que se pueden cambiar, pero no contribuiría el objetivo de hacerlo accesible.

6. Objetivos

6.1. Finalidad del Proyecto

A la hora de comprar un vehículo, uno no solo tiene en cuenta el precio del mismo sino también que las prestaciones se correspondan con este.

Muchas veces se observa que la escala de modelos desde la más básica hasta el tope de gama suele estar desbalanceada. Es decir, un aumento excesivo de precio no necesariamente involucra una gran incorporación de sistemas que faciliten y hagan más placentero el conducir.

Por lo general, para acceder a prestaciones útiles o que tienen valor para el usuario hay que pagar un precio muy elevado y no necesariamente acarrea un gran número de otros servicios (como fue el caso del levanta-vidrios eléctrico antiguamente). La incorporación de estos accesorios o funciones normalmente vienen acompañados de un motor más potente, asientos de un material *premium* y a veces hasta otro tipo de transmisión.

El objeto de este trabajo es dar el punto de inicio para la modularización en la incorporación de aditamentos y/o accesorios al automóvil. Puntualmente, ofrecer un equipamiento considerado *premium*, a un precio accesible y en forma de bloque. Permitiendo así la continua incorporación de nuevos servicios de forma independiente.

6.2. Planteamiento del Problema a Resolver

La problemática planteada como finalidad del proyecto no puede ser encarada de forma genérica ya que las partes que lo componen no necesariamente están conectadas o funcionan naturalmente de forma encadenada.

Para poder resolver esta situación se separó el proyecto en las siguientes partes de forma tal que puedan ser solventadas de forma modular y escalable, y la incorporación de cada submódulo al total sea lo más desvinculada de las operaciones del resto.

Reconocimiento de posición del conductor y versatilidad: todo el sistema se basa fundamentalmente en la correcta ubicación del conductor. Se debe evaluar cuál es el punto óptimo para lograr la mejor posición posible. Esto puede ser los ojos o el ceño. Dependiendo de qué se elija para lograr este objetivo, se definirá la elección de sensores a utilizar. Si éste paso del proceso falla, el resultado puede no tener sentido alguno anulando el objetivo por completo. Es importante la versatilidad en estos casos, dejando al conductor realizar un ajuste fino llegado el caso.

Adaptabilidad: otro aspecto importante es qué mecanismo se implementará para que el sistema se adapte a las diversas formas de autos que existen en la actualidad. Es necesario conocer el largo de vehículo como también su altura. Es posible que estas medidas sean ingresadas por el fabricante cuando incorpora el sistema de fábrica. También existe la posibilidad de utilizar sensores de manera de crear un *stop* para saber cuando detener el movimiento del espejo.

Posicionamiento del espejo: la tecnología utilizada por cada fabricante para sus espejos definirá que tipo de mecanismo se utilizará para determinar cómo posicionar el espejo. El software utilizado para determinar el movimiento del espejo debe ser independiente de la tecnología. La diferencia radica básicamente en qué sensor se utilizará.

6.3. Alcance

A lo largo de este informe se demuestra el desarrollo de un sistema de detección de ojos de conductor de automóvil, no invasivo, y de posicionamiento automático de espejos laterales en base a ellos.

Durante el proyecto se desarrollará un prototipo funcional de carácter no comercial, que cumplirá con los requerimientos de funcionalidad básicos, sin satisfacer aquellos requerimientos más exhaustivos (como compatibilidad electromagnética o análisis vibratorio, entre otros), pero planteando la necesidad de ellos. A su vez, el software desarrollado contará con suficiente versatilidad como para ser transferido a cualquier sistema operativo de una computadora de a bordo capaz de compilar la librería OpenCV, ejecutar programas de Python3, posea puerto I²C y 7 GPIOs.

Este trabajo no incluye la realización del producto final, aunque serán desarrollados los criterios para uno, acompañado de esquemas de conexionado y de submódulos que permitan la producción final en masa.

El alcance de este trabajo llega hasta una instancia anterior a incorporar un procesador fuera de un kit de desarrollo embebido. Es decir, se exponen todas las partes que conforman al sistema, incluidos desarrollos propios, requerimientos y especificaciones de todo subsistema, excluyendo el desarrollo de una placa controladora central.

7. Definición de Producto

7.1. Requerimientos de Cliente

7.1.1. Encuesta

Se desarrolló una encuesta para poder realizar un producto que satisfaga las necesidades de los usuarios. Se optó por realizar una breve descripción y pocas preguntas para no perder la atención de los encuestados.

1. **¿Comparte su vehículo con otra/s persona/s?**
 - a) Sí
 - b) No
2. **¿Le resulta importante que no se modifiquen sus preferencias?**
 - a) Sí
 - b) No
3. **¿Modifica sus preferencias a lo largo del mismo viaje?**
 - a) Si
 - b) No
4. **¿Qué acción/es de su auto le gustaría que fueran automáticas?**
5. **¿Le gustaría que existiera una función que posicione los espejos automáticamente?**
 - a) Si, me gustaría
 - b) No, me gusta acomodarlos a mi gusto
 - c) Me es indiferente
6. **¿Cuánto estaría dispuesto a pagar para agregar esta funcionalidad a su vehículo?**
7. **¿Consideraría esta opción a la hora de comprar un vehículo?**
 - a) Sí
 - b) No
 - c) Puede ser
8. **¿Le gustaría que el espejo adecuado se acomode para estacionar?**
 - a) Si, me gustaría
 - b) No, me gusta ver para atrás
 - c) Me es indistinto

7.1.2. Relevamiento de Datos

Sobre la base de 140 encuestados, se obtuvieron las siguientes respuestas:

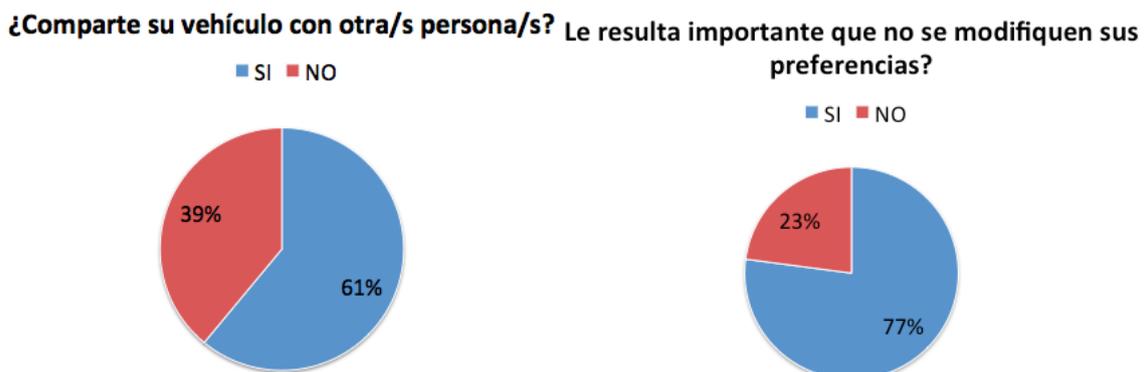
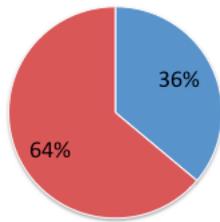


Figura 1: Preguntas 1 y 2

¿Modifica sus preferencias a lo largo del mismo viaje?

■ SI ■ NO



¿Le gustaría que existiera una función que posicione los espejos automáticamente?

■ Si, me gustaría ■ No, me gusta acomodarlos a mi gusto ■ Me es indiferente

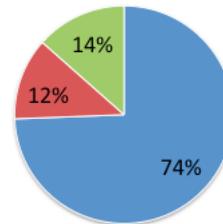
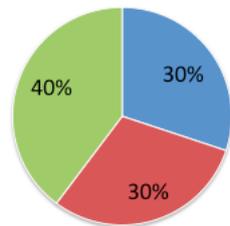


Figura 2: Preguntas 3 y 5

¿Consideraría esta opción a la hora de comprar un vehículo?

■ Si ■ No ■ Puede ser



¿Le gustaría que el espejo adecuado se acomode para estacionar?

■ Si, me gustaría ■ No, me gusta ver para atrás ■ Me es indistinto

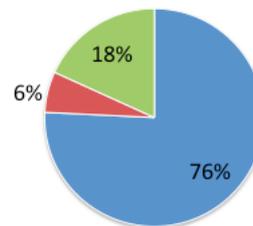


Figura 3: Preguntas 7 y 8

La Pregunta 4 arrojó resultados parecidos al equipamiento que ofrecen autos de muy alta gama que controlan todo desde la computadora central (nuestra competencia más directa). El objetivo de nuestro producto es ofrecer a autos que cuenten con espejos eléctricos, uno de los tantos equipamientos de lujo ofrecidos en autos de alta gama, a bajo costo. En su mayoría, los encuestados están dispuestos a pagar entre \$1000 y \$1500 (valor a 2016) para agregar esta función sin hacer distinción en los ingresos del usuario. Se destaca que la Pregunta 8, que apunta a una función considerada adicional, como es acomodar el espejo para estacionar, resultó muy bienvenida. Esto presenta un giro en la concepción del producto, ya que esta opción no puede considerarse como adicional, si no que es obligatoria para atraer al cliente.

7.1.3. Requerimientos finales para trazabilidad

Después de analizar los resultados de la encuesta sumado a la comprensión de comentarios y sugerencias realizadas por fuera de la encuesta, se pudieron determinar los siguientes requerimientos por parte de potenciales usuarios del sistema.

ID	Descripción	Origen
REQ-01	El producto deberá cumplir con la norma IRAM ISO 11451 de seguridad eléctrica	Ley Argentina
REQ-02	El producto deberá cumplir con la norma IRAM ISO 4 de seguridad eléctrica	Ley Argentina
REQ-03	El producto deberá poder operarse en todas las condiciones normales de temperatura, presión y humedad que se pueden encontrar en el territorio argentino.	Tácito
REQ-04	El producto deberá poder funcionar correctamente pese a una desconfiguración manual externa de la posición del espejo.	Tácito
REQ-05	El producto debe ser un sistema robusto que no requiera de calibración continua	Tácito
REQ-06	El producto debe ser sencillo de usar	Tácito
REQ-07	El producto debe poseer un tiempo de respuesta rápido	Tácito
REQ-08	El producto no debe necesitar una fuente de alimentación adicional a la batería del auto.	Tácito
REQ-09	El producto debe poder funcionar en cualquier momento que el auto se encuentre en marcha o en contacto	Tácito
REQ-10	El costo del producto deberá ser menor a U\$S100 (valor a 2016)	Encuesta
REQ-11	El producto debe ser posible de incorporar a cualquier vehículo terrestre de 4 ruedas que posea espejos eléctricos.	Encuesta
REQ-12	Al momento de estacionar, el usuario debe poder seleccionar si quiere que los espejos se acomoden automáticamente para visualizar el suelo y la rueda lateral de algún lado.	Encuesta
REQ-13	El producto debe ser capaz de funcionar sin importar la altura del conductor o la distancia del mismo al volante.	Tácito
REQ-14	El producto no funcionará de forma automática al ser energizados. Su accionar comenzará cuando el usuario lo requiera y de alguna forma de la orden.	Tácito
REQ-15	El equipo deberá poder funcionar las 24 horas del día siempre que el vehículo se encuentre en contacto.	Tácito
REQ-16	De ser necesario un aditamento que cierre el lazo de control, debe poseer dimensiones tal que pueda ser incorporado al receptáculo del espejo lateral.	Tácito
REQ-17	La vida útil del equipo debe ser equiparable a la vida útil de un automóvil.	Tácito
REQ-18	El tiempo medio entre fallas debe ser mayor al tiempo medio entre <i>service</i> del vehículo.	Tácito
REQ-19	El producto debe poder funcionar para el promedio de altura de la población.	Tácito

Cuadro 2: Requerimientos

7.1.4. Casa de Calidad

Para la construcción de la Casa de Calidad se tomaron en cuenta los requerimientos de los clientes obtenidos a partir de la encuesta. Una vez hecho esto, se evaluaron los requerimientos funcionales. Es decir, aquellos que garantizan la funcionalidad del sistema. Se construye una matriz de correlación entre dichas necesidades a partir de la cual se obtiene una medida de qué tanto impacta un requerimiento funcional sobre los del cliente.

Los aspectos más importantes para el usuario son la rapidez de respuesta, que sea intuitivo y fácil de usar, y también su precio a la hora de adquirirlo y agregarlo en vehículos viejos. En cuanto al aspecto que mayor trabajo requiere durante el desarrollo del proyecto, se puede ver en la Casa de calidad que se resume en la optimización del sistema de reconocimiento.

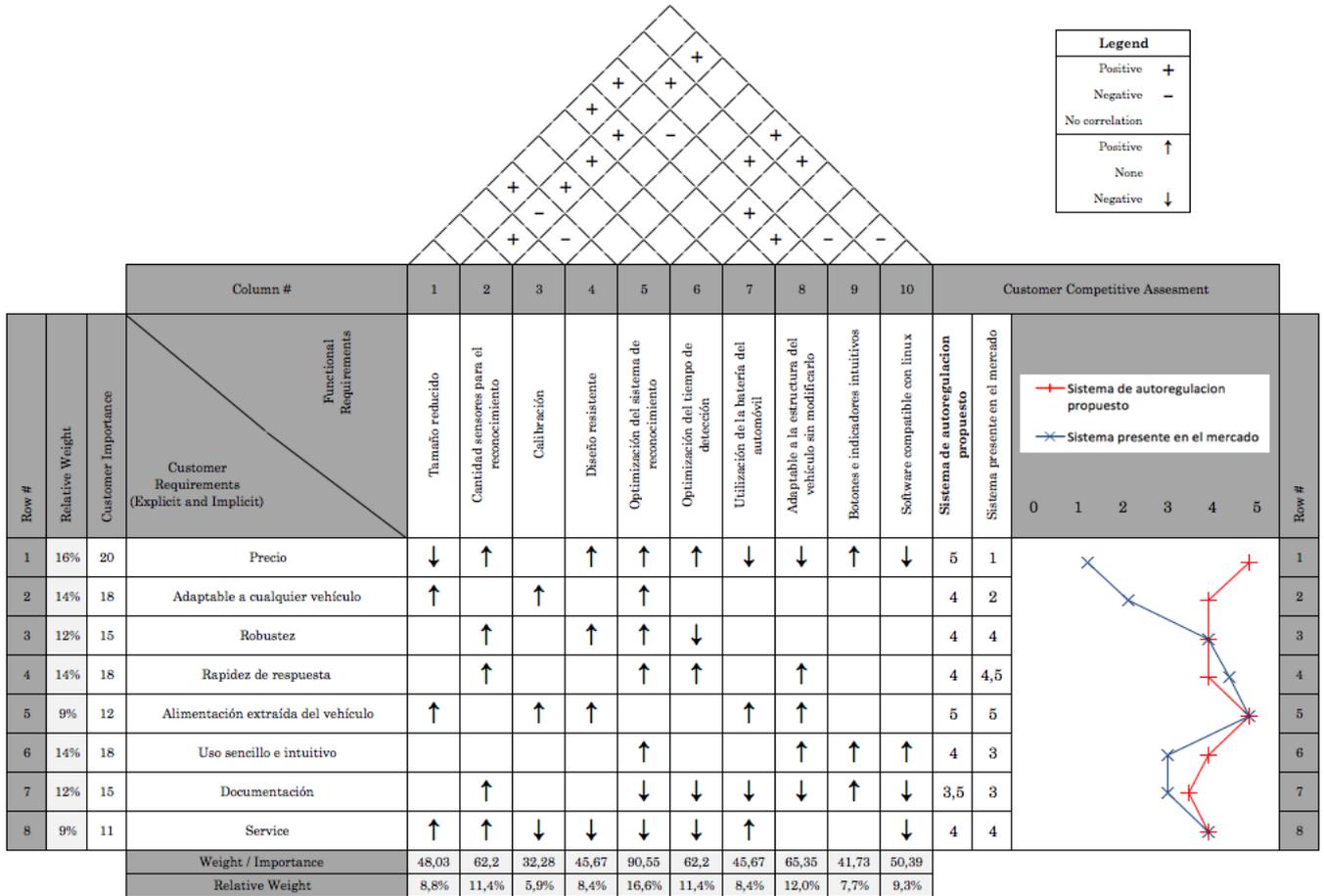


Figura 4: Casa de calidad

Cabe aclarar que actualmente no existe un sistema como el propuesto en el mercado. Dicho esto, la competencia propuesta es simplemente los sistemas integrados en los vehículos actuales, los cuales guardan la posición del asiento, como así también la de los espejos para diferentes perfiles de conductor. Por lo general, estas prestaciones están presentes en los tope de gama de cada modelo, si es que existe tal opción. Por este motivo, algunos aspectos no son comparables directamente, como puede ser la rapidez o versatilidad. Para comparar el precio, se consideró la diferencia promedio que se debe abonar para poder acceder a un tope de gama con estas prestaciones.

7.2. Diagrama Funcional de Interfaces

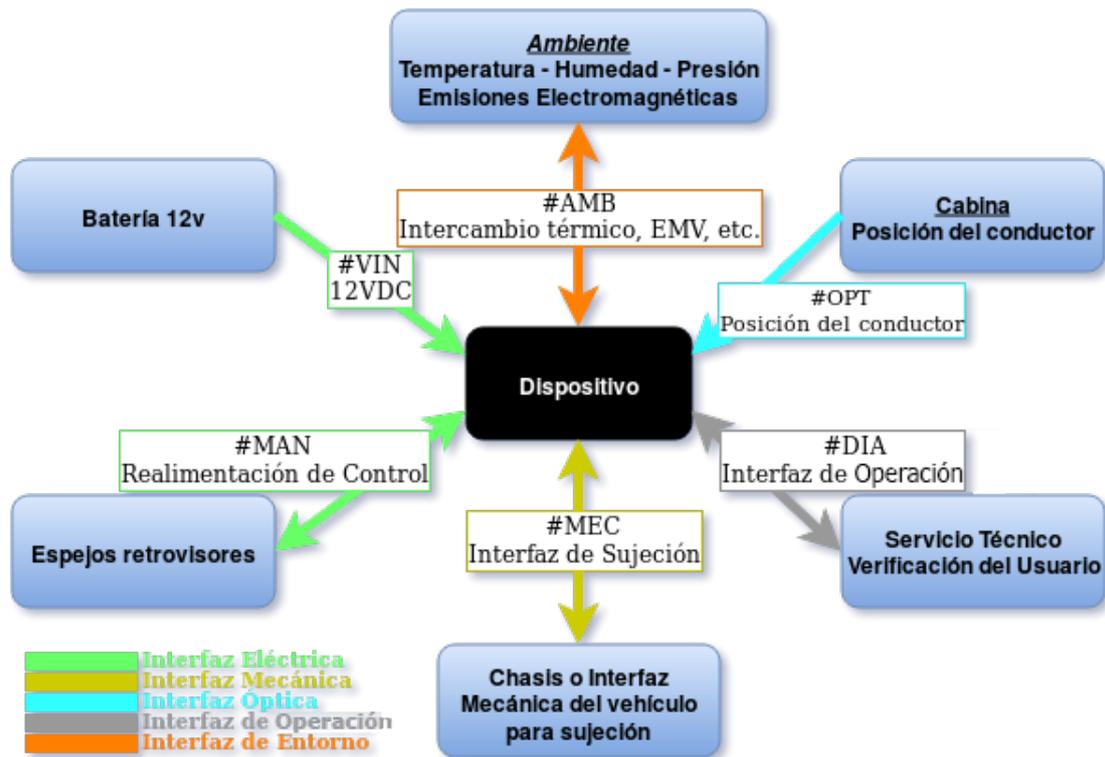


Figura 5: Diagrama Funcional de Interfaces

7.3. Especificaciones de Diseño

7.3.1. Especificaciones Funcionales

Leyenda para Especificaciones	
Aplicabilidad	Validación
DP: Dispositivo, Prototipo	I: Inspección Visual
IP: Instalación, Prototipo	D: Documentación de Diseño
DF: Dispositivo, Producto Final	S: Simulación
IF: Instalación, Producto Final	T: Test

Cuadro 3: Leyenda de uso en especificaciones

ID	Descripción	Origen	Aplicabilidad Validación
FUN-POW-01	El producto deberá tomar una señal de tensión continua proveniente de una batería de 12 Volts en su entrada, y garantizar su funcionamiento.	REQ-08	DP, IP, DF, IF D, T
FUN-POW-02	El producto deberá poder operar cuando esté sujeto a un vehículo en reposo o en movimiento, siempre y cuando esté en marcha o en contacto.	REQ-09	DP, IP, DF, IF D, T
FUN-POW-03	El producto deberá poder operar incluso con pequeños obstáculos que dificulten encontrar la posición de los ojos del conductor (como anteojos).	REQ-05 REQ-06	DP, DF I, D, T
FUN-POW-04	El producto deberá poder regular correctamente la posición de los espejos independiente de la posición en la que se encuentren previamente a inicializar el sistema.	REQ-03 REQ-04 REQ-05	DP, DF I, D, T
FUN-POW-05	El producto deberá poder regular correctamente la posición de los espejos al momento de estacionar para poder visualizar el contacto entre la rueda trasera y el asfalto, y volver a colocarlos en la posición de confort luego.	REQ-13	DP, DF D, T
FUN-POW-06	El equipo deberá poder operar en las condiciones lumínicas presentes a lo largo del día: Desde $8192 \frac{cd}{m^2}$ para pleno sol, hasta $0.03125 \frac{cd}{m^2}$ para noche con luna	REQ-15	DP, DF D, T
FUN-POW-07	El equipo deberá poder operar para el usuario promedio de la región, cuya altura promedio es de 1.74m para hombres y 1.59m para la mujer[16]	REQ-19	DP, IP, DF, IF D, T

Cuadro 4: Especificaciones Funcionales

7.3.2. Especificaciones de Interfaz

ID	Descripción	Origen	Aplicabilidad Validación
INT-VIN-01	El equipo debe poder operar con una tensión de entrada entre 9.6 VDC y 14.7 VDC, siendo estos los consumos topes de las baterías de los automóviles en operación normal (arranque y carga).	REQ-08	DP, IP, DF, IF D, T
INT-VIN-02	El equipo debe poder operar con una corriente de entrada nominal continua de 0.52A, siendo esta la corriente de descarga estándar para varios modelos de baterías de automóviles[14]	REQ-08	DP, IP, DF, IF D, T

Cuadro 5: Especificaciones de Interfaz VIN

ID	Descripción	Origen	Aplicabilidad Validación
INT-VOUT-01	El equipo debe entregar en el conector de salida una tensión continua de entre 9.6 y 14.7 Volts para los motores de los espejos.	REQ-08	DP, DF D, T
INT-VOUT-02	El equipo debe entregar en el conector de salida una corriente continua de entre 30 y 50 mili-amperes para los motores de los espejos (medida).	REQ-08	DP, DF D, T

Cuadro 6: Especificaciones de Interfaz VOUT

ID	Descripción	Origen	Aplicabilidad Validación		
INT-MEC-01	Siendo que se trata de poder controlar la posición de los espejos, el equipo deberá poder ser incorporado a los espejos laterales del automóvil sin reemplazar el encapsulado de estos.	REQ-06 REQ-11 REQ-16	DP, IP, DF, IF D		
INT-MEC-02	El equipo deberá poder ser incorporado al automóvil sin dificultar la visibilidad del conductor.	REQ-06 REQ-11	DP, IP, DF, IF I, D		
INT-MEC-03	El equipo deberá tolerar vibraciones mecánicas del siguiente modo:	REQ-03 REQ-05 REQ-11	DF D, T		
				Densidad espectral de aceleración	Rango de frecuencias
	Motor			$0.4g^2/Hz$	50-500 Hz ($25g_{rms}$)
	Caja de transmisión			$0.15g^2/Hz$	50-500 Hz ($15g_{rms}$)
	Chasis			$0.1g^2/Hz$	50-500 Hz ($7g_{rms}$)
	Tambor de freno			$0.45g^2/Hz$	50-500 Hz ($15g_{rms}$)
	Cabina	$0.02g^2/Hz$	50-500 Hz ($3.3g_{rms}$)		

Cuadro 7: Especificaciones de Interfaz MEC

7.3.3. Especificaciones de Performance

ID	Descripción	Origen	Aplicabilidad Validación
PER-POW-01	Considerando que una persona suele tardar 10 segundos en posicionar el espejo de forma manual, el tiempo desde el inicio de la operación hasta mover el espejo en la mejor posición debe ser no mayor 10 segundos.	REQ-07	DP, DF D, S, T
PER-POW-02	El equipo no deberá consumir más de 6W de potencia continua, basado en la corriente típica de descarga de las baterías de automovil[14].	REQ-08 REQ-09	DP, IP, DF, IF D, T

Cuadro 8: Especificaciones de Performance

7.3.4. Especificaciones de Implementación

ID	Descripción	Origen	Aplicabilidad Validación
IMP-OPE-01	El dispositivo deberá poder operar normalmente cuando la temperatura ambiente sea $-10^{\circ}\text{C} < \text{TAMB} < 70^{\circ}\text{C}$ Si bien la temperatura más baja observada en el territorio argentino es de -40°C , se acepta que son circunstancias excepcionales y que a partir de -10°C las baterías comunes comienzan a presentar problemas para arrancar un vehículo. Si bien la temperatura máxima observada es de casi 50°C , la concentración de calor en ambientes cerrados puede hacer que sea superior a este valor en hasta 20°C	REQ-03	DF, IF D, T
IMP-OPE-02	El dispositivo deberá poder operar normalmente cuando la humedad sea: $0\% < \text{RH} < 100\%$	REQ-03	DF, IF D, T
IMP-OPE-03	El dispositivo deberá poder operar normalmente cuando la presión atmosférica sea: $60\text{ kPa} < \text{PATM} < 110\text{ kPa}$ Esto equivale a 4000m de altura para el mínimo de presión, y un máximo ligeramente superior al máximo observado a la altura del mar.	REQ-03	DF, IF D, T
IMP-OPE-04	El dispositivo debe poder operar las 24 horas del día	REQ-09	DP, IP, DF, IF T

Cuadro 9: Especificaciones de Operación

ID	Descripción	Origen	Aplicabilidad Validación
IMP-AYT-01	El dispositivo no deberá sufrir daños cuando, estando desenergizado, la temperatura ambiente sea $-30^{\circ}\text{C} < \text{TAMB} < 80^{\circ}\text{C}$	REQ-03	DF, IF D, T
IMP-AYT-02	El dispositivo no deberá sufrir daños cuando, estando desenergizado, la humedad sea $0\% < \text{RH} < 100\%$	REQ-03	DF, IF D, T
IMP-AYT-03	El dispositivo no deberá sufrir daños cuando, estando desenergizado, la presión atmosférica sea $60\text{ kPa} < \text{PATM} < 110\text{ kPa}$	REQ-03	DF, IF D, T

Cuadro 10: Especificaciones de Almacenamiento y Transporte

ID	Descripción	Origen	Aplicabilidad Validación
IMP-EMC-01	El dispositivo deberá soportar un campo eléctrico de: 50V/m de 10kHz a 2Mhz 100V/m de 2 MHz a 1GHz Campos que no influyen en la seguridad vial 25V/m de 10kHz a 2Mhz 50V/m de 2 MHz a 1GHz El campo electromagnético debe estar modulado en amplitud con un tono de 1kHz y un índice de 100 %	REQ-02	DF, IF D, T
IMP-EMC-02	La emisión eléctrica del dispositivo no debe superar -15dBuV (0.2uV) sobre 50 ohms) en un rango de frecuencia de 20 mHz a 1 GHz. 0dBuV en un rango de frecuencia de 500 kHz a 20 Mhz.	REQ-02	DF, IF D, T

Cuadro 11: Especificaciones de Compatibilidad Electromagnética

ID	Descripción	Origen	Aplicabilidad Validación
IMP-DIM-01	De ser necesario un aditamento que cierre el lazo de control, la colocación del mismo no debe interferir con el movimiento de los motores, por lo que para cumplir con ello y caber detrás del espejo sus dimensiones deben ser: Largo < 10cm Ancho < 10cm Alto < 3cm Estas medidas fueron tomadas para un espejo de un Peugeot 302	REQ-16	DP, IP, DF, IF I, D

Cuadro 12: Especificaciones Dimensionales y de Peso

ID	Descripción	Origen	Aplicabilidad Validación
IMP-COS-01	El precio de venta del producto deberá ser inferior a \$1500 (al año 2017)	REQ-10	DF, IF D

Cuadro 13: Especificaciones de costos

7.3.5. Especificaciones de Servicio (RAMS)

ID	Descripción	Origen	Aplicabilidad Validación
RAM-CON-01	El producto deberá tener una vida útil no menor a 10 años teniendo en cuenta el promedio de edad del parque automotor argentino.	REQ-17 Informe AFAC [12]	DF, IF D
RAM-CON-02	La MTBF debe ser superior a 1 año, ya que se estima que en promedio esa es la frecuencia a la que se realizan las tareas de mantenimiento del vehículo.	REQ-18	DF, IF D

Cuadro 14: Especificaciones de Confiabilidad

ID	Descripción	Origen	Aplicabilidad Validación
RAM-MAN-01	Si el equipo no puede realizar su función, esto deberá ser claramente indicado.	REQ-06	DP, DF I, D, T
RAM-MAN-02	Cualquier cambio de repuestos deberá realizarse en un estado desenergizado.	REQ-02	IP, IF D
RAM-MAN-03	Deberá haber una indicación visual clara cuando el equipo esté en funcionamiento.	REQ-02 REQ-06	DP, DF I, D, T
RAM-MAN-04	En caso de utilizar SW o FW, deberá ser posible para técnicos calificados realizar actualizaciones del mismo.	REQ-06	DP, IP, DF, IF D
RAM-MAN-05	En caso de requerir una calibración, esta será permitida como parte del producto final.	REQ-04 REQ-05 REQ-06	DP, IP, DF, IF D
RAM-MAN-06	El equipo deberá contener la siguiente documentación: <ul style="list-style-type: none"> • Manual de Usuario • Esquemáticos de circuitos • Esquemáticos de Placas 	REQ-06	DF D

Cuadro 15: Especificaciones de Mantenibilidad

ID	Descripción	Origen	Aplicabilidad Validación
RAM-SEG-01	La puesta a tierra de cada parte del equipo deberá ser tal que no puedan aparecer corrientes de superficie en partes metálicas que lleven a descargas en un cuerpo humano de más de 25 mA.	REQ-02	DF, IF D
RAM-SEG-02	No deberá ser posible para una persona realizarse ningún tipo de corte por contacto con el equipo.	REQ-06	DF, IF D
RAM-SEG-03	Si en algún lugar (accesible o no) hay tensiones peligrosas, deberá haber un cartel que lo advierta.	REQ-02	DF, IF D

Cuadro 16: Especificaciones de Seguridad

8. Plan de Validación

8.1. Diseño de Bancos de Pruebas

ID	Requerimientos	Descripción	Especificaciones
01 - Prueba Eléctrica	<ul style="list-style-type: none"> ■ Osciloscopio ■ Fuente de alimentación ■ Puntas de osciloscopio ■ Cables de alimentación de 50cm ■ Carga de prueba 	Medición de la entrada y salida del dispositivo	<ul style="list-style-type: none"> ■ La fuente de alimentación que simula la batería del vehículo debe tener una resolución de al menos 0.1V y poder entregar a la entrada una tensión entre 9.6V y 14.7V. ■ Se deberá usar un osciloscopio con una resolución de 0.1V para poder medir a la entrada y a la salida una tensión entre 9.6V y 14.7V. ■ Se requieren una punta de osciloscopio para medir la entrada del dispositivo y otra para medir su salida. ■ La tierra de ambos artefactos debe estar compartida y la impedancia a tierra debe ser de 0.53Ohms según la norma IEEE 142-2007. ■ La carga fue calculada teniendo en cuenta la máxima tensión con la que se puede alimentar a los espejos y corriente máxima que circula (50mA medido) dando como resultado 290Ω, llevado a 270Ω por ser valor comercial, con una potencia de 1W.
02 - Prueba ambiental	<ul style="list-style-type: none"> ■ Cámara térmica ■ Cámara de humedad ■ Cámara de presión ■ Osciloscopio ■ Batería de auto de 12Vdc ■ Puntas de osciloscopio ■ Cables de alimentación de 30 cm ■ Carga de prueba 	Pruebas ambientales	<ul style="list-style-type: none"> ■ La cámara térmica debe operar en el rango de -10°C a 70°C +/- 0.5°C ■ La cámara de humedad debe operar en el rango de 0% a 100% +/- 0.5%HR ■ La cámara de presión debe operar en el rango de 60 kPa a 110 kPa +/-0.5 kPa ■ Se deberá usar un osciloscopio con una resolución de 0.1V para poder medir a la entrada y a la salida una tensión entre 9.6V y 14.7V. ■ Se requieren una punta de osciloscopio para medir la entrada del dispositivo y otra para medir su salida.

			<ul style="list-style-type: none"> ■ El osciloscopio tener una impedancia a tierra debe ser de 0.53Ohms según la norma IEEE 142-2007. ■ La carga fue calculada teniendo en cuenta la máxima tensión con la que se puede alimentar a los espejos y corriente máxima que circula (50mA medido) dando como resultado 290Ω, llevado a 270Ω por ser valor comercial, con una potencia de 1W.
03 - Pruebas de uso	<ul style="list-style-type: none"> ■ Grupo de personas de estatura variada entre 1.50m y 2.10m 	Someter al dispositivo a un grupo de personas para validar la robustez del mismo	<ul style="list-style-type: none"> ■ Precisión: Círculo de radio 1cm entorno al centro del espejo. ■ Simular variar la profundidad del asiento del conductor entre 60cm y 90cm desde el volante. ■ Los usuarios podrán tener rasgos u objetos (ej. anteojos, gorro, entre otros) que puedan dificultar el reconocimiento de los ojos
04 - Prueba de versatilidad	<ul style="list-style-type: none"> ■ Dimensiones de vehículos de 3 segmentos: compact, sedan y sport utility vehicle (SUV) 	Pruebas de funcionamiento	<ul style="list-style-type: none"> ■ Altura del espejo lateral izquierdo respecto al suelo. ■ Altura de manija trasera izquierda respecto al suelo. ■ Distancia entre espejo lateral izquierdo y manija trasera izquierda. ■ Distancia entre centro del volante y centro del espejo lateral izquierdo. ■ Distancia entre centro del volante y parante izquierdo

Cuadro 17: Bancos de prueba

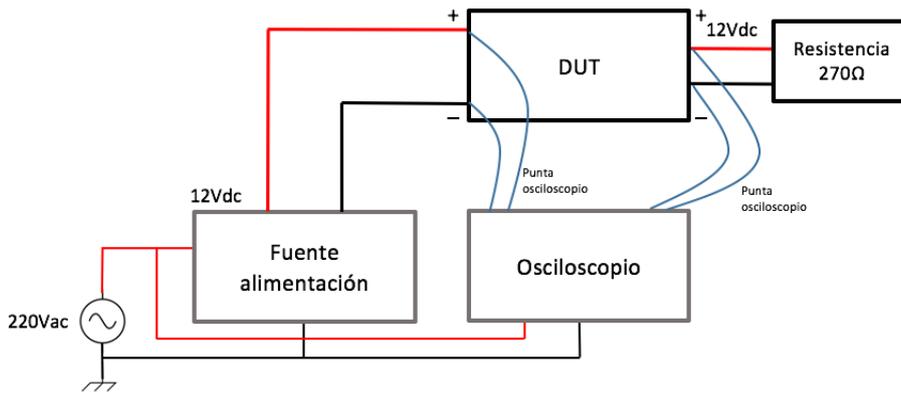


Figura 6: Diagrama de conexión Banco de prueba 1

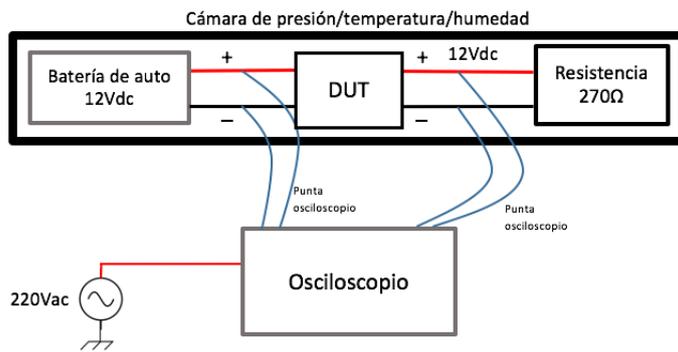


Figura 7: Diagrama de conexión Banco de prueba 2

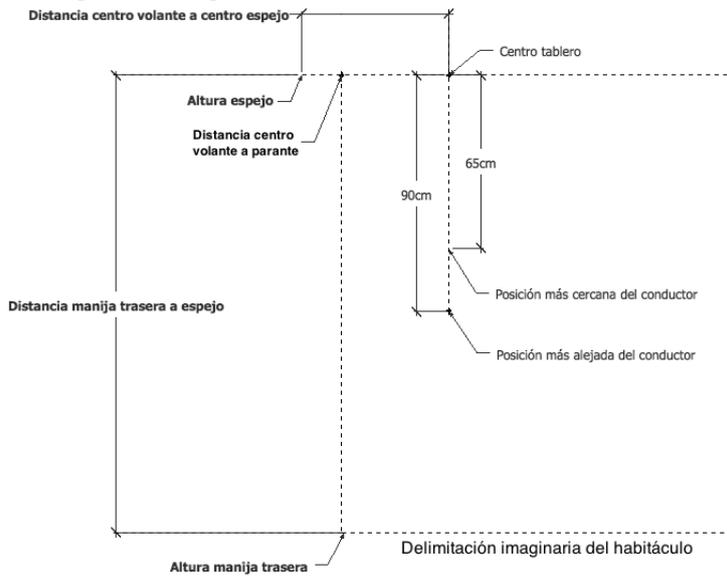


Figura 8: Diagrama de conexión Banco de prueba 3 y 4

Dimensión	Sedán	Compacto	SUV
Altura espejo	95	102	125
Altura manija	80	86	110
Distancia centro volante a centro espejo	55	55	65
Distancia manija trasera a espejo	160	170	170
Distancia centro volante a parante	40	40	40

Cuadro 18: Medidas de habitáculos a simular

8.2. Especificaciones de Pruebas

ID Aplicabilidad	Procedimiento	Criterio
<p>Precondiciones:</p> <p>Procedimiento General: Para todas estas pruebas, se deberá conectar el dispositivo a probar al banco de pruebas #1. Verifique que las puntas del osciloscopio estén correctamente conectadas al osciloscopio, y que tengan sus referencias conectadas al pin de tierra inmediatamente al lado de lo que intentan medir. El canal 1 deberá medir la tensión de entrada VIN y el canal 2 la salida VOUT. El osciloscopio deberá estar ajustado para mostrar en pantalla un período de 10µs (siendo la máxima frecuencia en la banda de ruido). La amplitud que se pueda ver en pantalla deberá ser no mayor a 15VDC (siendo el máximo para los espejos 14.7VDC). El trigger deberá estar puesto en manual, configurado según lo indique el test, y el display deberá estar puesto en estado normal (es decir, sin persistencia). Las puntas deberán estar en impedancia de entrada x10, con el ajuste acorde en el osciloscopio (para influir lo menos posible en la métrica de ruido). Salvo que el test así lo indique, no deberá haber ningún tipo de filtro. Los tests comienzan con todos los artefactos ya conectados y apagados.</p>		
<p>T-INT-01 Proto, Final</p>	<ol style="list-style-type: none"> 1. Verificar que la alimentación esté configurada en 12VDC/3.5A (como una batería de automóvil), que esta y el DUT estén apagados, correctamente conectados y la carga desconectada. 2. Configurar el trigger en flanco ascendente, en el canal de VIN, verticalmente a 1V respecto de la referencia y horizontalmente a 1 cuadrícula respecto del margen izquierdo. 3. Utilizar las funciones de medición del osciloscopio para que se registren los valores de tensión máximos y mínimos en la entrada VIN y salida VOUT del DUT. 4. Verificar que el DUT no ejecutará ningún procedimiento. 5. Encender la alimentación y registrar los valores obtenidos para un período en pantalla. 6. Apagar alimentación. 	<p>Post transitorio $V_{in_{MAX}} = 14,7V$ $V_{in_{MIN}} = 9,6V$ $V_{out} = 0V \pm 0,1V$ Indicador de operación = Apagado</p>
<p>T-INT-02 Proto, Final</p>	<ol style="list-style-type: none"> 1. Verificar que la alimentación esté configurada en 12VDC/3.5A (como una batería de automóvil), que esta y el DUT estén apagados, correctamente conectados y la carga desconectada. 2. Configurar el trigger en modo automático. 3. Utilizar las funciones de medición del osciloscopio para que se registren los valores de tensión máximos y mínimos en la entrada VIN y salida VOUT del DUT. 4. Verificar que el DUT no ejecutará ningún procedimiento. 5. Encender la alimentación y esperar 10 segundos. 6. Variar VIN en el rango de 9.6VDC a 14.7VDC, con pasos de 0.3V. 	<p>$V_{in_{MAX}} = 14,7V$ $V_{in_{MIN}} = 9,6V$ $V_{out} = 0V \pm 0,1V$ Indicador de operación = Apagado</p>

<p>T-INT-03 Proto, Final</p>	<ol style="list-style-type: none"> 1. Verificar que la alimentación esté configurada en 12VDC/3.5A (como una batería de automovil), que esta y el DUT estén apagados, correctamente conectados y la carga desconectada. 2. Configurar el trigger en modo automático. 3. Utilizar las funciones de medición del osciloscopio para que se registren los valores de tensión máximos y mínimos en la entrada VIN y salida VOUT del DUT. 4. Verificar que el DUT no ejecutará ningún procedimiento. 5. Encender la alimentación y esperar 10 segundos. 6. Configurar el DUT para que opere de forma automática (es decir, que no espere la información de la posición del usuario e intente alimentar la carga). 7. Variar VIN en el rango de 9.6VDC a 14.7VDC, con pasos de 0.3V. 	<p>$V_{in_{MAX}} = 14,7V$ $V_{in_{MIN}} = 9,6V$ $V_{out_{MAX}} = 14,7V$ $V_{out_{MIN}} = 9,6V$ Indicador de operación = Encendido</p>
----------------------------------	--	---

Cuadro 19: Tests de Interfaz VIN-VOUT

ID Aplicabilidad	Procedimiento	Criterio
<p>Precondiciones: Deberán haber sido completadas con éxito las pruebas T-INT-01, T-INT-02 y T-INT-03.</p>		
<p>Procedimiento General: Para la siguiente prueba, se deberá conectar el dispositivo a probar a la fuente de alimentación del banco de pruebas #1. Los tests comienzan con todos los artefactos ya conectados y encendidos por no menos de 10 segundos.</p>		
<p>T-MAN-01 Proto, Final</p>	<ol style="list-style-type: none"> 1. Dejar libre el lugar donde iría sentado el conductor. 2. Configurar el DUT para que comience a operar. 3. Esperar 10 segundos. 	<p>Indicador de operación = Encendido Indicador de falla de reconocimiento de rostro = Encendido</p>

Cuadro 20: Tests de Mantenibilidad

ID Aplicabilidad	Procedimiento	Criterio
<p>Precondiciones: Deberán haber sido completadas con éxito las pruebas T-INT-01, T-INT-02, T-INT-03 y T-MAN-01.</p>		
<p>Procedimiento General: Para la siguiente prueba, se deberá conectar el dispositivo a probar a la fuente de alimentación del banco de pruebas #1. Se hará uso del banco de pruebas #3 por lo que los voluntarios deberán estar listos para participar, y del banco de pruebas #4 por lo que se deberá poder manipular fácilmente el entorno para simular los vehículos. Los tests comienzan con todos los artefactos ya conectados y apagados.</p>		
<p>T-FUNC-01 Proto, Final</p>	<ol style="list-style-type: none"> 1. Configurar el escenario para representar el entorno del auto compacto. 2. Encender la alimentación y esperar 10 segundos. 3. Calibrar el DUT para adecuarse a la situación. 4. Colocar al primer voluntario en la posición del conductor, a la distancia al volante que habitualmente se encuentre. 5. Activar el DUT. 6. Esperar 10 segundos. 7. Escoger la opción para estacionar el vehículo y esperar que los espejos se acomoden solos. 8. Repetir los pasos 5 y 6 para cada voluntario, modificando la distancia hacia al volante acorde a las comodidades del voluntario. 9. Apagar la alimentación. 	<p>Indicador de operación = Encendido Manija izquierda trasera, dentro del círculo de precisión Indicador de falla de reconocimiento de rostro = Apagado</p>
<p>T-FUNC-02 Proto, Final</p>	<ol style="list-style-type: none"> 1. Repetir el test T-FUNC-01, esta vez configurando el escenario para representar el entorno del auto sedan. 	<p>Indicador de operación = Encendido Manija izquierda trasera, dentro del círculo de precisión Indicador de falla de reconocimiento de rostro = Apagado</p>
<p>T-FUNC-03 Proto, Final</p>	<ol style="list-style-type: none"> 1. Repetir el test T-FUNC-01, esta vez configurando el escenario para representar el entorno del auto SUV. 	<p>Indicador de operación = Encendido Manija izquierda trasera, dentro del círculo de precisión Indicador de falla de reconocimiento de rostro = Apagado</p>

Cuadro 21: Tests de funcionalidad

ID Aplicabilidad	Procedimiento	Criterio
Precondiciones: Deberán haber sido completadas con éxito las pruebas T-INT-01, T-INT-02, T-INT-03 y T-MAN-01.		
Procedimiento General: Para todas estas pruebas, se deberá conectar el dispositivo a probar al banco de pruebas #1. Verifique que las puntas del osciloscopio estén correctamente conectadas al osciloscopio, y que tengan sus referencias conectadas al pin de tierra inmediatamente al lado de lo que intentan medir. El canal 1 deberá medir la tensión de entrada VIN y el canal 2 la salida VOUT. El osciloscopio deberá estar ajustado para mostrar en pantalla un período de 10µs (siendo la máxima frecuencia en la banda de ruido). La amplitud que se pueda ver en pantalla deberá ser no mayor a 15VDC (siendo el máximo para los espejos 14.7VDC). El trigger deberá estar puesto en manual, configurado según lo indique el test, y el display deberá estar puesto en estado normal (es decir, sin persistencia). Las puntas deberán estar en impedancia de entrada x10, con el ajuste acorde en el osciloscopio (para influir lo menos posible en la métrica de ruido). Salvo que el test así lo indique, no deberá haber ningún tipo de filtro. Los tests comienzan con todos los artefactos ya conectados y apagados.		
T-OP-01 Final	1. Repetir el test T-INT-03 colocando el dispositivo en una cámara térmica que lleve la temperatura a 70°C.	$V_{inMAX} = 14,7V$ $V_{inMIN} = 9,6V$ $V_{outMAX} = 14,7V$ $V_{outMIN} = 9,6V$ Indicador de operación = Encendido
T-OP-02 Final	1. Repetir el test T-INT-03 colocando el dispositivo en una cámara térmica que lleve la temperatura a -10°C.	$V_{inMAX} = 14,7V$ $V_{inMIN} = 9,6V$ $V_{outMAX} = 14,7V$ $V_{outMIN} = 9,6V$ Indicador de operación = Encendido
T-OP-03 Final	1. Repetir el test T-INT-03 colocando el dispositivo en una cámara de humedad que lleve la humedad a 0 %.	$V_{inMAX} = 14,7V$ $V_{inMIN} = 9,6V$ $V_{outMAX} = 14,7V$ $V_{outMIN} = 9,6V$ Indicador de operación = Encendido
T-OP-04 Final	1. Repetir el test T-INT-03 colocando el dispositivo en una cámara de humedad que lleve la humedad a 100 %.	$V_{inMAX} = 14,7V$ $V_{inMIN} = 9,6V$ $V_{outMAX} = 14,7V$ $V_{outMIN} = 9,6V$ Indicador de operación = Encendido
T-OP-05 Final	1. Repetir el test T-INT-03 colocando el dispositivo en una cámara de presión que lleve la presión a 60 kPa.	$V_{inMAX} = 14,7V$ $V_{inMIN} = 9,6V$ $V_{outMAX} = 14,7V$ $V_{outMIN} = 9,6V$ Indicador de operación = Encendido
T-OP-06 Final	1. Repetir el test T-INT-03 colocando el dispositivo en una cámara de presión que lleve la presión a 110 kPa.	$V_{inMAX} = 14,7V$ $V_{inMIN} = 9,6V$ $V_{outMAX} = 14,7V$ $V_{outMIN} = 9,6V$ Indicador de operación = Encendido

Cuadro 22: Tests de operación

8.3. Diseño y Especificaciones de Simulaciones

SIM ID	SIMULACIÓN	DESCRIPCIÓN CORTA
SIM-01	Control de motores	Se coloca un analizador lógico en los puertos GPIO en los que se conectarán los motores. Se utiliza una rutina en la cual se crea un vector entre -1 y 1, con un umbral en 0, con un paso de 0.03. Se asignó a dos GPIO un 1 lógico y a los otros dos un 0 lógico para valores entre -1 y 0. Se recorre el vector y cuando se cruce el umbral, se podrá observar la inversión en los GPIO, es decir, que los motores se moverán en sentido contrario.
SIM-02	Lazo de control	Se desarrolla una rutina a través de la cual se le ordena a los motores rotar una cierta cantidad de grados en algún sentido. Se corrobora que el ángulo es correcto utilizando un láser, colocado en el centro del espejo de forma paralela a la normal del espejo, y una grilla alineada a los ejes de rotación de la IMU. De esta manera, se calcula el ángulo rotado por los motores mediante el desplazamiento del láser.

Cuadro 23: Especificaciones de simulaciones

8.4. Matriz de Trazabilidad de Validación

Origen	REQ ID Descripción corta	ESP ID	TEST ID o SIM ID
Ley Argentina	REQ-01 El producto deberá cumplir con la norma IRAM ISO 4 de seguridad eléctrica REQ-02 El producto deberá cumplir con la norma IRAM ISO 4 de seguridad eléctrica	FUN-POW-02	T-INT-01 T-INT-02 T-INT-03
Tácito	REQ-03 El producto deberá poder operarse en todas las condiciones normales de temperatura y humedad que se pueden encontrar en el territorio argentino.	IMP-OPE-01 IMP-OPE-02 IMP-OPE-03 IMP-AYT-01 IMP-AYT-02 IMP-AYT-03	T-OP-01 T-OP-02 T-OP-03 T-OP-04 T-OP-05 T-OP-06
Tácito	REQ-04 El producto deberá poder funcionar correctamente pese a una desconfiguración manual externa REQ 5 El producto debe ser un sistema robusto que no requiera de calibración continua	FUN-POW-04	T-FUNC-01 T-FUNC-02 T-FUNC-03
Tácito	REQ-06 El producto debe ser sencillo de usar	FUN-POW-03 FUN-POW-04 T-MAN-01 INT-MEC-01 INT-MEC-02	T-FUNC-01 T-FUNC-02 T-FUNC-03
Tácito	REQ-07 El producto debe poseer un tiempo de respuesta rápido	PER-POW-01	T-FUNC-01 T-FUNC-02 T-FUNC-03
Tácito	REQ-08 El producto no debe necesitar una fuente de alimentación externa	FUN-POW-01 PER-POW-02	T-INT-01 T-INT-02 T-INT-03
Tácito	REQ-09 El producto debe poder funcionar en cualquier momento que el auto se encuentre en marcha o en contacto	FUN-POW-02 PER-POW-02	T-FUNC-01 T-FUNC-02 T-FUNC-03
Encuesta	REQ-10 El costo del producto deberá ser menor a U\$S100	IMP-COS-01	No requiere
Encuesta	REQ-11 El producto debe ser posible de incorporar a cualquier vehículo terrestre de 4 ruedas que posea espejos eléctricos.	FUN-POW-01 INT-VIN-01 INT-VIN-02 INT-VOUT-01 INT-VOUT-02 INT-MEC-01	T-FUNC-01 T-FUNC-02 T-FUNC-03
Encuesta	REQ-12 Al momento de estacionar, el usuario debe poder seleccionar si quiere que los espejos se acomoden automáticamente para visualizar el suelo y la rueda lateral de algún lado.	FUN-POW-05	T-FUNC-01 T-FUNC-02 T-FUNC-03
Tácito	REQ-13 El producto debe ser capaz de funcionar sin importar la altura del conductor o la distancia del mismo al volante.	FUN-POW-04	T-FUNC-01 T-FUNC-02 T-FUNC-03
Tácito	REQ-14 El producto no funcionará de forma automática al ser energizados. Su accionar comenzará cuando el usuario lo requiera y de alguna forma de la orden.	RAM-MAN-03	T-INT-01 T-INT-02 T-INT-03
Tácito	REQ-15 El equipo deberá poder funcionar las 24 horas del día siempre que el vehículo se encuentre en contacto.	FUN-POW-06	-

Tácito	REQ-16 De ser necesario un aditamento que cierre el lazo de control, debe poseer dimensiones tal que pueda ser incorporado al receptáculo del espejo lateral.	INT-MEC-01 IMP-DIM-01	No requiere
Tácito	REQ-17 La vida útil del equipo debe ser equiparable a la vida útil de un automóvil.	RAM-CON-01	No requiere
Tácito	REQ-18 El tiempo medio entre fallas debe ser mayor al tiempo medio entre <i>service</i> del vehículo	RAM-CON-02	No requiere
Tácito	REQ-19 El producto debe poder funcionar para el promedio de altura de la población.	FUN-POW-07	T-FUNC-01 T-FUNC-02 T-FUNC-03

Cuadro 24: Matriz de Trazabilidad para Validación

8.5. Plan de Tests

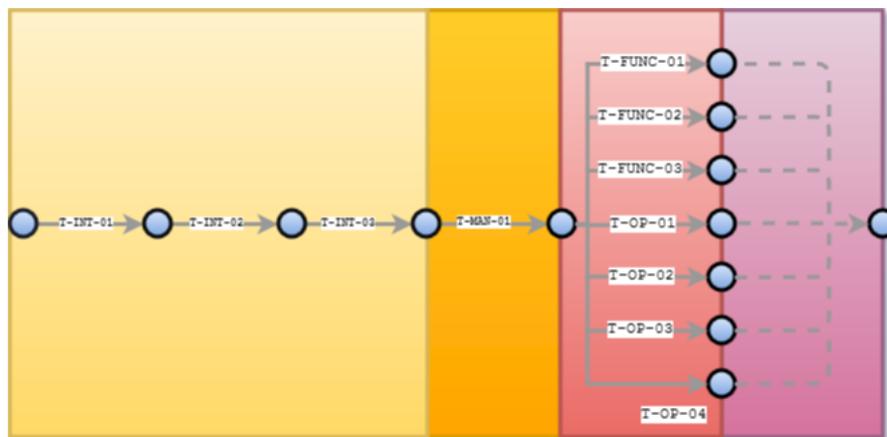


Figura 9: Diagrama de dependencias de Validación

9. Análisis de Factibilidad

El objetivo de este análisis es verificar la posibilidad de crear un módulo capaz de controlar el movimiento de los espejos del automóvil, utilizando las herramientas disponibles en el mercado y los conocimientos adquiridos a lo largo de la carrera. Para evaluar dicha factibilidad, se analizan diversas alternativas que se pueden implementar para lograr el objetivo. Se analizan aspectos económicos, tecnológicos, legales y hasta temporales para determinar cual de todas las posibilidades planteadas es la más viable y que optimiza la solución.

9.1. Factibilidad tecnológica

9.1.1. Explicación teórica

El siguiente inciso tiene por objeto demostrar la posibilidad de la realización de este trabajo.

Es sabido que la luz se comporta de un modo bastante predecible. Si un rayo de luz pudiese ser observado acercándose y reflejándose de un espejo, el comportamiento de la luz al reflejarse seguiría una ley predecible conocida como ley de reflexión. El diagrama a continuación ilustra la idea.

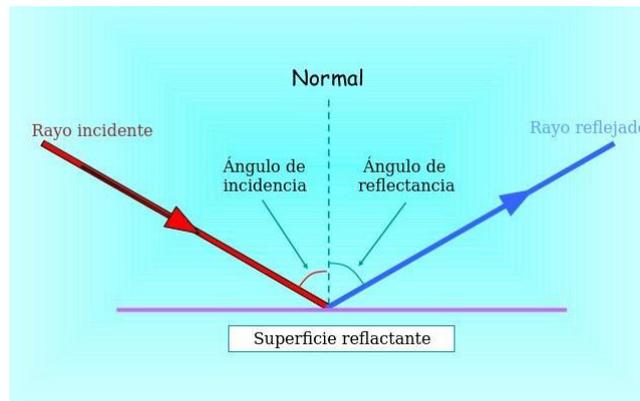


Figura 10: Diagrama de reflexión

En el diagrama, el rayo de luz que se aproxima al espejo se conoce como el rayo incidente. El rayo de luz que sale del espejo se conoce como el rayo reflejado. En el punto de incidencia donde el rayo golpea el espejo, se puede dibujar una línea perpendicular a la superficie del espejo. Esta línea se conoce como una línea normal (aplica también para espejos curvos). La línea normal divide el ángulo entre el rayo incidente y el rayo reflejado en dos ángulos iguales. El ángulo entre el rayo incidente y la normal se conoce como el ángulo de incidencia θ_i . El ángulo entre el rayo reflejado y lo normal se conoce como el ángulo de reflexión θ_r . La ley de la reflexión dice que **cuando un rayo de luz se refleja en una superficie, el ángulo de incidencia es igual al ángulo de reflexión**.

9.1.2. Aplicación al proyecto

A continuación se detalla como funciona el control de posicionamiento del espejo lateral izquierdo. El mismo es aplicable de manera análoga.

Se establece como premisa que al momento de calibrar el sistema de posicionamiento se hace respecto a un punto fijo del automóvil, por ejemplo la puerta trasera del lado del conductor.

La siguiente figura demuestra el sistema de referencia empleado para la determinación de los ángulos.

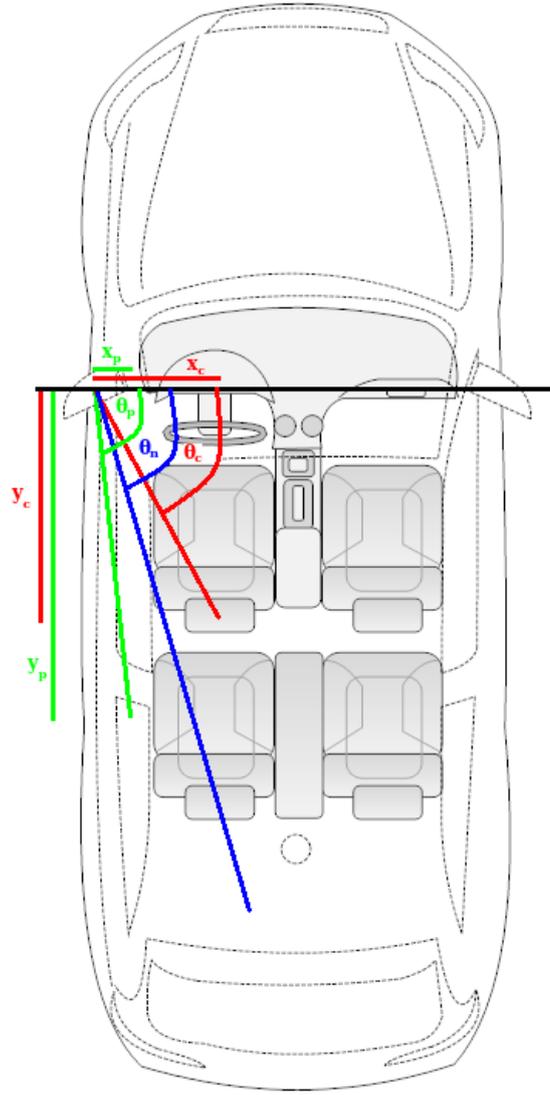


Figura 11: Diagrama óptico

dónde

$$\tan(\theta_p) = \frac{y_p}{x_p}, \quad \tan(\theta_c) = \frac{y_c}{x_c}$$

De acuerdo a la ley de reflexión se establece que como $\theta_p = \theta_c$, el ángulo normal al espejo θ_n es

$$\theta_n = \theta_c + \frac{\Delta\theta}{2} = \theta_c + \frac{\theta_p - \theta_c}{2} = \frac{\theta_c}{2} + \frac{\theta_p}{2}$$

Ahora, al haber calibrado por primera vez el sistema, x_c , siendo la distancia en el eje horizontal del centro del espejo al cabezal del asiento del conductor, puede considerarse constante y conocida de poder establecer θ_c , como así también x_p e y_p la distancia horizontal y vertical del centro del espejo al punto de calibración respectivamente.

Si y_c varía, el nuevo sistema queda determinado por

$$\theta'_n = \frac{\theta'_c}{2} + \frac{\theta'_p}{2} = \frac{\theta'_c}{2} + \frac{\theta_p}{2},$$

reemplazado θ_p en la ecuación anterior

$$\theta'_n = \theta_n + \frac{\theta'_c}{2} - \frac{\theta_c}{2} = \theta_n + \frac{1}{2} \left(\arctan\left(\frac{y'_c}{x'_c}\right) - \arctan\left(\frac{y_c}{x_c}\right) \right) = \theta_n + \frac{1}{2} \left(\arctan\left(\frac{y'_c}{x_c}\right) - \arctan\left(\frac{y_c}{x_c}\right) \right)$$

Es decir, para calcular un nuevo re-acomodamiento la única variable que permanece como incógnita es la nueva posición del conductor respecto al eje vertical.

Este razonamiento hecho respecto a la profundidad del conductor es completamente análogo a la situación de desplazamiento en altura.

9.1.3. Propuesta de alternativas de diseño

9.1.3.1. Obtención de la imagen Se evaluaron dos alternativas que nos permita obtener una imagen utilizable para lograr obtener la posición del conductor. Estas posibilidades son las siguientes:

- Dos cámaras
- Una cámara

Para la primera alternativa, una de las cámaras situadas del lado del conductor, la otra cámara ubicada del lado del acompañante para lograr obtener un mayor campo de visión y poder tomar una imagen más clara para procesar. La idea de esta disposición es la de poder reconstruir una imagen tridimensional a partir de 2 imágenes bidimensionales, imitando el sistema ocular humano, y así poder ubicar al sujeto en el espacio, poder analizar altura y profundidad del conductor, etc. La desventaja es que para garantizar la correcta recomposición del conductor, se necesitan imágenes lo más nítidas posibles. Esto conlleva a que los tiempos de procesamiento sean grandes. Este requerimiento no puede ser reducido debido a la necesidad de poder reconocer los rasgos del conductor por la cámara mas alejada.

Además, si por ejemplo, la cámara del acompañante encuentra algún bloqueo en el campo visual, la reconstrucción no podrá ser realizada. Esto conlleva a una alta dependencia de la disposición de objetos en el habitáculo.

Otra posibilidad radica en eliminar una de las cámaras, la más alejada del conductor, y reemplazarla por un sensor. De este modo, la responsabilidad de la cámara radica únicamente en detectar altura del sujeto mientras que el sensor detecta profundidad del conductor. La desventaja de esta alternativa es la ubicación del sensor: De considerarse sobre la consola del automóvil, debe encontrarse un punto que no sea bloqueado (por el volante por ejemplo). De considerarse en la misma posición que la cámara, el sensor debe tener la capacidad de moverse para poder seguir al sujeto. Colocarlo en el parabrisas queda fuera de consideración ya que puede boquear la visión del conductor.

Un sensor ubicado en la corredera utilizado para detectar la profundidad es muy rápido. Por otro lado, se puede perder precisión dado que no se estaría considerando qué tan reclinado está el respaldo del asiento del conductor. Sin embargo, el error puede resultar lo suficientemente pequeño como para generar una aproximación suficiente para garantizar un correcto posicionamiento de los espejos.

Es por ello que en principio se decidió adoptar esta última opción, utilizando solamente una cámara y un sensor sobre la corredera.

Sin embargo, como será detallado en la sección 9.1.3.5, el sensor fue descartado quedándose solamente con una cámara.

9.1.3.2. Obtención de la posición del espejo Se estudiaron distintos métodos para medir el movimiento de los espejos. El problema surge de que los motores del espejo son de corriente continua, motivo por el cual solo es posible regular su velocidad y sentido de rotación pero no conocer su posición. Sin intención de cambiar los motores se evaluaron dos métodos posibles:

- Medición a mano alzada del movimiento de la cremallera del motor
- Utilización de un sensor para determinar dicho movimiento

El primer método utilizado, a lazo abierto, es el que más error produce. Se intentó estimar cuántos milímetros se mueve la cremallera de cada motor durante cierto período de tiempo. La medición se realizó utilizando un calibre. Se tomó la medida de la ubicación de la cremallera, luego se conectaron los motores a una señal cuadrada de un período de 4 segundos y duty cycle de 50 %. Una vez apagado el motor, se midió la posición final. Este método se realizó seguidas veces con ambos motores. No solo se observó que la velocidad de los motores es distinta, si no que se terminó concluyendo que el método introducía un error muy importante que, sumándose en cada movimiento, haría imposible conocer la posición.

El siguiente método evaluado implica la colocación de algún sensor que pueda determinar el movimiento del espejo de forma más precisa. Para esto, se evaluaron las ventajas y desventajas de los siguientes sensores por su factibilidad de ser colocados en el receptáculo del espejo:

Sensor	Ventaja	Desventaja
Sensor óptico	Gran resolución. Permite realizar mediciones lineales en base a rotación del espejo (transformación rotación-lineal).	Alta susceptibilidad a la luz lo que lo hace vulnerable a diversos ambientes de uso.
Encoder	Gran resolución. Sin necesidad de adicionar lógica para controlar el movimiento, solo contar pasos.	Necesidad de modificar toda la estructura del espejo y/o cambiar los motores del mismo.
Potenciómetro Lineal	Resolución aceptable. Transformación rotación-lineal	Al ser sensores mecánicos, la necesidad de contacto implica ofrecer resistencia al movimiento, lo cual aumenta el consumo de los motores.
IMU	Resolución aceptable. Tamaño diminuto.	Relación simétrica entre el aumento de la resolución con el aumento de la variación en la lectura de los valores. Siendo que el acelerómetro posee mucho error en estado estacionario, es necesario aplicar un filtro pasabajos. Siendo que el giróscopo tiene mucho error para variaciones rápidas, es necesario un filtro pasa bajos.

Teniendo en cuenta el análisis recién realizado, se dispone contar con una IMU y aplicar las consideraciones mencionadas como desventajas.

Se debe aclarar que el sistema de comunicación de la IMU es I²C por lo que el dispositivo que interactúe con ella debe contar con dicho puerto.

9.1.3.3. Microcontrolador Es necesario una unidad de procesamiento que trabaje con la imagen obtenida por la cámara para aislar los ojos. Es necesario que esta unidad sea capaz de procesar imágenes rápidamente. Además debe contar con puertos GPIO para poder controlar los motores, contar con I²C para interactuar con la IMU y ser de bajo consumo dado que va a estar alimentado por la batería del vehículo. Entre las opciones evaluadas se encuentran

Unidad de Procesamiento	Ventajas	Desventajas
Micro-procesador	Alta velocidad de procesamiento	Requiere dimensionar todo periférico que se le sea conectado
Micro-controlador	Velocidad de procesamiento aceptable.	Escalabilidad. Cualquier cambio físico que se le desee hacer, no será posible.
DSP	Altísima velocidad de procesamiento	Costo muy elevado. Se desperdicia enormemente su capacidad ante una aplicación con poco costo computacional.
Micro-Procesador + GPU	Alta velocidad de procesamiento para aplicaciones gráficas	Costo elevado. Se desperdicia su capacidad de procesamiento ante una única aplicación

Para minimizar los costos del prototipo y dispositivo final, dimensionar el producto a modo de abarcar una gran variedad de capacidades de las computadoras de a bordo de los automóviles y aprovechar al máximo sus capacidades de procesamiento es que se decidió por el desarrollo en un micro-controlador.

Se decide optar por el VoCore2 ya que no solo cumple con los requerimientos detallados si no que además se puede encontrar gran cantidad de bibliografía sobre su utilización.

9.1.3.4. Procesamiento de imagen A continuación se plantean las diversas estrategias de programación analizadas a ser implementadas en el proyecto así como también aquellas finalmente seleccionadas. En un plano abarcativo, la única parte que permite versatilidad a la hora de elegir entre diversas alternativas es la de reconocimiento del rostro y es por ello que es en la que más cuidado se tuvo.

Es el paso inmediatamente posterior a la captura de imagen y aquel que conlleva más tiempo.

Reconocimiento mediante correlador Puede pensarse a la correlación como una cualidad que indica la similitud entre dos señales del mismo tipo.

Luego de capturar una imagen de la cabina, la misma podría ser correlacionada con un banco de imágenes patrón, a las cuales ya se les hayan extraído sus características. Es decir, comparar la imagen frente a un grupo del que ya se conocen los parámetros necesarios para calcular la posición de los espejos.

La principal ventaja que brinda esta opción es su velocidad de procesamiento por correlación. Además, con un sistema con múltiples núcleos la comparación podría realizarse en paralelo cuadruplicando, para el caso de 4 núcleos, la velocidad de procesamiento.

Por otro lado, se necesitarían demasiadas imágenes para identificar cada caso en particular teniendo en cuenta facciones del rostro, posición, etc., lo que implica que su puesta en marcha de modo robusto resulta lento y costosa ya por el espacio de memoria necesario. A su vez, haciendo uso de este método se pierde versatilidad ya que luego de todo el procesamiento el resultado final sería a qué imagen conocida se parece más y se perdería granularidad.

Reconocimiento mediante red neuronal Mucho estudio se ha realizado a lo largo de la historia a la hora de poder dotar un algoritmo con suficiente capacidad para discernir como un ser humano. Varios modelos de redes neuronales podrían ser implementados en este proyecto en particular, con cualidades de ser supervisados o no, con resultado binario (sí o no), etc.

Frente a la propuesta anterior, una red neuronal no requiere de mucha memoria salvo en la etapa de entrada ya que lo único necesario son valores de las neuronas y de conexión entre ellas. Su tiempo está directamente relacionado con el grado de robustez que se requiera: Una red capaz de discernir de forma correcta frente a un gran abanico de posibilidades podría resultar muy compleja.

La cualidad por la que dicha alternativa fue descartada y se hace presente en todo tipo de red es su resultado final. Una red de resultado binario finalizará el procesamiento decidiendo si la imagen contiene o no un rostro y ello no es el objetivo final del proyecto. Por otro lado, una red con resultado fraccional (como back-propagation) tendrá el mismo defecto que usar correlador, que es que su salida da una idea que tan parecido es a un caso conocido pero no permite extraer datos de casos nuevos.

Reconocimiento mediante Viola-Jones La estructura descrita por Paul Viola y Michael Jones en 2001[1] representa el primer detector de objetos con tasas competentes de detección. Su uso se extiende hasta las necesidades del usuario, sin embargo su motivación fue dada por la detección de caras.

Las características que destacan al algoritmo de Viola-Jones son:

- Robustez. Alta tasa de detección y baja tasa de falsos positivos.
- Tiempo real. Para aplicaciones prácticas al menos 2 cuadros por segundo son procesados.
- Detección vs reconocimiento. El objetivo es distinguir objetos de otros y conforma el primer paso en la línea de reconocimiento.

Además de que este método es utilizado en casi toda aplicación que requiera detección de rostros, posee dos cualidades llamativas en este proyecto: Al presentar la posibilidad de ser entrenado con diversas imágenes, no se encuentra limitado a su uso en rostros de frente y permite utilizar rostros a 45°. Además, Viola-Jones identifica en la imagen los rasgos deseados a distinguir a diferencia de los dos métodos previamente mencionados.

Si bien el tiempo de entrenamiento puede resultar grande (en cuanto se requiera mayor robustez) es un procedimiento que se realiza por única vez fuera del producto final. El código a utilizar ya posee las cualidades de detector por lo que su implementación es rápida y eficaz.

Por robustez, costo computacional, velocidad y portabilidad frente a diversas plataformas es que Viola-Jones es el método elegido a ser aplicado al proyecto.

9.1.3.5. Correcciones por experimentación Al momento de llevar a cabo la segunda alternativa para determinar la posición del conductor, se percató que el código de reconocimiento de ojos es lo suficientemente robusto para detectar profundidad, así como altura.

Considerando que el sujeto permanecerá centrado respecto del volante, la proyección del mismo sobre el plano inclinado de la cámara solo se verá afectada por variaciones de profundidad. Es decir, la coordenada espacial horizontal del sujeto solo variará cuando varíe su distancia al volante ya que las únicas maneras de variar posición son verticalmente y en profundidad.

A través de esta cualidad es que se decidió prescindir de un sensor de profundidad, quedando un proyecto más barato y sencillo sin dejar de lado la funcionalidad.

9.1.4. DFMEA

Se adjunta el análisis de fallas y sus efectos. Esta herramienta se perfeccionó constantemente con el objetivo de establecer un seguimiento de los obstáculos encontrados durante el desarrollo. De esta manera, se tabuló la falla encontrada y su resolución para que quede registrado en caso de ser necesario una nueva reparación. La tabla que se muestra adjunta presenta un progreso temporal.

#	Módulo	Modo de falla potencial	Efectos potenciales de la falla	Causas potenciales de falla	Severidad (S)	Detección (D)	Ocurriencia (O)	RPN	Control de prevención actual	Control de detección actual	Acciones recomendadas	Responsable	Fecha objetivo de resolución	Acción Tomada	S	D	O	RPN
1	Alimentación	Valores de alimentación inadecuados	No se puede garantizar la funcionalidad	Envejecimiento de batería, sobre poco	80	10	70	56000	-	-	Incorporar un módulo capaz de proteger el sistema	SH	15-05-2017	Incorporación de módulo step-down	80	10	20	16000
2	Alimentación	Inadecuada sujeción del conector de alimentación	No se puede garantizar la funcionalidad	Conector débil	100	10	40	40000	-	-	Reforzar el conector entre el módulo y la alimentación	BM	20-05-2017	Incrementar longitud del conector con la batería	100	10	5	5000
3	Procesamiento	No se encuentra condición de fin en rutina de procesamiento	Biqueno del sistema hasta reinicia total	Imposibilidad de detectar error	100	10	30	30000	-	-	Incorporar una condición de fin de procesamiento	SH	21-05-2017	Time-out de procesamiento	30	10	30	6000
4	Procesamiento	Error en la lectura de interrupciones	Funcionalidad azarosa	Alta sensibilidad del interruptor	80	10	70	56000	Sistema anti rebote para activar el interruptor	-	-	BM	04-07-2017	-	80	10	10	8000
5	Detección	Otros no reconocidos	Imposibilidad de garantizar la correcta detección	Distorsión por el efecto de la imagen	100	30	40	120000	Algoritmo de detección más sensible	-	Quedarse quieto mientras se toma la foto	SH	20-05-2017	Time-out de procesamiento. En caso de no encontrar 'gts', tomar punto medio de rostro reconocido	40	30	40	48000
6	Detección	Rostro no reconocido	Imposibilidad de funcionar	Imposibilidad de resolver el rostro dada la calidad de la imagen	100	10	30	30000	Camara con resolución mínima necesaria	-	Quedarse quieto mientras se toma la foto	SH	20-05-2017	Indicar mediante led que se debe volver a hacer la foto	100	10	5	5000
7	Detección	Inadecuada sujeción del conector de la cámara	Imposibilidad de detectar al conductor	Conector débil	100	10	40	40000	-	-	Reforzar el conector entre el módulo de procesamiento y la cámara	BM	02-08-2017	Soldar el conector de la cámara directamente a la placa	100	10	5	5000
8	Detección	Imposibilidad de detectar al conductor en todo el recorrido del asiento	Imposibilidad de asegurar al conductor	Imposibilidad de resolver el rostro dada la calidad de la imagen	100	10	40	40000	Imposibilidad de resolver el rostro dada la calidad de la imagen	-	Calcular altura del parante a la cual la cámara capte al conductor en todo el recorrido del asiento	BM	22-05-2017	Posicionar la cámara a 3/4 de la altura del parante	100	10	10	10000
9	Motores	Controlado en los variables de los motores de los espejos	Quemar el módulo de control	Conexión de los motores a una fuente de alimentación	100	30	50	150000	Imposibilidad de mover ambos motores del espejo a la vez	-	-	SH	10-04-2017	Imposibilitar mover ambos motores del espejo a la vez	100	30	0	0
10	Motores	Inadecuada sujeción de los conectores de los motores	Imposibilidad de garantizar el movimiento del motor	Conector débil	100	10	10	10000	-	-	Reforzar el conector entre el módulo de procesamiento y los motores	BM	-	Cablear conector con tibia	100	10	10	10000
11	MU	Inadecuada sujeción de la IMU al módulo de procesamiento	Imposibilidad de garantizar un correcto lazo de control	Conector débil	80	10	10	8000	-	-	Reforzar el conector entre el módulo de procesamiento y la IMU	BM	-	Cablear conector con tibia	80	10	10	8000
12	MU	Alta variación en los valores medidos de la IMU	Imposibilidad de garantizar un correcto lazo de control	Resolución muy alta del giroscopio/accelerometro	70	10	80	56000	-	-	Fitjar los valores lidos por la IMU	SH	14-05-2017	Filtro de media para aplicación y muestreo por separado. Filtro de Kalman entre ambos	70	10	20	14000
13	MU	Desajuste del sensor del espejo	Imposibilidad de garantizar un correcto lazo de control	Inadecuada sujeción al espejo de la IMU	80	90	50	360000	Reforzar la sujeción mediante cinta	-	Reforzar el soporte entre el espejo y la IMU	BM	20-05-2017	Utilizar mismo adhesivo aplicado al espejo reforzador	80	90	10	72000
14	Procesamiento	Poca resolución para determinar parámetros de calibración	Imposibilidad de garantizar un correcto lazo de control	Método de calibración con poca tolerancia	50	10	80	40000	-	-	Especificar distancia entre espejo y volante al momento de instalar el dispositivo	SH	20-08-2017	Especificar distancia entre espejo y volante al momento de instalar el dispositivo	50	10	10	5000
15	Control	Pérdida de referencia del control	Imposibilidad de garantizar un correcto lazo de control	Atenuación externa de la posición del espejo	80	10	50	40000	Rutina de homing	-	Verificar mediante inspección visual que el sistema no haya sido modificado	SH	-	Rutina de homing	20	10	50	10000
16	Control	Susceptibilidad a la inclinación del vehículo	Imposibilidad de garantizar un correcto lazo de control	Dependencia del sistema de control a la inclinación del vehículo	60	20	80	96000	-	-	Verificar inconsistencia en el control	SH	22-08-2017	Independizar el control del terreno	60	20	40	48000
17	Interfaz con el usuario	Concordancia entre botones y módulo de control	Posibilidad de quemar el módulo de control	Usuario interacciona con botones mientras el sistema está en funcionamiento	100	80	80	640000	-	-	Bloquear al botones cuando el sistema está en funcionamiento	BM	15-05-2017	Indicar mediante led que el sistema está en funcionamiento	100	80	80	64000
18	MU	No poder referenciar ángulos correctamente	Imposibilidad de garantizar un correcto lazo de control	Incorrecta alineación del módulo	80	80	90	576000	-	-	Tomar una referencia robusta para los ángulo	BM	10-04-2017	Alinear eje de rotación vertical de la IMU al eje de rotación vertical del espejo	80	80	20	8000
19	Interfaz con el usuario	Resistencia de programa de posicionamiento	Funcionalidad azarosa	Usuario oprime el botón de activación mientras el sistema está en funcionamiento	40	10	10	4000	Biqueno de interrupción durante ejecución	-	-	SH	28-05-2017	-	40	10	10	4000

Figura 12: DFMEA de diseño modular

Tanto la severidad como el nivel de detección y de ocurrencia se miden en una escala de 0 a 100. La severidad y la ocurrencia son proporcionales a la puntuación. Es decir, mientras más alta la puntuación, más severa y recurrente es la falla. Por otro lado, el nivel de detección es inversamente proporcional. Mientras más alta la puntuación, menos detectable. De este modo, cuando se calcula el producto entre ellos, aquellas fallas con puntuación más alta son las que son consideradas críticas y que requieren acciones para bajar el nivel de criticidad.

9.2. Factibilidad de tiempos

9.2.1. Planificación (PERT y simulación de Montecarlo)

El siguiente cuadro de planificación detalla el tiempo en días que se estima tomará realizar cada tarea.

Código	Actividad	Tiempo Optimista	Tiempo promedio	Tiempo Pesimista
A	Investigación inicial	27	30	35
B	Análisis de mercado	11	14	15
C	Análisis de alternativas	10	14	20
D	Diseño del prototipo	1	3	7
E	Importación/compra de componentes del prototipo	5	7	20
F	Armado de prototipo	18	21	25
G	Diseño de software	12	14	20
H	Implementación del software sobre el prototipo	4	7	12
I	Prueba en vehículos/revisión de software	1	4	8
J	Diseño de esquemáticos para producto final	5	7	10
K	Diseño de PCBs para producto final	3	7	13
L	Documentación final	10	14	16

Cuadro 25: Actividades

Investigación inicial: Profundizar el conocimiento sobre equipamientos actuales presentes en vehículos de alta gama y cuáles de ellos puede ser incorporados a aquellos de gamas bajas sin significar un alto costo.

Análisis de mercado: Analizar cuál el máximo mercado posible a satisfacer.

Análisis de alternativas: Una vez elegido el producto a desarrollar, se investigan distintas alternativas para lograr el objetivo.

Diseño del prototipo: La selección de la alternativa se ve afectada por la disponibilidad de componentes en el mercado. Se buscan los componentes para cada alternativa.

Importación/compra de componentes del prototipo: Una vez seleccionada una alternativa, se compran los componentes necesarios. En el mejor de los casos se compran en el país, de lo contrario hay que importarlos.

Armado de prototipo: Se ensambla el prototipo. En este paso se sueldan las placas que sean necesarias y se las prueba.

Diseño de software: Análisis, simulación, prueba y descarte de las diversas alternativas potenciales a utilizar en el sistema.

Implementación del software sobre el prototipo: Programación del módulo principal.

Prueba en vehículos/revisión de software: Una vez programado el módulo, se procede a probar su funcionamiento tanto con un modelo armado, como también encontrando la mejor ubicación de los sensores en el vehículo. En caso de ser necesario, se revisa el software.

Diseño de esquemáticos para producto final: Confección de circuitos en base a requerimientos particulares del proyecto y disponibilidad de insumos.

Diseño de PCBs para producto final: Realización de las placas a incorporarse al proyecto en base a dimensiones, consideraciones especiales, etc.

Documentación final: Se termina de completar la documentación con información para que el usuario final sepa utilizar el producto.

9.2.2. Simulación de Monte Carlo

A partir de los tiempos optimistas, promedio y pesimistas plasmados en el Cuadro 8, se realiza una simulación de 1000 iteraciones para estimar el promedio de duración del proyecto. Si se considera que cada actividad sigue una distribución Beta, se obtiene el siguiente histograma.

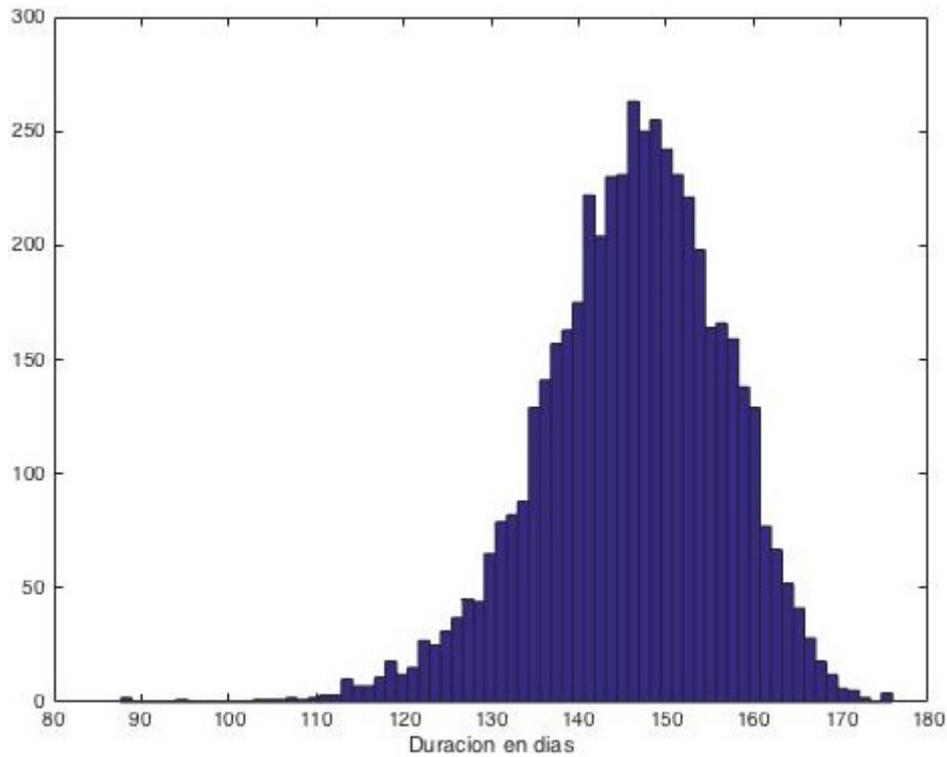


Figura 13: Histograma de duración en días

Como se puede observar en la Figura 13, el tiempo medio de duración del proyecto es de 148 días, con un tiempo optimista de 110 días y un pesimista de 175. Se puede apreciar que el tiempo promedio de duración es comparable con el tiempo obtenido a partir de la programación delineada en el siguiente punto.

9.2.3. Programación (Gantt)

Denominamos el Gantt de la figura 14 como el diagrama de Gantt ideal.



Figura 14: Diagrama de Gantt

Durante el desarrollo del proyecto surgieron problemas de índole particular que obligaron a demorar el el trabajo durante aproximadamente un año.

9.2.4. PERT

En el diagrama a continuación se gráfica el flujo del proyecto. Se indican las fechas tempranas de inicio y finalización, así como también las fechas tardías de comienzo y fin. El camino crítico esta indicado en color rojo.

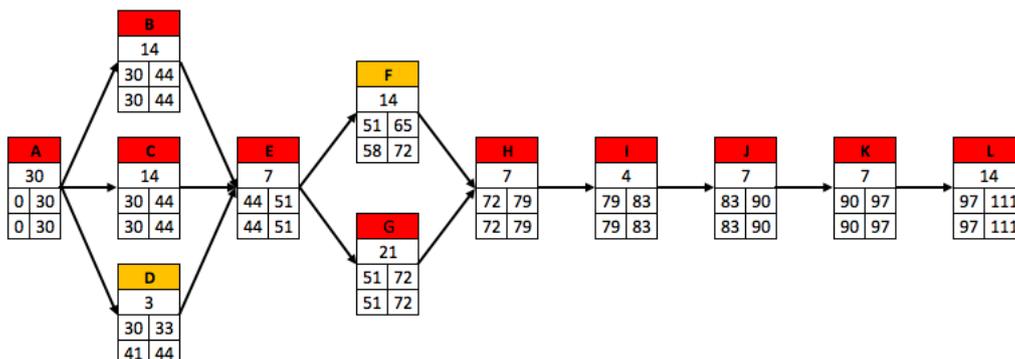


Figura 15: Diagrama PERT

9.3. Factibilidad económica

9.3.1. Mercado

El mercado automotor argentino se ha visto envuelto en una turbulencia económica provocando graves caídas en los últimos dos años. Esto se debe a la imposición de un gravamen a los autos que superen cierto monto en su precio de fábrica (previo a otros gravámenes). En el año 2014, las ventas de 0km resultaron en 683.485, casi un 30 % menos que el 2013.

Se apunta a las empresas con mayor participación en el mercado. Durante 2014 las tres empresas con más ventas fueron Volkswagen (17 %), Ford (14 %) y Fiat (13 %), unas 350 mil unidades[13].

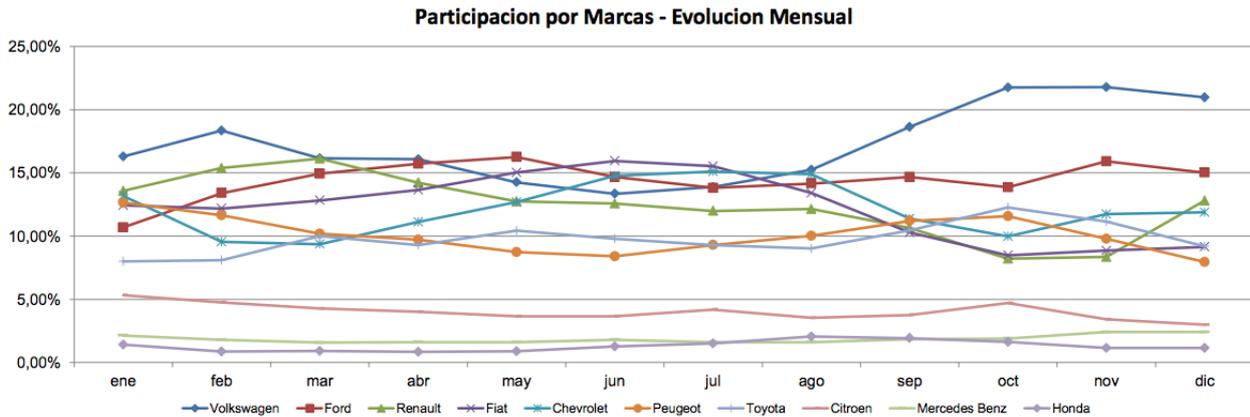


Figura 16: Participación en 2014

Según el informe de la AFAC del año 2016, la flota circulante a fines del año 2015 en el territorio de la República Argentina fue de 12.012.150 vehículos. Si tenemos en cuenta que la incorporación del espejo eléctrico como una función estándar en vehículos de gama media/alta se dio en los últimos 10 años, se puede esperar que la mayoría de aquellos de dicha gama cuya antigüedad sea de 10 años o menos la incluyan. Siguiendo el informe de AFAC, el 55.39 % del parque automotor tiene una antigüedad menor a 10 años.

Ahora bien, suponiendo que el proyecto no resulte como un *commodity*, y vaya a ser incorporado como un accesorio lujoso, se intentó clasificar a aquellos modelos a los que podría ser incorporado. Para ello, se determinó como umbral de lujo a aquellos autos que posean espejos plegables.

Para las tres marcas mencionadas, se investigó este aspecto por medio de sus páginas oficiales arrojando que Volkswagen tiene un 40 % de modelos de lujo (con base en 15 modelos), Ford 35 % (con base en 28 modelos) y Fiat 28 % (con base en 22 modelos).

Por ende se estima que el mercado potencial al cabo de 5 años sea del parque automotor aquellos cuya antigüedad sea menor a 10 años, que sean de las marcas mencionadas y de cada marca aquellos que sean modelos de lujo, es decir 1.020.651 vehículos.

9.3.2. Costo estimado

Las alternativas analizadas de prototipo presentaban como costo la adquisición de una placa *Raspberry Pi* para el procesamiento de imágenes y señales. A pesar de que este costo está incluido en el flujo de caja como una inversión, no fue considerado a la hora de elegir entre las alternativas para evaluar cuál era más viable económicamente. En este sentido, también resultó con más ventajas la alternativa 2 teniendo en cuenta el precio de una cámara contra el del sensor infrarrojo. Esta diferencia se puede utilizar, en caso de ser necesario, para invertir en una cámara con mejor definición que ayude a reducir los tiempos de procesamiento.

Además, por tratarse que el prototipo es una etapa anterior al desarrollo del producto final y puede considerarse como investigación, el alcance de este llega hasta el control de tan sólo uno de los dos espejos laterales del automóvil, tratándose de el del lado del conductor. Esto también tiene soporte en el hecho de que cerrar el control del otro espejo conlleva el mismo hardware y software, aunque por encontrarse el conductor a una distancia mayor al parante del acompañante, las variaciones en ángulo respecto al espejo serán menores y no se llegaría a apreciar la funcionalidad del sistema.

Ítem	Precio/unidad (U\$D)	Cantidad
Raspberry Pi	35.00	1
IMU	15.00	1
Cámara	12.00	1
Puente H (doble)	1.50	3
LM2596	3	1
Gabinete	9	1
Cables	0.0014	100cm
Botones	1.10	2
Conectores	0.0212	12
Pulsador	0.0816	1
Conector alimentación	0.50	2
PCB	7.90	1
TOTAL	86.32	

Cuadro 26: Costos prototipo

Teniendo en cuenta los costos listados, se espera que el prototipo del módulo en sí no supere los U\$D 100. Es necesario aclarar que existe un costo extra al adquirir espejos eléctricos para probar el módulo. Este gasto solo debe ser tenido en cuenta como costo en los períodos de prueba. El módulo se utilizará con los espejos ya incluidos sin necesidad de cambiarlos. Este costo extra ronda los U\$D 50 por espejo.

El mayor factor que contribuye al prototipo es la placa *Raspberry Pi*. Esta placa no puede ser utilizada con fines comerciales motivo por el cual, el producto final debe tener una placa enteramente propia. Esto reduce mucho el costo dado que no se incluirán muchas prestaciones de la *Raspberry Pi* que no se utilizan en el prototipo.

9.3.3. Modelos de negocio

El modelo de negocios aquí planteado se basa en la comercialización de un sistema formado por una placa (con microcontrolador y software incluido) y una cámara web.

Debe considerarse que para lograr llevar a cabo dicho modelo se requiere incurrir en un costo operativo por pago de propiedad de planta y equipos (se estaría alquilando un lugar de pequeño tamaño).

Además, se ve la necesidad de contratar solamente a 3 trabajadores, al cabo de un año calendario, de los cuales no se requiere que posean un nivel educación profesional superior (universitario) ya que serán capacitados por los autores en cuanto a lo que sea requerido para llevar a cabo el proyecto. El salario bruto de los mismos fue considerado tras un relevamiento por diferentes consultoras en dicho ámbito.

Se debe, además, contemplar los siguientes factores:

- Contribuciones a la seguridad social de los empleados[17].
- Impuesto a la ganancia (IG) de acuerdo a lo estipulado por el Banco Central de la República Argentina.
- Costos legales iniciales estimados.
- Impuesto al valor agregado (IVA) de acuerdo a lo estipulado por el Banco Central de la República Argentina.
- Inflación semestral para la actualización media de los salarios en cuestión.
- Porcentaje de ingresos brutos (IIBB) de acuerdo a lo estipulado por el Banco Central de la República Argentina.
- Tasa de interés de acuerdo a lo estipulado por el Banco Central de la República Argentina.
- Salario bruto de los autores.

Para estimar la cantidad de productos a producir por año se siguió el siguiente razonamiento.

1. Se estima, por la simplicidad del producto en cuanto a su colocación, que una persona capacitada puede colocar 8 sistemas al cabo de un día.
2. Luego del primer año, la producción crecerá 150 % debido a la incorporación de nuevos empleados.

3. Durante el año 2017, en Argentina hubieron 248 días laborales[18].

Finalmente, el precio de venta fue contemplado para lograr una TIR de alrededor del 30 % y cuyo valor no exceda lo relevado en la encuesta (menor a U\$S130 al año 2017).

Salario Mensual por Autor	\$30,000	Salario Anual Autores	\$720,000
Contribución Mensual Empleado	0.35		
Salario Mensual por Empleado	\$18,000	Salario Anual Empleados	\$874,800
Costos Operativos Mensuales	\$18,000	Costos Operativos Anuales	\$216,000
Costos Legales	\$12,000		
Prototipado	\$1,500		
Precio Placa	\$128		
Precio Cámara	\$192		
Impuesto a las Ganancias	0.35		
Ingresos Brutos	0.04		
Tasa de Interés	0.25		
Inflación	1.17		
Impuesto al Valor Agregado	0.21		
Precio de venta	\$ 850.00		

Figura 17: Tabla de valores de referencia

Año	0	1	2	3	4	5
Producción	4000	10720	10720	10720	10720	10720
Inversión	-\$ 13,500.00	\$ 0.00	\$ 0.00	\$ 0.00	\$ 0.00	\$ 0.00
Salarios	-\$ 720,000.00	-\$ 1,865,916.00	-\$ 2,183,121.72	-\$ 2,554,252.41	-\$ 2,988,475.32	-\$ 3,496,516.13
Costos Fijos	-\$ 216,000.00	-\$ 252,720.00	-\$ 295,682.40	-\$ 345,948.41	-\$ 404,759.64	-\$ 473,568.78
Costos Variables de Producción	-\$ 1,280,000.00	-\$ 4,013,568.00	-\$ 4,695,874.56	-\$ 5,494,173.24	-\$ 6,428,182.69	-\$ 7,520,973.74
Ingresos	\$ 3,400,000.00	\$ 10,661,040.00	\$ 12,473,416.80	\$ 14,593,897.66	\$ 17,074,860.26	\$ 19,977,586.50
Ingresos Brutos	\$ 136,000.00	\$ 426,441.60	\$ 498,936.67	\$ 583,755.91	\$ 682,994.41	\$ 799,103.46
Impuesto a las ganancias	\$ 0.00	-\$ 1,585,092.60	-\$ 1,854,558.34	-\$ 2,169,833.26	-\$ 2,538,704.91	-\$ 2,970,284.75
Impuesto al Valor Agregado	-\$ 28,560.00	-\$ 89,552.74	-\$ 104,776.70	-\$ 122,588.74	-\$ 143,428.83	-\$ 167,811.73
Flujo Neto	\$ 1,277,940.00	\$ 3,280,632.26	\$ 3,838,339.75	\$ 4,490,857.51	\$ 5,254,303.28	\$ 6,147,534.84
VAN	\$ 1,264,440.00	\$ 2,624,505.81	\$ 2,456,537.44	\$ 2,299,319.04	\$ 2,152,162.62	\$ 2,014,424.22
VAN acc.	\$ 1,264,440.00	\$ 3,888,945.81	\$ 6,345,483.25	\$ 8,644,802.29	\$ 10,796,964.92	\$ 12,811,389.13

TIR(3años)	32%
TIR(5años)	32%

Figura 18: Flujo de fondos

9.3.3.1. Comentarios adicionales El sistema propuesto abre las posibilidad al licenciamiento del software desarrollado para su posible venta. El costo del desarrollo de dicho modelo se puede considerar como un costo hundido dentro del desarrollo del prototipo.

9.4. Factibilidad legal y responsabilidad civil

Las normas legales a las cuales se debe atener están descritas en la Resolución N° 32/94 del Mercosur. Allí se especifica lo que se denomina "CAMPO DE VISIÓN" en la sección 5.4 de la citada resolución.

"CAMPO DE VISIÓN"

Los campos de visión definidos abajo, serán establecidos utilizando visión ambinocular; los ojos, estando en "los puntos oculares del conductor" [...].

Ellos se establecerán a través de las ventanillas que tienen un factor de transmisión total de luz de; como mínimo un 70 % de la medida normal de la superficie.

Especios retrovisores exteriores lado izquierdo

El campo de visión será tal que el conductor pueda ver como mínimo una amplitud de 2.50m vertical perpendicular al plano medio llanos sobre la porción horizontal de su ruta, la cual es circunscrita a la derecha

por el plano paralelo al plano medio longitudinal/vertical del vehículo pasando a través del punto más saliente del vehículo a la izquierda y extenderse hasta 10m detrás de los puntos oculares del conductor hacia el horizonte (VER FIGURA 8).

Espejos retrovisores exteriores lado derecho

El campo de visión será tal que el conductor pueda ver como mínimo una amplitud de 3,50m vertical perpendicular al plano medio llanos sobre la porción horizontal de su ruta, la cual es circunscrita a la derecha por el plano paralelo al plano medio longitudinal/vertical del vehículo pasando a través del punto más saliente del vehículo a la derecha y extenderse hasta 30m detrás de los puntos oculares del conductor hacia el horizonte.

Además la ruta será visible para el conductor una amplitud de 0.75m desde un punto de 4m ubicado detrás del plano vertical que atraviesa los puntos oculares del conductor (VER FIGURA 8)."

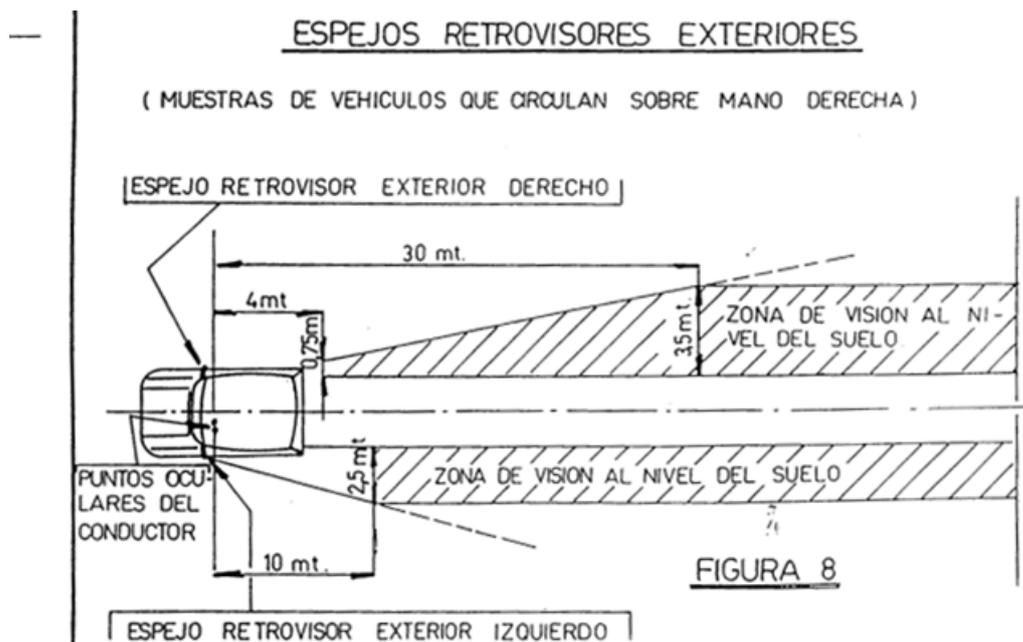


Figura 19: Esquema de espejos retrovisores exteriores

9.4.1. Seguridad eléctrica

"La Resolución N° 508/2015 de la Secretaría de Comercio, exige que el equipamiento eléctrico de baja tensión que se comercializa dentro del país cuente con una certificación que acredite el cumplimiento de los requisitos esenciales de seguridad, y debe ser emitida por un Organismo de Certificación acreditado por el OAA (Organismo Argentino de Acreditación) y reconocido por el Gobierno Nacional.

Esta Resolución establece, entre otras cosas, el cumplimiento de las normas IRAM o IEC aplicables a cada producto. De acuerdo a lo establecido en la Resolución N° 508/2015 se pueden aplicar los Sistemas de Certificación ISO 4, ISO 5 o ISO 7 dependiendo del tipo de equipamiento eléctrico."[2]

En nuestro caso, aplica la certificación ISO 4 que comprende "los ensayos de tipo sobre un producto representativo de la familia a través de ensayos de muestras provenientes del mercado o de fábrica. La formalización del presente sistema se realiza con el certificado de conformidad."[3]

9.4.2. Normas aplicables sobre la compatibilidad electromagnéticas

Las normas sobre la compatibilidad electromagnéticas están diseñadas de forma de establecer límites para que la radiación emitida por cualquier componente instalado en el vehículo no interfiera con el resto de los sistemas presentes en dicho vehículo. Para lo cual se requiere cumplir con requisitos de compatibilidad electromagnética (EMC) y de interferencia electromagnética (EMI).

CISPR 25 es una norma estándar que presenta varios métodos que sugieren límites para evaluar las emisiones de los componentes a instalar.

Otros estándares están dados por las normas ISO 11451 específicas para EMC en vehículos.

9.4.3. Patentes

Una posibilidad a considerar en el futuro es tramitar una patente regional para el producto.

A tenor de esto último es necesario tener en cuenta lo siguiente:

Se conoce la existencia de las patentes "Automatically adjusting side mirror assembly for a vehicle US 8285457 B1" y "Automatic rearview mirror adjustment system for vehicle US 20100017071 A1" las cuales presentan ambas un sistema similar al desarrollado en este informe.

La primer patente ya venció, además de que sólo poseía alcance solo dentro de EEUU. La segunda sólo tiene jurisdicción en China, EEUU y Alemania.

Dadas estas premisas, existiría la posibilidad de tramitar la patente del producto dentro del ámbito regional.

10. Ingeniería de detalle

10.1. Hardware

10.1.1. Diagrama de bloques

En el siguiente diagrama se presenta la composición en bloques del sistema.

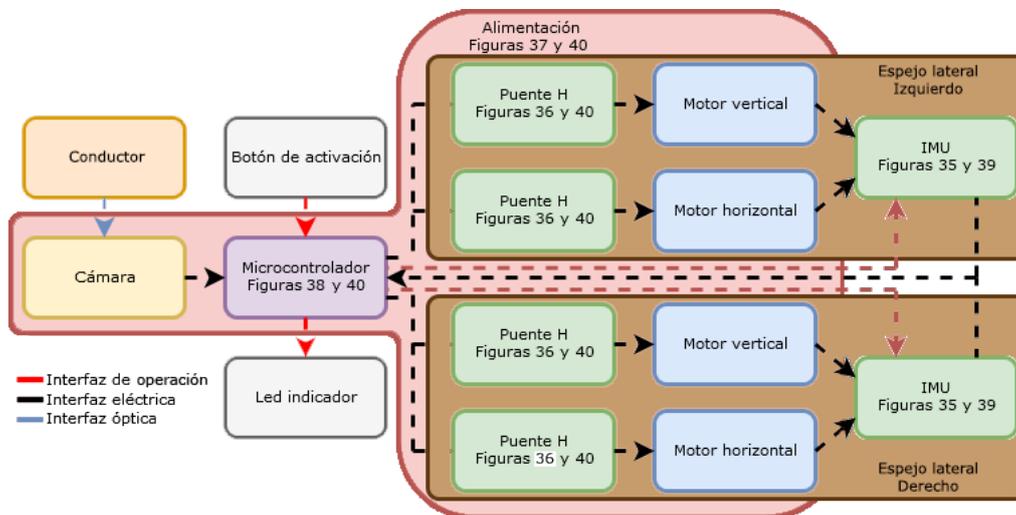


Figura 20: Diagrama en bloques

El color celeste, distinto al de la interfaz óptica, del bloque de los motores indican que los mismos son externos al sistema desarrollado.

Las IMUs y los puentes H poseen el mismo color por tener contacto directo con los motores.

El microcontrolador y la cámara poseen dos colores distintos por tratarse de dos unidades separadas.

Por medio de la interfaz eléctrica se alimentará con la batería del automóvil al microprocesador (en el caso del prototipo, la Raspberry Pi), la cámara, los puentes H necesarios para mover los motores y los motores para cuando el usuario requiera moverlos de forma manual. Notese que las IMUs serán alimentadas por medio del microcontrolador, por ello las flechas del mismo color que el bloque alimentación.

El bloque cabina fue añadido para poder referenciar la interfaz óptica para la cámara.

El botón y el led forman parte de la interfaz con el usuario.

Se da comienzo a la secuencia oprimiendo un botón que inicia el programa. La cámara y la IMU luego envían sus respectivos datos de vuelta al microprocesador. Una vez calculada la distancia que se deben mover los espejos, se acciona el movimiento de los mismos.

10.1.2. Descripción detallada de cada bloque

Alimentación

Este módulo se ocupa de alimentar los puentes H y el microprocesador. El mismo va conectado directamente a la batería del vehículo, la cual provee 12V. Se encarga de bajar los niveles de tensión para poder alimentar el microprocesador con la tensión adecuada y también mantener los niveles constantes en 12V para que la velocidad de los motores no se vea afectada por variaciones en los niveles de tensión de la batería.

Microcontrolador

Es el responsable de correr el software de reconocimiento de ubicación de los ojos de conductor. Controla la cámara y el sensor para el cálculo de la profundidad.

Cámara

La cámara debe estar ubicada en el habitáculo de forma tal de lograr captar y hacer foco al conductor en la posición más alejada y más cercana al volante. Para lograr captar esto, la cámara es posicionada en el parante izquierdo del auto, a una altura a detallar en la sección 11.3.3.

IMU

La IMU es responsable de cerrar el lazo. Con ella es posible detectar la posición del espejo y mandar esa información al procesador para corregirla según sea necesario de acuerdo a la ubicación de la cara que fue detectada por la cámara. Este ciclo continuará hasta llegar al umbral de error aceptado.

Puente H y motores

Cada espejo contiene en su interior dos motores para permitirle moverse en los ejes X e Y. Es decir que estamos en presencia de cuatro motores. La corriente necesaria para mover cada uno de ellos excede la corriente máxima de los pines GPIO de las *Raspberry Pi*. Esta corriente máxima es de unos 16mA. Con lo cual, es necesario incluir un puente H por cada motor para poder alcanzar la corriente necesaria para acomodar los espejos.

10.1.3. Detalles de selección y cálculo de los elementos circuitales de cada bloque (del final)

Puente H

Utilizando una fuente de tensión, se aplicaron 12V a cada uno de los motores para medir el consumo de corriente que requieren para moverse. Utilizando un multímetro digital UNI-T UT39C se obtuvieron las corrientes necesarias. El motor necesita 30mA (con una precisión +/- (1%+3)) de corriente para moverse libremente en cualquier sentido, y esta aumenta a 50mA cuando alcanza el fin de carrera.

Estos valores son muy elevados para los puertos GPIO del microprocesador. Por ende, se requiere la utilización de un puente H para cada uno de los motores. Se decide utilizar el puente H doble L293D que es capaz de manejar 2 motores de forma independiente. Alimentando el mismo con 12V regulados, se obtiene la corriente necesaria.

Una de las ventajas de este integrado es que cuenta con protección incluida a la entrada. De esta manera no es necesario incluir protección externa que encarecería el desarrollo.

Cámara

El método para la selección de la cámara partió de la base de tomar una imagen de un conductor con la mejor resolución posible de una cámara web y se desarrolló un código reconocedor de centro de ojos que fuese capaz de funcionar para este caso.

Una vez alcanzado este objetivo, se procedió, de forma iterativa, a bajar la resolución de la cámara e intentar reconocer el centro de ojos en la misma situación hasta el punto donde esto no fuese posible.

La decisión de utilizar la mínima resolución posible se hizo en base a que el precio de una cámara web aumenta en función de la resolución deseada y a que el tiempo de respuesta del código aumenta al reducir la resolución.

Las distintas resoluciones fueron 1920x1080, 1280x1024, 1280x720, 800x600, 640x480, 352x288, 320x240, 160x120.

El siguiente cuadro documenta la prueba realizada.

Resolución	1920x1080	1280x1024	1280x720	800x600	640x480	352x288	320x240	160x120
Tiempo medio de reconocimiento	30s	20s	20s	15s	10s	5s	5s	-
Tasa de reconocimientos exitosos	100 %	100 %	100 %	80 %	80 %	20s	10 %	0 %

Cuadro 27: Prueba de resolución de cámara

Finalmente, 640x480 fue la última situación donde el código pudo reconocer los ojos del conductor. Esto fue verificado en base a 5 conductores, de diversas estaturas (entre 1.55m y 2.10m) a diferentes distancias del volante del auto (60cm y 90cm).

Para menores resoluciones fue necesario implementar diversas modificaciones al código que no garantizaban robustez.

Cabe aclarar que las cámaras web poseen un sensor pequeño al punto que analizar su apertura carece de sentido ya que puede considerarse una profundidad de plano grande, por no decir infinita. Es decir, el campo visual de la cámara consiste en la sección de la cabina donde se encontraba el conductor. Si bien secciones circundantes también aparecen en la imagen (como el asiento trasero), la parte con mayor predominio es la del conductor.

IMU

Dentro de todas las opciones que se consideraron para poder controlar la posición de los motores, se debió optar por aquella que ocupase menor espacio, garantizase precisión, rápida respuesta y longevidad.

Una vez decidida la opción de la IMU, el paso siguiente consistió en seleccionar cuál de todos los modelos utilizar.

Teniendo en cuenta que se necesitaba mínimamente giróscopo y acelerómetro, dentro de todas las posibilidades se buscó aquella que fuese más comúnmente utilizada.

Ello conllevó a la selección del módulo LSM9DS0.

10.1.4. Plan de pruebas de cada módulo (del prototipo)

IMU

Se inicia la Terminal del sistema operativo Linux desde la unidad de procesamiento y se inicia el proceso de lectura de todas las direcciones I2C. La primera lectura, sin la IMU conectada mostrará que no existe ningún dispositivo conectado.

Luego se procede a conectar la IMU en los puertos SDA y SCL (señales de I2C datos y I2C *clock* respectivamente) del microcontrolador y se vuelve a leer las direcciones I2C. Se corrobora que el dispositivo fue correctamente reconocido por el microcontrolador al verlo figurar entre las direcciones I2C.

```

      0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:    -- -- -- -- -- -- -- -- -- -- -- -- -- --
10:    -- -- -- -- -- -- -- -- -- -- -- -- 1d -- --
20:    -- -- -- -- -- -- -- -- -- -- -- -- -- --
30:    -- -- -- -- -- -- -- -- -- -- -- -- -- --
40:    -- -- -- -- -- -- -- -- -- -- -- -- -- --
50:    -- -- -- -- -- -- -- -- -- -- -- -- -- --
60:    -- -- -- -- -- -- -- -- -- -- 6b -- -- -- --
70:    -- -- -- -- -- -- -- -- -- -- -- -- -- --

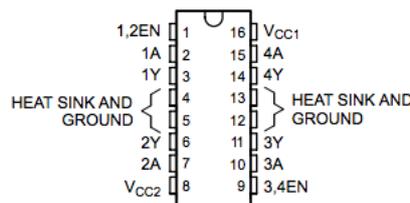
```

Figura 21: Lectura I2C

En la figura 21 se puede ver las dos direcciones I2C de las cuales se levantan los datos de la IMU. El magnetómetro y el acelerómetro comparten dirección (6b) y el giróscopo se encuentra en la dirección 1d. Según el tamaño del registro se puede distinguir entre magnetómetro y acelerómetro.

Puente H

Para probar el funcionamiento del puente H se realizó el siguiente procedimiento.



Pin Functions

PIN		TYPE	DESCRIPTION
NAME	NO.		
1,2EN	1	I	Enable driver channels 1 and 2 (active high input)
<1;4>A	2, 7, 10, 15	I	Driver inputs, noninverting
<1;4>Y	3, 6, 11, 14	O	Driver outputs
3,4EN	9	I	Enable driver channels 3 and 4 (active high input)
GROUND	4, 5, 12, 13	—	Device ground and heat sink pin. Connect to printed-circuit-board ground plane with multiple solid vias
V _{CC1}	16	—	5-V supply for internal logic translation
V _{CC2}	8	—	Power VCC for drivers 4.5 V to 36 V

Figura 22: Distribución de *pins* de L293D

Siguiendo el *layout* de la Figura 22 se conectó una alimentación de 12V en el *pin* 8 que simula ser la alimentación provista por la batería del vehículo. También se aplican 5V en el *pin* 16 como indica el *datasheet* que deben utilizarse para valores lógicos en sus entradas. Para simular los puertos GPIO, se aplicaron 3.3V al *pin*

enable del canal a probar y otros 3.3V al *pin* de *input* correspondiente a ese canal. Lo mismo se realizó con los otros 3 canales y se verificó la salida en cada uno de ellos.

Cámara

Por otro lado, el módulo de la cámara se probó conectándola a la RPi. Corriendo un pequeño programa de prueba utilizando OpenCV, se tomaron distintas fotos para corroborar su funcionamiento.

Lectura IMU

Finalmente el módulo IMU se probó corriendo un programa de prueba provisto por *BerryIMU* que arroja los valores de los distintos sensores que la integran. Dentro de estos valores, los de mayor importancia son los del magnetómetro y giróscopo. Se utiliza un filtro de Kalman para calibrarlos. Como primera prueba, simplemente se movió la IMU manualmente y se verificó la variación de los valores arrojados. Luego, para hacer una prueba más fiel, se acopló la IMU al espejo que será utilizado como parte del prototipo. De esta manera se pudo observar perfectamente que valores se modificaban de acuerdo al movimiento de cada motor y así se pudieron identificar cuales son los que se deben utilizar a la hora del cálculo del posicionamiento de los espejos.

```
Loop Time | 0.01| ACCX Angle 45.00 ACCY Angle -45.00          GRYX Angle 0.00 GYRY Angle 0.01
GRZ Angle -0.03          CFangleX Angle 27.00 CFangleY Angle -27.00 HEADING 45.00
tiltCompensatedHeading 32.15 kalmanX 1.62 kalmanY -1.61 |
Loop Time | 0.05| ACCX Angle 45.00 ACCY Angle -45.00          GRYX Angle 0.03 GYRY Angle 0.03
GRZ Angle -0.16          CFangleX Angle 37.81 CFangleY Angle -37.79 HEADING 45.00
tiltCompensatedHeading 32.15 kalmanX 9.57 kalmanY -9.52 |
Loop Time | 0.05| ACCX Angle 45.00 ACCY Angle -45.00          GRYX Angle 0.04 GYRY Angle 0.06
GRZ Angle -0.29          CFangleX Angle 42.13 CFangleY Angle -42.10 HEADING 45.00
tiltCompensatedHeading 32.15 kalmanX 19.07 kalmanY -19.00
```

Figura 23: Lectura IMU

Prueba de módulos combinados

Luego de probar cada módulo por separado, se conectaron todas las partes de forma tal de simular el prototipo final. Se conectó el espejo al puente H y este a su vez a los puertos GPIO correspondientes. En el primer programa de prueba surgió el problema de que los puertos GPIO no estaban mapeados correctamente. Se llegó a esa conclusión desconectando la RPi y probando mover los espejos alimentando el puente H con una fuente externa. Una vez mapeados los puertos correctamente, se logró mover al espejo en los cuatro sentidos y al mismo tiempo ver como la IMU registraba tales movimientos.

10.2. Software

10.2.1. Diagrama de estados y flujogramas

Para poder posicionar los espejos retrovisores, es necesario conocer la posición de los ojos del conductor. Cabe aclarar que para ello, este debe adoptar la posición con la cual uno se desenvuelve normalmente.

Es por ello que se presentarán diversos enfoques dedicados a la obtención del eje focal del usuario.

La estructura general propuesta se resume en el siguiente diagrama

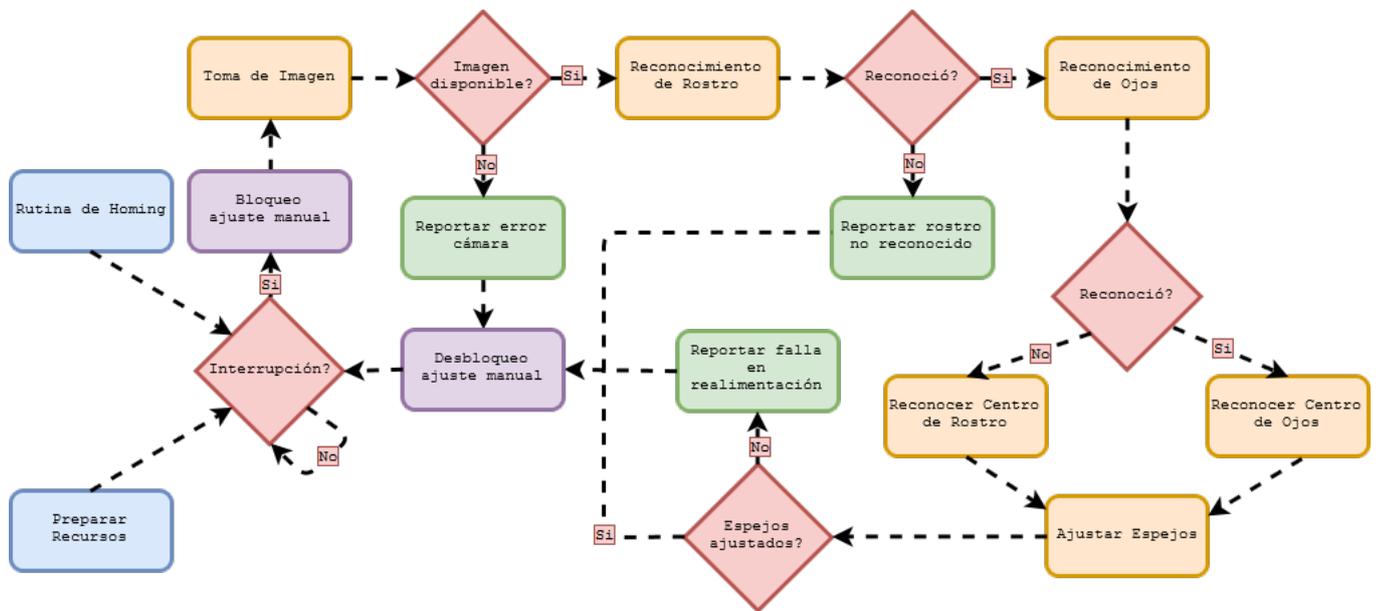


Figura 24: Diagrama de estados de software

El diagrama fue dividido por colores para un mejor entendimiento de cada etapa.

- Azul: Estas etapas son consideradas separadas al sistema de control. Preparar recursos refiere a inicializar variables, leer la configuración anterior a apagar el sistema, cargar el reconocedor de rostros, etc.; que son detalles necesarios pero que se realizan solo una vez al alimentar el dispositivo. La rutina de homing, si bien puede considerarse como parte del control, no es una etapa que se realiza siempre que un nuevo ciclo de control sea requerido. El homing se realiza esporádicamente como calibración y por ello no entra en el flujo cíclico del diagrama. Vale aclarar que aunque no entre en el flujo cíclico, puede solicitarse en cualquier momento que no el sistema no se encuentre intentando posicionar los espejos y el auto se encuentre en marcha.
- Rojo: Etapas que determinan la dirección en el flujo del sistema.
- Violeta: Son bloques que determinan el inicio y fin de la cadena de control.
- Verde: Son la interfaz con el usuario. Cada rectángulo tiene asociado un mensaje único para el usuario.
- Amarillo: Son las etapas que efectúa el programa.

El sistema inicia al ser energizado, preparando los recursos. Las acciones son determinadas luego por el usuario al accionar la interrupción, donde dichas acciones pueden ser el homing o el reacomodamiento de los espejos.

Previo a iniciar el ciclo y finalizado el mismo, el usuario puede optar por reacomodar los espejos de forma manual.

10.2.2. Análisis de complejidad

Previo a adentrarse en el detalle del software desarrollado, se esquematiza aquí una vista general de cómo funciona el sistema desde el punto más genérico y práctico, hasta el paso más interno.

En primer lugar, la secuencia comienza cuando una fotografía del conductor es tomada. En los pasos subsiguientes, se intentará posicionar los espejos en función del centro de ambos ojos.

En caso haber reconocido al conductor pero no ambos ojos, se posicionarán respecto del centro del rostro dado que ese punto guarda una simetría similar al punto que se desea detectar.

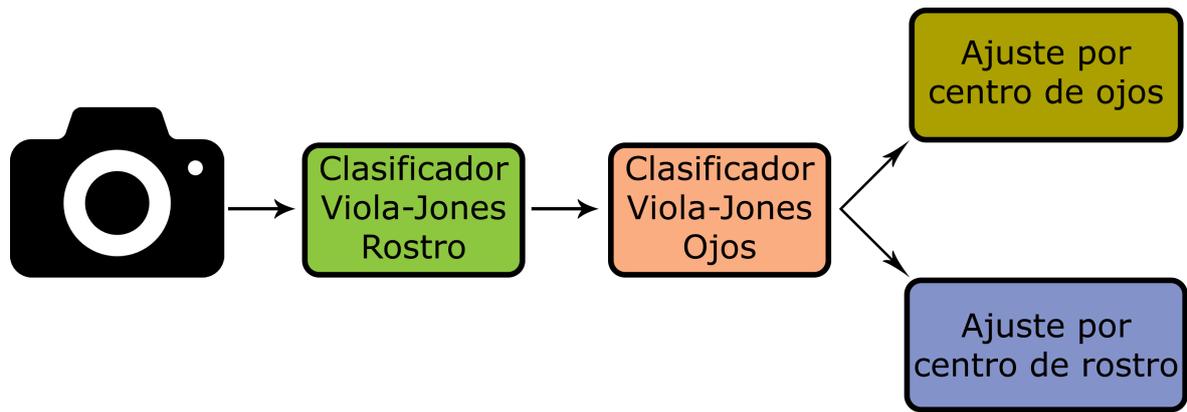


Figura 25: Plano general de software

Estos clasificadores son internos al programa. Es decir, ya fueron creados y entrenados.

El método por el cual fueron entrenados, es un método llamado Viola-Jones que se basa en la utilización de AdaptiveBoosting (será explicado con mayor detalle en las siguientes secciones).

Básicamente, se genera un clasificador en base a un conjunto preclasificado de rostros y no rostros.

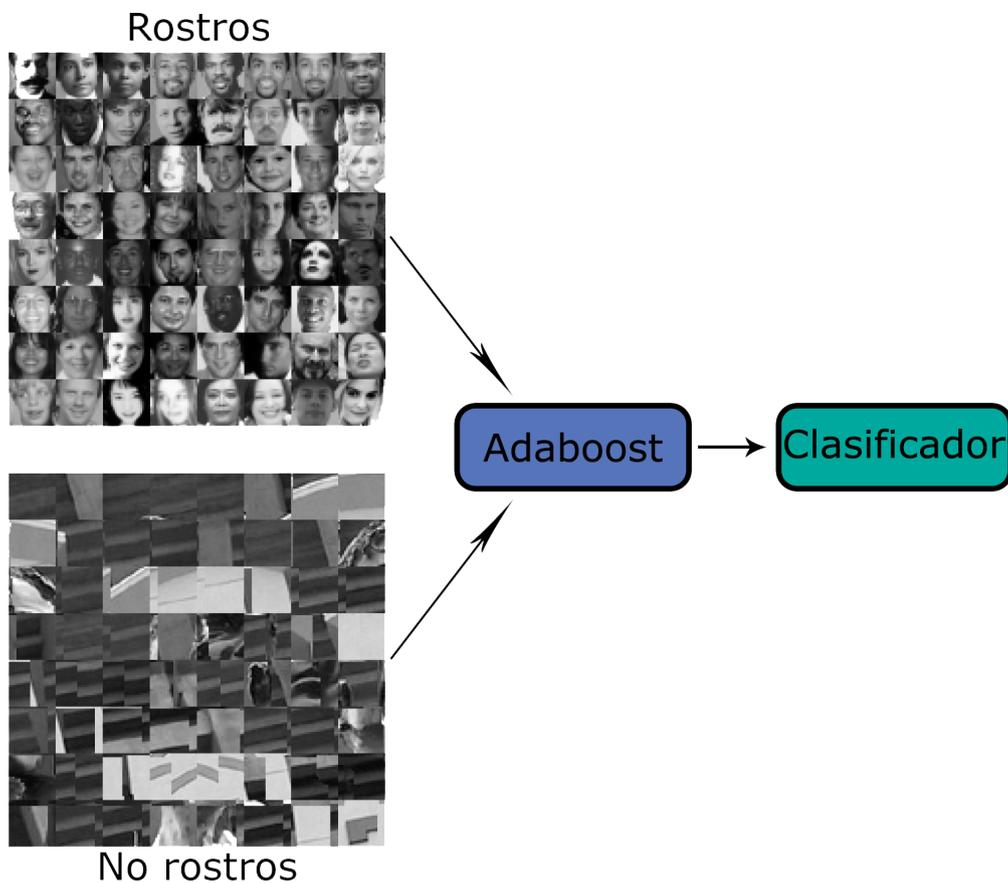


Figura 26: Método Viola-Jones

Ahora bien, la idea de utilizar AdaptiveBoosting es la de generar un comité de *microclasificadores* donde la decisión final será tomada en base al voto mayoritario de este comité.

Estos clasificadores son considerados débiles en el sentido de la falta de robustez que presenta cada uno por separado. Es decir, el voto del comité resulta eficiente pese a que los votos individuales de los miniclasificadores resultan poco confiables.

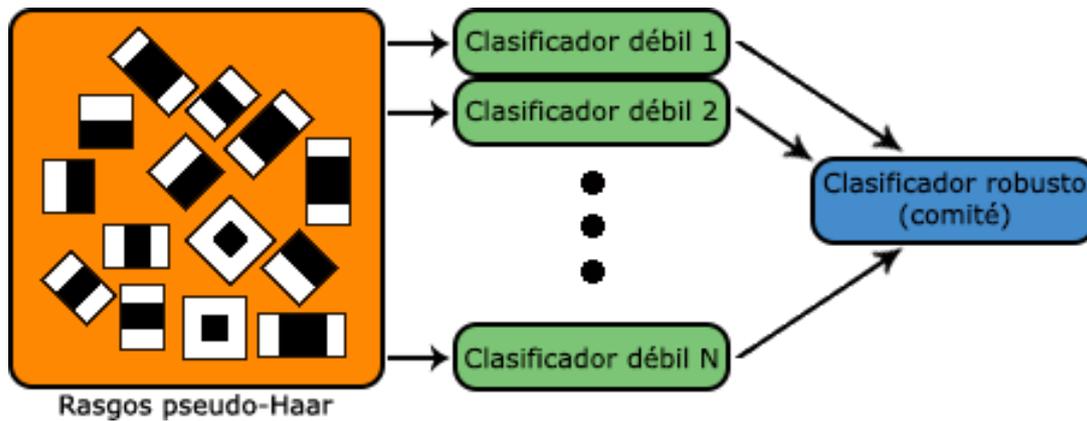


Figura 27: Adaptive Boosting

En la figura anterior, los rasgos pseudo-haar son utilizados para filtrar aquellos detalles con los cuales se desea entrenar a los clasificadores débiles.

Un comentario que no debe ser dejado de lado es que el método de diagnóstico del proyecto consta de únicamente un LED, cuyo modo de iluminación (encendido para indicar que se están moviendo los motores, parpadeando para indicar que no se reconoció el rostro, o apagado para indicar que el sistema no se encuentra en operación) determinará para el usuario de forma absoluta en qué estado se encuentra el sistema.

10.2.2.1. Reconocimiento del rostro del conductor En el siguiente apartado se detalla el funcionamiento de la parte principal de software del proyecto y las bases que fundamentan la correcta utilización de este tipo de algoritmos para la situación en particular en la que este informe se aboca.[1]

10.2.2.2. Motivación El algoritmo de Viola-Jones, aquí utilizado, es el primer sistema de detección de rasgos en tiempo real.

Tres subsistemas componen la detección rápida y eficaz: La imagen a ser analizada, *Adaboost* para la selección de facciones y un método en cascada de detección para una asignación de recursos computacionales eficientes.

Una de las principales ventajas de este método es su versatilidad al tipo de imágenes que permite ser aplicado sin importar el color de fotografía, implicando entonces independencia a la luminosidad hasta el punto de la pérdida de información.

Siendo que Viola-Jones típicamente devuelve múltiples detecciones, un paso final de pos-procesamiento es utilizado para reducir el número de redundancias.

Introducción Un detector de rostros, así como uno de ojos en particular, debe ser capaz de detectar aquel rasgo deseable en una imagen de tamaño y color arbitrario. Un sistema natural para considerar este problema es aquel de clasificación binaria, en el cual un clasificador es construido para minimizar el riesgo de error de clasificación. Siendo que ninguna distribución objetiva puede describir la probabilidad a priori de que una imagen posea dicho rasgo, el algoritmo debe minimizar tanto el riesgo de falso negativo como de falso positivo para alcanzar un rendimiento *acceptable*.

Esta tarea requiere una descripción numérica precisa de aquello que separa el rasgo a detectar de otros objetos. Resulta que estas características pueden ser extraídas con un algoritmo de aprendizaje basado en votación llamado *Adaboost*, el cual se basa en un conjunto de clasificadores débiles para formar otro, robusto, a través de un mecanismo de voto. Un clasificador es débil si, en general, no puede predefinir el objetivo a clasificar en términos de error.

Un algoritmo operacional debe además trabajar con un costo computacional razonable. Algunas técnicas como integración de imágenes y cascada supervisada hacen que Viola-Jones sea un algoritmo de alta eficacia.

10.2.2.3. Algoritmo Para estudiar el algoritmo en detalle, se comenzará con los rasgos de la imagen para la tarea de clasificación.

10.2.2.3.1. Rasgos e imagen integral El algoritmo de Viola-Jones utiliza cualidades pseudo-Haar, esto es, un producto escalar entre la imagen y alguna plantilla pseudo-Haar. Más preciso, llamando I a la imagen y P a un patrón, ambos de la misma dimensión $N \times N$ (ver Figura 28). La facción asociada con el patrón P y la imagen I se define como

$$\sum_{1 \leq i \leq N} \sum_{1 \leq j \leq N} I_{(i,j)} P_{blanco(i,j)} - \sum_{1 \leq i \leq N} \sum_{1 \leq j \leq N} I_{(i,j)} P_{negro(i,j)}$$

Para compensar los efectos de diferentes condiciones de luminosidad, toda la imagen debe ser normalizada previamente en base a su media y varianza. Aquellas imágenes con varianza menor que uno son las que poseen poca información de interés en primer lugar y por ende no son consideradas.

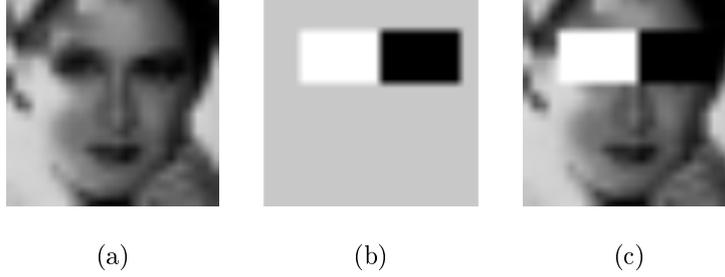


Figura 28: Rasgos pseudo-Haar. El fondo de la plantilla (b) es gris para resaltar el patrón. Solo los pixels blancos y negros son utilizados.

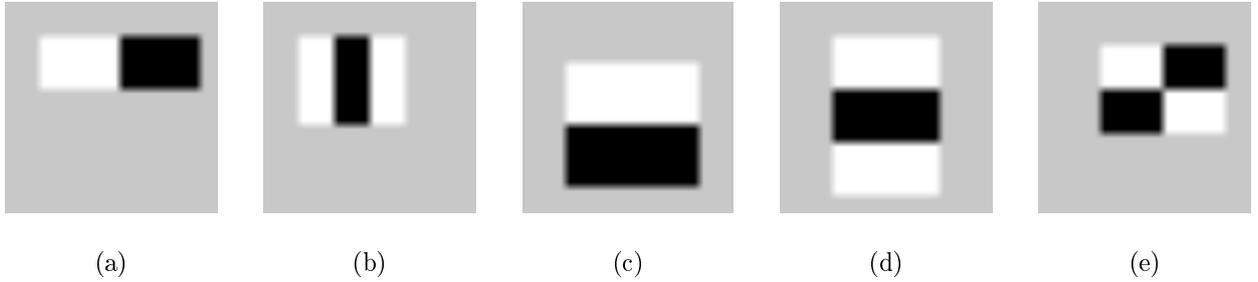


Figura 29: Cinco patrones pseudo-Haar. El tamaño y la posición de los patrones puede variar, siempre manteniendo el mismo tamaño en cuanto a sus rectángulos negros y blancos.

En practica, cinco patrones son considerados (ver Figura 29). Los caracteres derivados asumen tener toda la información necesaria para caracterizar una cara. Siendo que los rostros poseen un cierto tipo de regularidad por naturaleza, el uso de los patrones pseudo-Haar es justificado. A su vez, hay otro elemento crucial que permite que este conjunto de rasgos tengan prioridad: La imagen integral que permite calcularlos a un costo computacional bajo. En lugar de sumar todos los pixels dentro de una ventana rectangular, esta técnica imita el uso de una función de distribución acumulativa. La imagen integral II de I resulta

$$II_{(i,j)} \begin{cases} \sum_{1 \leq s \leq i} \sum_{1 \leq t \leq j} I_{(s,t)}, & 1 \leq i \leq N \cap 1 \leq j \leq N \\ 0, & \text{caso aparte} \end{cases}$$

Definido de forma tal que

$$\sum_{N_1 \leq i \leq N_2} \sum_{N_3 \leq j \leq N_4} I_{(i,j)} = II_{(N_2, N_4)} - II_{(N_2, N_3-1)} - II_{(N_1-1, N_4)} + II_{(N_1-1, N_3-1)} \quad (1)$$

para todo $N_1 \leq N_2$ y $N_3 \leq N_4$. Como resultado, computar la suma de un rectángulo de la imagen requiere como mucho cuatro operaciones elementales. Además, obtener la totalidad de la imagen puede realizarse de forma directa: Proponiendo $N_1 = N_2$ y $N_3 = N_4$ en (1), obtenemos que:

$$I_{(N_1, N_3)} = II_{(N_1, N_3)} - II_{(N_1, N_3-1)} - II_{(N_1-1, N_3)} + II_{(N_1-1, N_3-1)}$$

Llevando a una relación recursiva.

Cabe recalcar que una vez que se ha seleccionado el rasgo útil a través del Boosting Algorithm, uno debe escalarlo acorde al tamaño de la ventana. Ventanas pequeñas no serán consideradas.

10.2.2.3.2. Selección de rasgos con Adaboost

Darle un sentido a estas fracciones es tarea de *Adaboost*. Algunas terminologías. Un clasificador asigna una observación a una clasificación ponderada en un conjunto finito. Para detección de cara, asume la forma de $f : \mathbb{R}^d \rightarrow \{-1, 1\}$, donde 1 implica que hay una cara y -1 lo contrario (ver figura 30) y el superíndice d es el número de rasgos pseudo-Haar extraídos de la imagen. Dados los pesos probabilísticos $\omega_i \in \mathbb{R}_+$ asignados a un conjunto de entrenamiento hecho de n pares de clasificadores de observación (x_i, y_i) , *Adaboost* intenta de forma iterativa bajar el límite superior de la pérdida empírica bajo condiciones técnicas suaves.

$$\sum_{i=1}^n \omega_i 1_{y_i \neq f(x_i)}$$



Figura 30: Ejemplos supervisados. (a) Ejemplos positivos (b) Ejemplos negativos

Sorprendentemente, la regla de decisión construida por *Adaboost* sigue siendo razonablemente simple de modo que no es propenso a *overfitting* (tendencia a clasificar erróneamente debido a pequeñas variaciones como ruido), lo que significa que la regla aprendida empíricamente suele generalizar correctamente.

A pesar del éxito, debe ser dicho que *Adaboost* no aprende cómo un rostro debe verse ya que son los humanos, en lugar de los algoritmos, los que realizan la pre-clasificación y la primera ronda de selección de rasgos, descrito en la anterior sección.

El bloque constructor del detector de caras de Viola-Jones es de *decisión de un paso*, o un árbol de decisión de un paso, parametrizado por un rasgo $f \in \{1, \dots, d\}$, un umbral $t \in \mathbb{R}$ y un conmutador $\tau \in \{-1, 1\}$. Dada una observación $x \in \mathbb{R}^d$, una decisión de un paso h predice su etiqueta usando la siguiente regla

$$h(x) = (1_{\pi_f x \geq t} - 1_{\pi_f x < t})\tau = (1_{\pi_f x \geq t} - 1_{\pi_f x < t})1_{\tau=1} + (1_{\pi_f x < t} - 1_{\pi_f x \geq t})1_{\tau=-1} \in \{-1, 1\} \quad (2)$$

donde $\pi_f x$ es la coordenada f del vector de rasgo. Las siguientes observaciones son válidas:

1. Cualquier patrón adicional producido al permutar los rectángulos negros y blancos en un patrón existente son superfluos. Debido a que dicho rasgo es meramente el opuesto a un rasgo existente, solo un cambio de signo para t y τ es necesario para tener la misma regla de clasificación
2. Si los ejemplos de entrenamiento están ordenados en orden ascendente dado un rasgo f , una búsqueda exhaustiva sobre el umbral y el conmutador puede encontrar una decisión de un paso usando este rasgo que atañe la menor pérdida empírica en el junto de entrenamiento.

$$\sum_{i=1}^n \omega_i 1_{y_i \neq h(x_i)}, \quad (3)$$

Imagínese un umbral colocado en algún lugar en la línea real, si el conmutador se pone en 1, la regla resultante declarara un ejemplo x positivo si $\pi_f x$ es mayor al umbral y negativo de otra manera. Esto permite evaluar el error empírico de la regla, por ende seleccionando el conmutador que haga que el conjunto de datos se ajuste mejor.

Dado el margen

$$\min_{i: y_i = -1} |\pi_f x_i - t| + \min_{i: y_i = 1} |\pi_f x_i - t|,$$

y el riesgo, o la expectativa del error empírico (3), están altamente relacionados, por dos decisiones de un paso teniendo mismo riesgo empírico, aquel con margen mayor es preferible. Por ende en la ausencia de un duplicado, hay $n + 1$ posibles umbrales y aquel con la menor pérdida empírica será seleccionado. Sin embargo, es posible tener los mismos valores de rasgos de diferentes ejemplos y se debe manipular con extra cuidado.

Al ajustar individualmente los pesos de los ejemplos, *Adaboost* pone más esfuerzo en aprender ejemplos más difíciles y agregar mayor decisiones de un paso en el proceso. Intuitivamente, en la votación final, un paso h_t con menor pérdida empírica es recompensada con mayor opinión cuando un miembro T del comité (clasificadores basados en voto) asignan un ejemplo acorde a

$$f^T(\cdot) = \text{sign} \left[\sum_{t=1}^T \alpha_t h_t(\cdot) \right].$$

Como deben ser valuados los ejemplos de entrenamientos será explicado en detalle en el Apéndice A. La figura 4 muestra una instancia donde *Adaboost* reduce las tasas de falso positivo y falso negativo simultáneamente cuanto mas pasos son agregados al comité. Para simplicidad de notación, se denota la pérdida empírica como

$$\sum_{i=1}^n \omega_i(1) 1_{y_i \sum_{t=1}^T \alpha_t h_t(x_i) \leq 0} := P(f^T(X) \neq Y),$$

donde (X, Y) es una pareja aleatoria distribuida acorde a la probabilidad P definida por los pesos $\omega_i(1)$, $1 \leq i \leq n$ establecidos cuando comienza el entrenamiento. Mientras la pérdida empírica se reduce a cero con T , también las tasas de falsos positivos $P(f^T(X) = 1|Y = -1)$ y la falsos negativos $P(f^T(X) = -1|Y = -1)$ debido a

$$P(f^T(X) \neq Y) = P(Y = 1)P(f^T(X) = -1|Y = 1) + P(Y = -1)P(f^T(X) = 1|Y = -1)$$

Por ende, la tasa de detección

$$P(f^T(X) = 1|Y = 1) = 1 - P(f^T(X) = -1|Y = 1)$$

tiende a 1.

Así, el tamaño T del comité entrenado depende de las tasas de falsos positivos y falsos negativos. Adicionalmente, cabe mencionar que, dado n_- ejemplos negativos y n_+ positivos de un conjunto de entrenamiento, es costumbre dar a un ejemplo negativo un peso inicial igual a $0,5/n_-$ así *Adaboost* no favorece a ninguna categoría al inicio.

10.2.2.3.3. Cascada atencional[1] En teoría, *Adaboost* puede producir un único grupo de decisiones de un paso que generalice correctamente. Sin embargo, para lograr esto un gigantesco set de ejemplos negativos es necesario para poder discriminar todos los patrones negativos. Adicionalmente, un comité único implica que todas las ventanas dentro de una imagen tienen que pasar por el mismo proceso de decisión largo. Debe haber un método mas costo-efectivo.

La probabilidad previa de que una cara aparezca en una imagen tiene poca relevancia para la construcción clasificadora presentada porque requiere que tanto la tasa de falsos negativos como los falsos positivos empíricos se aproximen a cero. No obstante, nuestra propia experiencia nos dice que en una imagen, un limitado numero de sub-ventanas merecen mayor atención que otras.

De ahí la idea de una cascada atencional multicapa que incorpora un principio similar al de la codificación de Shannon: el algoritmo debería desplegar más recursos para trabajar en aquellas ventanas con más probabilidad de contener una cara mientras se gasta el menor esfuerzo posible en el resto.

Se espera que cada capa de la cascada atencional cumpla con un objetivo de entrenamiento expresado en tasa de falsos positivos y falsos negativos: entre n ejemplos negativos declarados positivos por todas sus capas precedentes, la capa l debería reconocer al menos $(1 - \gamma_l)n$ como negativo y mientras tanto intentar no sacrificar su rendimiento en los positivos: la tasa de detección debe mantenerse por encima de $1 - \beta_l$.

Al final, sólo cuenta el error de generalización, que desafortunadamente sólo se puede estimar con algunos ejemplos de validación que *Adaboost* no puede ver en la fase de entrenamiento. Por lo tanto, se hace una elección conservadora en cuanto a cómo se evalúan las tasas de error: la tasa de falsos positivos más alta obtenida del entrenamiento y la validación se utiliza para evaluar qué tan bien el algoritmo ha aprendido para distinguir las caras de las no-caras. La tasa de falsos negativos se evalúa de la misma manera.

Debe tenerse en cuenta que *Adaboost* por sí mismo no favorece ninguna de las dos tasas de error: Apunta a reducir ambas simultáneamente en lugar de una a expensas de la otra. Para permitir flexibilidad, un control adicional $\epsilon \in [-1, 1]$ se introduce para sesgar el clasificador

$$f_s^T(\cdot) = \text{sign} \left[\sum_{t=1}^T \alpha_t (h_t(\cdot) + s) \right], \quad (4)$$

para que un s estrictamente positivo incline el clasificador a predecir una cara y viceversa.

Para imponer una asignación eficiente de recursos, el tamaño del comité debe ser pequeño en las primeras capas y luego crecer gradualmente para que un gran número de patrones negativos fáciles puedan eliminarse con poco esfuerzo computacional (ver Figura 5).

Adjuntando una capa a la cascada significa que el algoritmo ha aprendido a rechazar algunos nuevos patrones negativos vistos anteriormente como difíciles, al mismo tiempo de mantener más o menos el mismo grupo de entrenamiento positivo. Para construir la capa siguiente, se requieren más ejemplos negativos para hacer que el proceso de formación sea significativo. Para reemplazar los negativos detectados, ejecutamos la cascada en un gran conjunto de imágenes en gris sin rostro humana y recogemos sus falsas ventanas positivas. El mismo procedimiento se utiliza para construir y reponer el conjunto de validación.

Supongamos que en la capa l , se forma un comité de clasificadores débiles T_l junto con un desplazamiento s para que el rendimiento del clasificador en el conjunto de entrenamiento y validación pueda ser medido. Denotaremos la tasa de falsos positivos y falsos negativos obtenidos por $\hat{\gamma}_l$ y $\hat{\beta}_l$. Dependiendo de su relación con los objetivos γ_l y β_l , se presentan cuatro casos:

1. Si se alcanza el objetivo de entrenamiento en capas ($\hat{\gamma}_l \leq \gamma_l$ y $\hat{\beta}_l \leq \beta_l$), el algoritmo pasa a entrenar la capa siguiente si es necesario.
2. Si hay espacio para mejorar la tasa de detección ($\hat{\gamma}_l \leq \gamma_l$ y $\hat{\beta}_l > \beta_l$), s se incrementa en u , una unidad prefijada.
3. Si hay espacio para mejorar la tasa de falsos positivos ($\hat{\gamma}_l > \gamma_l$ y $\hat{\beta}_l \leq \beta_l$), s es disminuido por u .
4. Si ambos índices de error no alcanzan el objetivo ($\hat{\gamma}_l > \gamma_l$ y $\hat{\beta}_l > \beta_l$), el algoritmo, si el comité actual no excede un límite de tamaño específico de capa prefijado, capacita a un miembro más para que agregue al comité.

Se debe prestar especial atención aquí: El algoritmo podría alternar entre los casos 2 y 3 y crear un ciclo infinito. Una solución es reducir a la mitad la unidad u cada vez que sucede hasta que u sea menor que 10^{-5} . Cuando ello sucede, se recomienda una ronda más de entrenamiento en esta capa.

Como se mencionó anteriormente, para evitar que un comité crezca demasiado grande, el algoritmo deja de refinar una capa después de que se sobrepasa un límite de tamaño dependiente de la capa. En este caso, el factor s se establece en el valor más pequeño que satisface el requisito de falso negativo. Entonces, un caso de aprendizaje más difícil es traspasado a la siguiente capa. Esta estrategia funciona porque la incapacidad de *Adaboost* para cumplir con el objetivo de entrenamiento a menudo puede explicarse por el hecho de que un clasificador entrenado en un número limitado de ejemplos podría no generalizar bien en el conjunto de validación.

Para analizar cuan bien actúa la cascada, asumamos que en la capa l , *Adaboost* puede entregar un clasificador $f_{l,s_l}^{T_1}$ con falsos positivos γ_l y tasa de detección $1 - \beta_l$. En términos probabilísticos, significa que

$$P(f_{l,s_l}^{T_1}(X) = 1 | Y = 1) \geq 1 - \beta_l \text{ y } P(f_{l,s_l}^{T_1}(X) = 1 | Y = -1) \leq \gamma_l,$$

donde por abuso de notación P denota la probabilidad en el espacio de imagen.

Si el algoritmo frena luego de L iteraciones, la regla de decisión queda como

$$f_{cascada}(X) = 2 \left(\prod_{l=1}^L 1_{f_{l,s_l}^{T_1}(X)=1} - \frac{1}{2} \right)$$

Entonces una ventana es declarada positiva si y solo si todos sus componentes de la capa clasifican del mismo modo

$$\begin{aligned} P(f_{cascada}(X) = 1 | Y = -1) &= P\left(\bigcap_{l=1}^L \{f_{l,s_l}^{T_1}(X) = 1\} | Y = -1\right) \\ &= P\left(f_{L,s_L}^{T_L}(X) = 1 \mid \bigcap_{l=1}^{L-1} \{f_{l,s_l}^{T_1}(X) = 1\} \wedge Y = -1\right) P\left(\bigcap_{l=1}^{L-1} \{f_{l,s_l}^{T_1}(X) = 1\} | Y = -1\right) \\ &\leq \gamma_L P\left(\bigcap_{l=1}^{L-1} \{f_{l,s_l}^{T_1}(X) = 1\} | Y = -1\right) \end{aligned}$$

$$\leq \gamma_l^L$$

Del mismo modo, la tasa de detección global se puede estimar como sigue

$$\begin{aligned} P(f_{cascada}(X) = 1 | Y = 1) &= P\left(\bigcap_{l=1}^L \{f_{l,s_l}^{T_1}(X) = 1\} | Y = 1\right) \\ &= P\left(f_{L,s_L}^{T_L}(X) = 1 | \bigcap_{l=1}^{L-1} \{f_{l,s_l}^{T_1}(X) = 1\} \wedge Y = 1\right) P\left(\bigcap_{l=1}^{L-1} \{f_{l,s_l}^{T_1}(X) = 1\} | Y = 1\right) \\ &\geq (1 - \beta_l) P\left(\bigcap_{l=1}^{L-1} \{f_{l,s_l}^{T_1}(X) = 1\} | Y = 1\right) \\ &\geq (1 - \beta_l)^L \end{aligned}$$

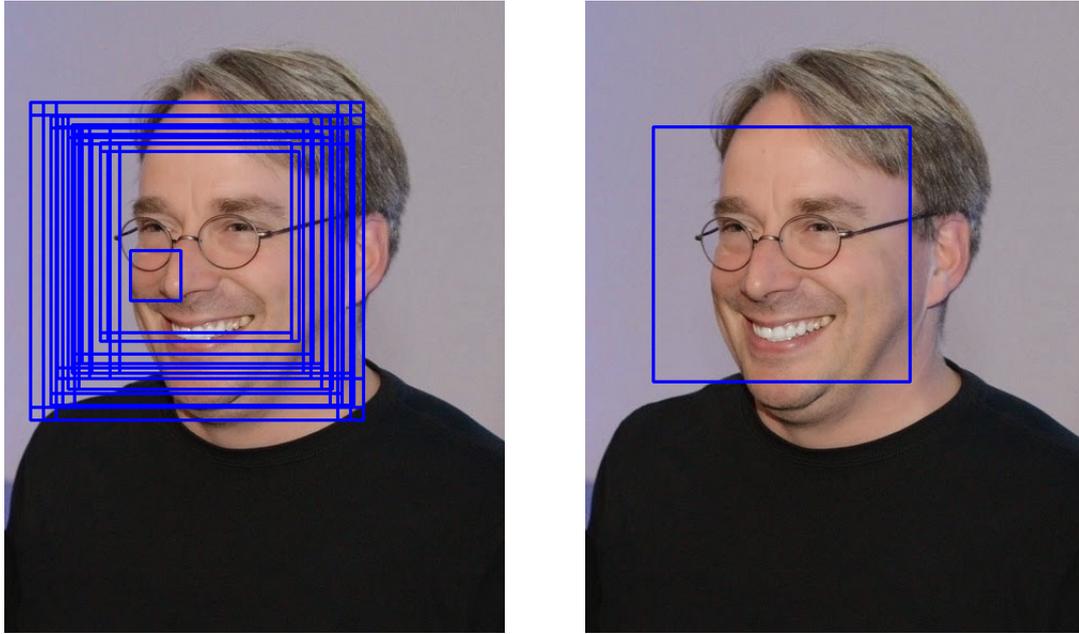
En otras palabras, si las tasas empíricamente obtenidas son alguna indicación, una ventana sin rostro tendrá una probabilidad mayor que $1 - \gamma_l$ para ser etiquetada como tal en cada capa, lo que dirige efectivamente la atención del clasificador en cascada sobre aquellos que tienen más probabilidades de tener un rostro.

10.2.2.3.4. Experimentos y datos

1. Un gran número de imágenes son necesarias para poder obtener un clasificador robusto. Teóricamente debe poder resolver rostros de diversas posiciones; para el caso particular al que trata este proyecto solo fotos de frente o 3/4 fueron necesarias.
2. La primera vez que se desea entrenar el algoritmo toma demasiado tiempo lo cual es coherente teniendo en cuenta de qué trata el proceso descrito anteriormente: Para una cascada de 31 capas puede llegar a tardar 24hs, donde 1000 imágenes son de entrenamiento y 1000 de validación. Toma tanto tiempo por la necesidad de validar una gran cantidad de falsos positivos. Además, las capas posteriores requieren un entrenamiento mas intenso siendo que son la última barrera de discernimiento.
3. Valores razonables para las tasas de falsos positivos y negativos suelen ser de 0.5 y 0.995 respectivamente.

10.2.2.3.5. Pos-Procesamiento La figura 31 muestra que la misma cara puede ser detectada múltiples veces por un clasificador correctamente entrenado. Esto se debe a que los ejemplos positivos permiten una cierta flexibilidad en la pose y la expresión. Por el contrario, muchos falsos positivos no gozan de esta estabilidad, a pesar de que fuera de contexto algunos de ellos parecen una cara. Esta observación apoya las siguientes *detecciones de confianza* basadas en la heurísticas para reducir aún más los falsos positivos y *limpiar* el resultado detectado:

1. Una ventana detectada contiene una cara si y sólo si un número suficiente de otras ventanas detectadas adyacentes lo confirma. En esta implementación, el multiplicador de tamaño de ventana es 1:5. Se supone que el número de ventanas de prueba que detectan una cara crece linealmente con su tamaño.
2. Es posible que las ventanas restantes detectadas se superpongan después de la prueba anterior. En este caso, distinguimos dos escenarios:
 - (a) Si el centro de la ventana más pequeña está fuera de la más grande, mantener ambas.
 - (b) Mantenga el que tenga mayor confianza en la detección de lo contrario.



(a)

(b)

Figura 31: Diferente desempeño por cantidad de capas

10.2.2.3.6. Fase final Un último paso de procesamiento es necesario para poder determinar fehacientemente la posición del conductor, donde todo lo dicho anteriormente es igual de válido.

Detección de ojos. El único propósito de la detección de caras, además de contextualizar las bases que fundamentan el algoritmo utilizado, es la de reducir el espacio de búsqueda de los ojos.

Una alternativa a este paso podría haber sido la de binarizar la imagen, sin embargo al aplicar ello se pierde resolución y el detector puede llegar a fallar.

Fue analizado también que la posición de la cámara en el parante del conductor permite la detección de ambos ojos del sujeto, prediciendo un código más certero. Cabe destacar que sin reducir el espacio de búsqueda, los ojos de otras personas dentro del automóvil influyen y hacen más lenta la correcta utilización del sistema.

Con un nuevo set de imágenes se entrenó un nuevo detector cuyo propósito sea el de identificar ojos, sin importar si es derecho o izquierdo, del mismo modo que fue hecho para detectar caras.

Los resultados fueron similares a lo ocurrido sin el pos-procesamiento de rostros detectando una gran cantidad de falsos positivos, por lo que mayor refinamiento fue impuesto.

En la figura 32 puede apreciarse el resultado final de aplicar el refinamiento a la imagen utilizada anteriormente.

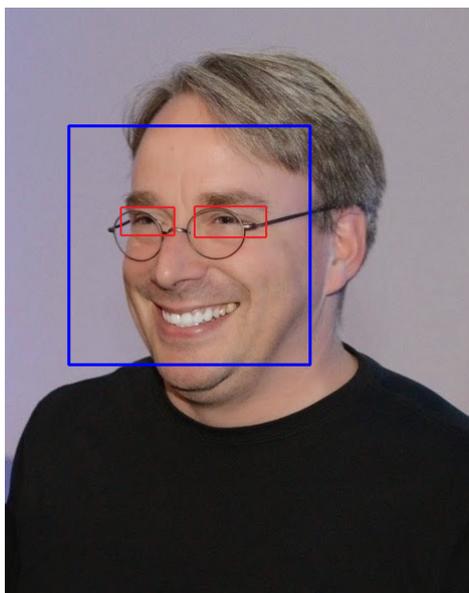


Figura 32: Detección de ojos

Cabe apreciar que sin importar que el sujeto use anteojos y por ellos presente una leve interferencia en la zona a detectar, Viola-Jones es capaz de ignorarlo. Eso es debido a una correcta selección de imágenes a entrenar y al hecho de no alcanzar un estado de *overfitting* (explicado anteriormente).

Centro de ojos. Finalmente queda el paso más sencillo de todos. Una vez que se consigue distinguir el par de ojos con sus respectivas coordenadas, el centro es calculado como la media horizontal y vertical de los recuadros.

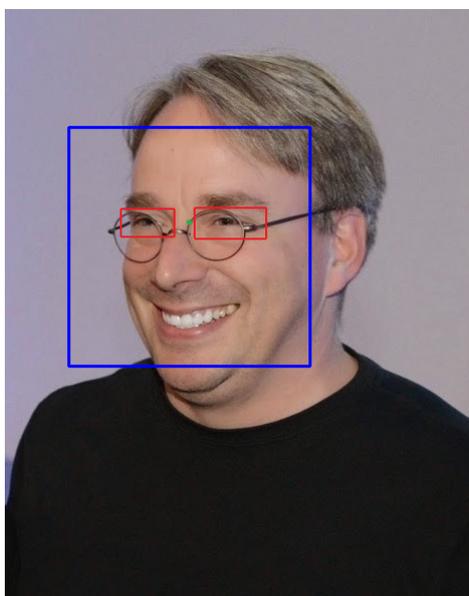


Figura 33: Resultado final

El punto verde en la figura 33 representa dónde fue calculado el centro.

Puede verse que el punto se encuentra desplazado respecto del centro en el eje horizontal. Este error, si bien pequeño, no merece la pena ser abordado para ser corregido ya que al ser una constante en todas las imágenes, cuando se requiera calcular un cambio de posición (equivalente a una derivada) la constante desaparece.

Finalmente solo resta hablar de los aspectos notorios al usuario como son los tiempos de procesamiento, versatilidad y robustez.

10.2.2.4. Tiempos. El aspecto que consume más tiempo de todo el sistema es el de cargar los detectores e inicializar los módulos. Ello puede demorar hasta 3 segundos, sin embargo esto puede ser enmascarada si se realiza por única vez en el encendido del auto mientras se ejecuta una rutina de *homing*.

Por otro lado, para una cámara web genérica con una resolución 640x480 el tiempo de detección del rostro fue de 500ms y la detección de los ojos de 200ms, utilizando una *Raspberry Pi* modelo 2B con una frecuencia de procesamiento de 800MHz. Tales aspectos resultan cómodos para el usuario e imperceptibles.

10.2.2.5. Versatilidad. Lo único que limita al proyecto de ser universalmente trasladable es la diferencia de tamaños en los vehículos. Sin embargo, al aplicar una calibración inicial por única vez al momento de la instalación el sistema queda totalmente funcional sin importar el modelo del auto.

La calibración consiste en encontrar la posición central del espejo mediante una serie de movimientos que el control realiza automáticamente.

Esto se realiza una primera vez para que el sistema posea un punto de partida para futuros reajustes.

Para poder realizar estos últimos mencionados, el conductor debe accionar el botón para dar inicio al programa, colocado cercano a la botonera de los espejos y luego el usuario poseerá 3 segundos para acomodarse como desee y el programa indicará que comenzó a funcionar, mediante un led.

10.2.2.6. Robustez. Pese a que el algoritmo de Viola-Jones es reconocido como uno de los más eficaces, su rendimiento depende fuertemente de como es entrenado y de sus situaciones de uso. Es decir, si el algoritmo es entrenado con imágenes de alta resolución mientras que es utilizado con imágenes ruidosas y pobres (o caso inverso), es esperable que su tasa de aciertos disminuya.

Por ello, el conjunto de imágenes utilizado para entrenar debe presentar las características de uso cotidiano. En otras palabras, las fotográficas utilizadas en la etapa de entrenamiento poseen diferentes tonos de luminosidad, diversos ángulo, diversas resoluciones, etc.

A su vez, la confiabilidad del proyecto puede verse en capas: Si la detección de ojos falla, el centro es considerado como el centro del rostro. Si la detección de rostros falla, una nueva fotografía es solicitada al informarle al usuario, informándole mediante la interfaz de operación (precisamente, mediante el parpadeo por 3 segundos de un led) que el reconocimiento ha fallado. Al usuario siempre se le permite modificar la posición recomendada en cualquier instancia de éxito y error.

10.2.2.7. Conclusión Mediante lo expuesto anteriormente, la imagen puede ser clasificada y por ende se le pueden determinar la posición de los ojos, respecto al centro de la cámara en el plano paralelo a la misma.

Sin embargo, el vector de corrección de los espejos también requiere de la profundidad de la persona.

A continuación se verán diversos enfoques analizados al tomar en cuenta lo recién mencionado.

10.2.2.8. Identificación de la profundidad

Método por posición del asiento

Este ha sido el primer punto tomado en cuenta al darse cuenta del factor de profundidad.

La alternativa consistía en, de alguna manera, detectar el espacio en la corredera del asiento donde se encontrara posicionado el mismo. De este modo, el cálculo del vector resultaba muy sencillo y hasta podría acomodarse uno de los ejes de los espejos previo a la detección facial.

Sin embargo, este fue rápidamente descartado debido a la poca robustez y simplicidad.

La profundidad se dividía en espacio discretos y no se tomaba en cuenta que no todas las personas conducen con la misma inclinación. Incluso considerando la inclinación del respaldo, las personas no siempre conducen totalmente apoyadas sobre este. A su vez, la geometría de la cabeza tampoco era tenida en cuenta.

Si bien este último parecería de poca importancia, conlleva a que el diseño sea más precario de lo requerido por los clientes.

Diversas fuentes de luz

Podría considerarse este enfoque como el más creativo de los propuestos.

Consistía en la aplicación, no simultánea de 2 fuentes de luz, las cuales corresponderían a 2 fotos distintas.

Si de alguna manera se pudiera obviar el fondo de la imagen y limitarse solo al sector de la cabeza (o mismo considerar el fondo para correlacionar ambas fotografías y tomarlo como base) con las proyecciones de las sombras se podría estimar que tan inclinada se encuentra la persona y así calcular la profundidad.

De esto surgían algunas cuestiones:

- Si una de las fuentes era obstruida, la sombra generada por el conductor no serviría.
- Cómo modificaba a cada una de las fotografías la luz natural? Esto se traduce a que dependiendo de como entre la luz externa al sistema, la sombra producida sería distinta y por ende se concluiría en falsas estimaciones.

Método combinado

Surge el nombre debido a que es la conformación de 2 enfoques: Simetría facial (que será explicado a continuación), para detectar diferencias de altura, y sensor de profundidad.

Este método, como ya fue mencionado, posee varias complicaciones a la hora de instalar un sensor de profundidad cuya métrica sea confiable en todo momento. Tomando por ejemplo un sensor IR, las complicaciones que traía eran encontrar un punto que no sea obstaculizado (por lo que no podría ser colocado detrás del volante), encontrar un punto para sensar siempre la cabeza del conductor (debido a diferencias de altura y profundidad, ello era algo bastante complicado de lograr; además sensar otro punto, como el abdomen, daba demasiada incerteza ya que el usuario puede conducir reclinado o encorvado), entre otras.

Simetría facial

Este modelo partió siendo una técnica de reconocimiento de altura y debía ser acompañado de un subsistema que permitiese medir profundidad (como fue mencionado anteriormente).

Si bien es sabido que los ojos no guardan exactamente igual simetría respecto al centro de la cara, la posición de ellos brindan una clara semejanza al centro de esta.

- Verticalmente, los ojos se encuentran aproximadamente en el centro del rostro.
- Horizontalmente, el punto medio entre ambos coincide con el centro del rostro.

Una vez detectada la posición de estos, se procede a estimar el punto medio entre los mismos.

Ahora bien, experimentalmente (sección 9.1.3.5) se ha descubierto que no es necesario incorporar un sensor de profundidad. Las razones que avalan dicha experimentación son las siguientes:

- En primer lugar, la posición del conductor es tridimensional. Sin embargo y para el alcance de este proyecto, se reduce el espacio a dos dimensiones ya que el conductor solo puede desplazarse en profundidad al volante y variar en altura. Ligeras variaciones en la dirección reducida pueden llegar a producir un corrimiento en pixels pequeño, lo cual es un error aceptable.
- La imagen de una cámara es bidimensional.
- Teniendo en cuenta estas dos premisas, el sistema de ecuaciones que permite resolver el cambio de perspectiva por tener la cámara desplazada y ligeramente rotada respecto al centro del conductor queda determinado, ya que se intentaría llevar un espacio 2-D a otro 2-D.

Una forma de poder determinar este cambio de perspectiva es mediante un modelo pinhole[19], sin embargo el mismo no fue llevado a cabo para poder dedicarle mayor tiempo a otros aspectos del trabajo (como la experimentación).

Con este argumento y mediante los cálculos propuestos en la sección 9.1.2 se puede estimar, con precisión dependiente de la densidad de pixels de la cámara, la distancia a la cual se haya el conductor.

Este es el método implementado en el proyecto.

10.2.3. Descripción de subrutinas

10.2.3.1. Toma de la imagen Es el primer paso en toda la cadena de control de los espejos laterales.

Por defecto, el sistema se encuentra siempre en una condición ideal, de bajo consumo, esperando a ser accionado por el usuario.

Al momento que el usuario decide ajustar los espejos, este acciona un botón que interrumpe el estado de bajo consumo del proyecto y el proceso comienza tomando una imagen del conductor.

10.2.3.2. Reconocimiento de rostro Inmediatamente después de tomar la imagen, se procede a intentar detectar el rostro del conductor con el procedimiento mencionado anteriormente mediante la librería de OpenCV de Python.

Este no es un paso propiamente dicho del proyecto debido a que el mismo podría ser obviado con un resultado final igual. El único motivo por el que es incluido es el de poder acortar los tiempos de la detección de ojos y por ende el de respuesta del sistema.

Es importante destacar que por como fue concebido el código, el detector solo reconocerá el rostro del conductor obviando al resto de los pasajeros, bajo condiciones de conducción normales.

10.2.3.3. Reconocimiento de ojos Una vez reducido el espacio de búsqueda gracias a la detección de rostros se procede a detectar los ojos bajo un procedimiento similar al anterior.

Este paso es el más crítico ya que define el grado de precisión del producto. Por ello, fue el paso con el que más cuidado se tuvo, tanto en velocidad de respuesta como en robustez.

Dado que en la imagen siempre se encuentra al menos uno de los ojos, el punto de referencia fue determinado como el centro entre ambos ojos, o la posición del que fue encontrado.

Con el par de ojos detectados, y esto es algo que se destaca: Ambos ojos siempre están presentes, se procede a calcular el centro de la cara como la semi diferencia de la posición de estos.

Si por algunas razón ningún ojo o solamente uno es detectado, la referencia se toma como el centro del rostro encontrado.

10.2.3.4. Cálculo de diferencia de posición respecto una referencia Tres entradas son necesarias para este paso: Posición de calibración, posición nueva y distancia al espejo.

Al momento de calibrar el sistema, la distancia al espejo es calculada de forma automática. En este paso, dos de las tres entradas son resueltas. Una vez obtenida la posición del conductor a partir del inciso anterior, se procede a calcular la rotación que debe realizar el espejo mediante ley de reflexiones ópticas.

10.2.3.5. Lazo de control del espejo Cuando la rotación es calculada, los motores comienzan a moverse, de a uno por vez. El control se establece con las IMUs con la resolución que es impuesta ($0.007^\circ/s$).

El único parámetro de control es la rotación del espejo sensada por la IMU y el movimiento cesa cuando se detecta que se rotó lo necesario.

Siendo que los motores giran a una velocidad considerablemente lenta ($60^\circ/8s$ medida), la tasa de muestreo de la IMU es alta (0.05 segundos) y no intereza regular la velocidad de rotación, el lazo de control fue minimizado al tipo encendido-apagado[15]. Esto quiere decir que el movimiento es regulado por dos estados absolutos: Moverse y frenar.

Al momento que la IMU sensa que el espejo rotó lo que debería, se le ordena a este último que frene. En sucesivas pruebas, no se apreció que el motor tuviese una inercia que hiciese que el espejo continuara moviéndose un poco más.

10.2.4. Plan de prueba de módulos y de depuración de Software

10.2.4.1. Detector de rostros

10.2.4.1.1. Objetivo Determinar el correcto código de detección de rostros, para poder acotar el recinto de búsqueda de ojos y de esta manera acortar tiempos e incrementar la tasa de aciertos.

10.2.4.1.2. Pre-requisitos Librería OpenCV instalada en el microcontrolador a utilizar.

Haber conseguido tomar una fotografía con una cámara web conectada a la placa de desarrollo.

10.2.4.1.3. Entrada Conjunto de fotografías en variadas condiciones. Es decir, distintos niveles de luminosidad, distintas alturas del conductor, distintas profundidades del conductor.

10.2.4.1.4. Procedimiento La etapa de reconocimiento de rostro fue dividida en 2 opciones: Utilizar el método en cascada *Haar* (descrito anteriormente) o el método en cascada LBP. Éste último, es una simplificación del método de Haar, siendo su velocidad de procesamiento mayor a costas de bajar la precisión.

El motivo de ello fue poder determinar el algoritmo que mejor tasa de eficacia tenga en función del tiempo que demore converger a una solución.

La tarea consistió en suministrar al código de 15 imágenes de sujetos en situaciones de conducción, dejando el algoritmo finalizar con acierto o con resultado vacío.

Los resultados se aprecian en la siguiente tabla.

Método	Tiempo medio hasta encontrar el rostro (de ser encontrado)	Tasa de aciertos	Tasa de fallos
LBP	25 segundos	73.34 %	26.67 %
Haar	1 minutos	100 %	0 %

Cuadro 28: Rostros: LBP vs Haar

10.2.4.1.5. Conclusión

- Si bien el método LBP resultó mucho más rápido que *Haar*, su tasa de fallos resultó más elevada que el último.
- A pesar de tomar imágenes donde más de un sujeto se encontraban en la toma, el código resultó lo suficientemente robusto como para detectar a aquel que fuese más predominante en la imagen.
- Pese a la robustez de *Haar*, el tiempo de convergencia podía demorar algunos minutos en contados escenarios. Se analizó que dicho efecto se debía a situaciones donde las condiciones lumínicas no eran óptimas o el contraste entre el sujeto y el ambiente no eran muy marcados. Ello es algo que no había sido tenido en cuenta.

Para poder solventar dicha situación las siguientes condiciones fueron tomadas:

1. Acotar las dimensiones del objeto a detectar.
2. Sobre el criterio del número de detecciones en una vecindad, iterar de más estricto a menos estricto. En el inciso 10.2.3.3.5 se describe este criterio, dónde más estricto conlleva a un mayor número de detecciones y menos estricto a detectar solo un rostro.
3. Acotar el factor de re-escalamiento de la imagen e iterar desde un valor más grande (que permita una mayor velocidad de procesamiento) a un valor más pequeño (que reduzca la velocidad pero tenga mayor tasa de detección).

El resultado de dichas modificaciones conllevó a un tiempo medio de encuentro de 1 segundos para *Haar* y 0.3 segundos para LBP. Además, siendo que las acotaciones no fueron lo suficientemente estrictas, la tasa de aciertos y errores se mantuvo igual.

10.2.4.2. Detector de ojos

10.2.4.2.1. Objetivo Determinar el correcto código de detección de ojos que garantice robustez y rapidez.

10.2.4.2.2. Pre-requisitos Librería OpenCV instalada en el microcontrolador a utilizar. Código detector de rostros funcionando.

10.2.4.2.3. Entrada Conjunto de fotografías de rostros en variadas condiciones. Es decir, distintos niveles de luminosidad, distintas perspectivas y distintos tamaños.

10.2.4.2.4. Procedimiento El único objetivo del detector de rostros es reducir el espacio de búsqueda del detector de ojos.

Una vez pre-procesadas las imágenes crudas, el detector de ojos recibe un rostro de dimensiones inciertas (pero acotadas por un máximo como fue mencionado) y aplica el mismo criterio en cascada con cualidades *Haar* y LBP para encontrar los dos ojos del conductor.

Sin importar la posición del conductor, al haber colocado la cámara de forma tal que su plano abarque todo el recorrido del asiento de este, los dos ojos siempre se encontrarán presentes en la imagen.

Una vez más, se demuestra la comparación entre LBP y *Haar*. A su vez, se verificó la posición en pixels donde el código encontró el centro de la cara respecto del verdadero centro, chequeado manualmente con un visualizador de imágenes.

Método	Tiempo medio hasta encontrar los ojos (de ser encontrados)	Tasa de aciertos	Tasa de fallos	Diferencia en pixels
LBP	12 segundos	33.34 %	66.67 %	6 pixels
Haar	2 minutos	86.67 %	13.34 %	2 pixels

Cuadro 29: Ojos: LBP vs Haar

10.2.4.2.5. Conclusión Pese a haber acotado el recinto de búsqueda a través del detector de rostros, el detector de ojos resultó significativamente más lento que este último (4 minutos en el peor de los casos).

Las mismas condiciones que para el inciso anterior fueron consideradas en este, resultando en que el método *Haar* tuviese un tiempo medio de convergencia de 12 segundos sin reducir su tasa de aciertos.

Siendo que el método de *Haar*, en términos de tiempos de resolución, se encuentra ahora al mismo nivel que LBP (el cual es suficiente para satisfacer los requerimientos del cliente) y su tasa de aciertos es cercana a la ideal, se determinó que el método a utilizar debe ser *Haar*.

10.2.4.3. Medición de movimiento

10.2.4.3.1. Objetivo Poder determinar aspectos y etapas que controlen de forma confiable la inclinación del espejo.

10.2.4.3.2. Pre-requisitos Tener lectura del giróscopo y acelerómetro de la IMU.

10.2.4.3.3. Entrada Valores del giróscopo y del acelerómetro.

10.2.4.3.4. Procedimiento Se encendió la IMU y se comenzó a tomar muestras de forma continua hasta obtener un estado estacionario (las primeras muestras tomadas durante los primeros 2 segundos son poco fiables por falta de referencia).

Siendo que el objetivo es poder medir diferencias de ángulos de forma confiable, se implementó un filtro de mediana para compensar el drift del giróscopo.

A su vez, con la ayuda del acelerómetro, se incluyó un filtro de Kalman[8] para poder refinar aún más los datos arrojados por la IMU.

Una vez garantizado un valor de desvío inerte a la métrica, se puede deducir un valor mínimo de tolerancia para un diferencial de ángulo que signifique movimiento.

10.2.4.3.5. Conclusión Con una tasa de muestro de 0.03s, cualquier diferencial de ángulo menor a 0.2 grados será considerado como ruido.

Dicho valor de tolerancia es menor al desplazamiento en profundidad del conductor para dos posiciones contiguas.

10.2.4.4. Control sobre los motores

10.2.4.4.1. Objetivo Garantizar magnitud y dirección del movimiento de los motores.

10.2.4.4.2. Pre-requisitos Correcta medición de inclinación del espejo (módulo anterior).
Control sobre los GPIOs del microcontrolador.
Espejo lateral completo (Motores, carcasa, espejo, cables).

10.2.4.4.3. Entrada Posición instantánea filtrada por la IMU.
Posición deseada a alcanzar.

10.2.4.4.4. Procedimiento Se comienza por determinar la posición final a la que el motor debe llegar. Para el caso de depuración del módulo, esta posición fue al azar; para situaciones reales, esta posición es un diferencial de movimiento que compensa la diferencia de profundidad/altura del conductor.

Una vez fijada la posición objetivo, se solicita la posición instantánea (mediante el módulo mencionado anteriormente) para determinar la dirección de movimiento.

Teniendo la magnitud y dirección del movimiento, se procede a dar comienzo a este habilitando los GPIOs del microcontrolador que cierran el circuito de los motores.

Como ya fue mencionado, el lazo de control es del tipo encendido-apagado y la tasa de muestro de la IMU de 0.03s, por lo que el sistema se realimenta con dicho período y cuando la posición es alcanzada, el movimiento es terminado instantáneamente.

10.2.4.4.5. Conclusión A través de las mediciones de la IMU se pudo verificar que el movimiento era realizado de forma exitosa.

Se encontraron casos donde el sistema se trababa (ejemplo, querer alcanzar una posición fuera de las limitaciones físicas). Para ello se incorporó un límite de tiempo de movimiento de 10s (para moverse de un extremo a otro el motor tarda menos de 10s).

11. Construcción del prototipo

11.1. Definición de los módulos

Para poder probar el algoritmo desarrollado para cumplir con el objetivo propuesto, se armó un prototipo que simule el producto final. Cada uno de los siguiente incisos describe un módulo adoptado en la construcción.

11.1.1. Alimentación

Todo el sistema debe poder ser abastecido por la batería del automóvil. Se corroboró con la placa de desarrollo que el equipo puede funcionar con 5V, 1A; el mínimo requerimiento de alimentación puede ser establecido.

Para poder determinar el módulo a utilizar se debieron considerar varios puntos: Entrada de 12VDC, máxima corriente de entrada 0.52A (INT-VIN-01, INT-VIN-02), salida 5Vdc 1A dc (estimación en base a los valores enunciados a continuación), salida constante pese a variaciones por ruido en la entrada, factor de eficiencia, inmunidad a interferencias electromagnéticas, baja emisión electromagnética, etc.

En la siguiente tabla se presentan los módulos que se conectan directamente a la etapa de alimentación (la IMU es alimentada desde la RPi y el puente H obtiene la corriente necesaria para mover los espejos directamente de la batería) y sus consumos nominales extraídos de sus respectivas hojas de datos.

Módulo	Consumo	Tensión
RPi	330mA	5VDC
Cámara	250mA	5VDC

Cuadro 30: Tabla de consumos

Los pines GPIO de la *Raspberry Pi* son capaces de entregar 16mA cada uno como máximo. Si se requiere utilizar todos al mismo tiempo, el consumo del total se limita a 50mA.

El módulo IMU se conecta a la *Raspberry Pi* alimentándose de la misma. En caso de que se quisiera armar un prototipo que acomode dos espejos al mismo tiempo, la *Raspberry Pi* es capaz de alimentar ambos módulos IMU con la misma corriente que figura en la tabla (cada una consume 6.5mA, por datasheet).

Por todas estas consideraciones, el módulo de alimentación para el sistema está constituido por el integrado LM2596.

11.1.2. Sensores para detectar movimiento de espejo

En primer lugar, se adquirió un espejo lateral con motores eléctricos para facilitar el armado del prototipo. No se buscó armar un espejo completo dado que el producto está destinado a utilizarse con espejos actuales sin modificar su forma ni movimiento. Se colocará en la parte trasera de cada espejo una IMU que será la encargada de informar la posición del espejo al microprocesador para que éste calcule cuando debe corregirse.

La IMU posee cuatro conectores: Alimentación, tierra, SCL (línea de sincronismo) y SDA (línea de datos). Debido a que los cables de estas señales se encuentran alejados de las principales fuentes de emisión electromagnética y las señales son de 3v3 con baja corriente, los valores en estas líneas (así como también los datos en las mismas) no son degradadas.

11.1.3. Unidad de procesamiento

A la hora de diseñar el prototipo se decidió utilizar un módulo de *Raspberry Pi* para correr el software de reconocimiento facial y movimiento de espejos. Para el caso donde el automóvil posee una computadora de a bordo y esta tiene capacidades similares al microcontrolador VoCore2 (uC seleccionado para el producto final), la idea es que cada fabricante sea libre de utilizar el microcontrolador que le sea más conveniente. Sin embargo, si el vehículo no posee computadora de a bordo, se adiciona un módulo complementario de diseño propio con el microcontrolador mencionado previamente o esta presenta una dificultad insalvable para su adaptación. Este módulo se lo define como unidad de procesamiento. La misma utilizará los puertos GPIO para controlar el movimiento de los espejos a través de los *drivers* L293D ya que la corriente utilizada por los motores es superior a la máxima corriente de los puertos GPIO. Los puertos I2C se utilizarán para interactuar con la IMU.

Por último, siendo que la cámara se comunica por protocolo UART, tanto en el prototipo como en el producto final se conectará a través del puerto USB.

11.1.4. Driver motores

Como ya fue mencionado, el objetivo es poder controlar los motores con un microprocesador. El problema es que debido a la tensión y corriente que requieren los motores para moverse, un microprocesador no puede

accionarlos directamente desde sus salidas. Por ello nace la necesidad de aislar la parte de procesamiento de la actuadora por medio de un *driver*.

Cabe destacar que algunos juegos de motores de espejos comparten un mismo borne de conexión. Por ende, si se quisiese controlar dos motores, la salida de dos *drivers* estarían conectadas en paralelo.

De ello surge una problemática ya que al accionar un motor y dejar el otro en reposo, la salida de uno de los *drivers* del motor en reposo estaría teniendo un nivel de tensión no acorde a lo requerido. Esto es, cuando se desee hacer circular corriente por uno de los motores, las terminales de este tendrán una diferencia de potencial; a su vez, si una de esas terminal se encuentra conectada con la terminal del otro motor el cual intenta colocar un nivel de tensión distinto, esto podría no solo ocasionar comportamiento extraño sino quemar el *driver*.

Otros factores a tener en cuenta a la hora de seleccionar un driver del conjunto de posibilidades disponibles son los consumos:

- Como ya fue mencionado en el apartado 11.1.1 los GPIOs que controlarán el accionar de los motores tienen limitada su corriente máxima a 16mA.
- Se impone que el driver comparta la salida del mismo módulo de alimentación que la unidad de procesamiento a modo de simplificar el diseño (5VDC).
- El driver debe poder suministrar a su salida la tensión y corriente necesaria para mover los motores del espejo lateral (12VDC / 30mA).
- Es deseable que la corriente de alimentación sea provista por los 12V de la batería en lugar de los 5V del módulo de alimentación. De esta forma, se exige menos de éste último mencionado.

Teniendo en cuenta esto dicho y luego de una exhaustiva investigación, se determinó utilizar el *driver* L293D. La decisión fue basada en que, a demás de poseer 4 *drivers* integrados lo cual posibilita la utilización de solo un integrado por set de motores, la salida de estos *drivers* poseen la protección necesaria para que las condiciones mencionadas no quemem el circuito.

11.1.5. Procesamiento óptico

Es necesario poder establecer los mínimos requerimientos para la cámara que será incorporada al vehículo de forma tal de garantizar la correcta funcionalidad.

Para poder hacer ello, se dispuso de una cámara web con diversos grados de resolución y se probó el código detector de ojos para diversas resoluciones hasta encontrar aquella mínima tal que asegurase la correcta detección.

Tal y como puede verse en la subsección 10.1.3, se determinó que la mínima resolución necesaria es de 640x480px.

11.2. Diseño de los circuitos impresos

El alcance de este proyecto abarca hasta la puesta en marcha de un prototipo capaz de cumplir lo deseado por un producto final. No obstante, se desarrolló un análisis (a continuación expuesto) de los aspectos que la placa del producto final debe cumplir para el caso que se lleve a la comercialización así como también se desarrollaron los esquemáticos adjuntados en siguientes incisos.

Las consideraciones que se hicieron conciernen al ruido de radiofrecuencia (RF) emitido por el microcontrolador. Este ruido es generado dentro del dispositivo y es acoplado por otros sistemas de diversas maneras. Está presente en todas las salidas, entradas, fuentes de alimentación, puestas a tierra.

Potencialmente, cada pin del microcontrolador puede ser un problema, ya que los caminos que hacen las conexiones entre si pueden formar una antena.

El segundo contribuidor más importante de ruido es la fuente de alimentación, incluido regulador de tensión y capacitores de filtro. Ellos son la fuente de energía de todo sistema y por ende de todo ruido.[7]

11.2.0.1. Sustrato El sustrato seleccionado para la confección de las placas del sistema debe ser fibra de vidrio FR4.

Este sustrato provee para los PCBs una base sólida tanto mecánica y eléctrica como también resistente cambios en las condiciones ambientales. Tiene una excelente aislación entre capas y rutas de cobre dentro de la misma capa. Además no presenta complicaciones en el rango de frecuencias que serán utilizados en el sistema. Se trata de un material muy utilizado en la industria durante muchos años, permitiendo que se perfeccionaran las técnicas a la hora del armado de placas y abaratando sus costos sin pérdida de calidad.

11.2.0.2. IPC-A-600 De acuerdo a lo establecido por el documento de aceptación de circuitos impresos[9], se establece que el circuito debe ser confeccionado bajo la clase 2.

Se trata de un circuito del cuál se desea que su vida útil sea larga y del cuál se espera que un servicio ininterrumpido.

No se solicitó mayor requisito ya que no se trata de un circuito de alto desempeño o que se someterá a ambientes hostiles.

11.2.0.3. IPC-A-601 De acuerdo a lo establecido por el documento de aceptación de ensamblajes electrónicos[10], se establece que el circuito debe ser confeccionado bajo la clase 2.

Se trata de un producto electrónico con aplicación dedicada y no general (por lo que se descarta la clase 1) donde su falla no implica la inutilización del vehículo o los espejos.

Siendo que no será expuesto a ambientes hostiles, la clase 3 también fue descartada.

11.2.0.4. Acabado en superficies El acabado del circuito, así como la colocación de los componentes sobre la placa, no juegan un factor determinante sobre el desempeño del producto.

Dentro del conjunto de acabados que se puede pedir al fabricante, se decidió por utilizar HASL ya que las desventajas que presenta frente a otros acabados no son apreciables para el funcionamiento del producto.

El análisis tenido en cuenta fue: Es un acabado de bajo costo el cual permite ser retrabajado en caso de fallas, las no uniformidades sobre la superficie no son problema y no importa el shock térmico al cuál es sometido, entre otras cosas.

11.2.0.5. Protecciones La IMU se conectará a conectores de baja corriente del microcontrolador. A su vez, siendo que la placa de alimentación garantiza niveles de tensión y corriente estables, esta placa no requiere protecciones adicionales.

El microcontrolador debe ser protegido para garantizar la longevidad de la operación. Pero ya que su operación no requiere grandes niveles de tensión o corriente, las protecciones implementadas no exceden la complejidad de un diodo Zener para limitar la tensión de entrada.

El fabricante del módulo utilizado para alimentar al sistema, adjunta en la hoja de datos de este el circuito a desarrollar para utilizarlo. Dicho circuito incluye ya ciertas protecciones que asegura su correcto funcionamiento para las condiciones a las cuales será sometido.

11.2.0.6. Conclusión En base a todos los aspectos recién mencionados, se pidió cotizar el diseño de la placa a un conjunto de fabricantes donde los precios son descritos a continuación.

Cabe mencionar que la cotización se hizo en base a un pedido de 100 placas. El proyecto se pensó para tener un alcance mayor pero con esta primer propuesta es suficiente para analizar costos. Además, de esta forma los precios quedan amortizados por al gran cantidad de placas.

Proveedor	País	Precio por placa	NRE	BOM	Precio assembly por placa	NRE assembly	Precio final
Edadoc	E.E.U.U.	U\$S 0.8	U\$S 150	U\$S 6.09	U\$S 1.50	U\$S 450	U\$S 1439
OurPCB	China	U\$S 0.248	U\$S 400	U\$S 1.947	U\$S 0.1	U\$S 158	U\$S 787.5
Bittele	E.E.U.U.	U\$S 2.39	U\$S 300	U\$S 4.3	U\$S 8.55	U\$S 250	U\$S 2074
Netvia Group	E.E.U.U.	U\$S 3.585	U\$S 450	U\$S 4	U\$S 12.3	U\$S 375	U\$S 2813.5
Hughes Circuits	E.E.U.U.	U\$S 5.3	U\$S 375	U\$S 5.23	U\$S 15.487	U\$S 561	U\$S 3537.7

Cuadro 31: Cotización

Teniendo en cuenta opiniones generales acerca de los fabricantes, obtenidas por diversas fuentes, se concluye que lo cotizado por OurPCB será suficiente para el proyecto.

11.3. Detalles de construcción y precauciones especiales de montaje

A la hora de equipar el sistema en el vehículo, cada módulo posee precauciones particulares en cuánto a su incorporación.

11.3.1. IMU

La IMU es la unidad que más restricciones poseía debido a las limitaciones físicas de los espejos retrovisores de hoy en día.

Los aspectos a analizar de la misma son:

1. Lugar de instalación

2. Aspectos relevantes del montaje
3. Qué tipo de conexionado tendría.

Lugar de instalación

Debido al diseño de los espejos retrovisores, la IMU debe ser incorporada de forma tal que no colisione con nada de forma que sea movida o removida.

Ya que la misma es utilizada solamente para medir ángulos y el alto es de unos cuantos milímetros, ello da la libertad de ser posicionada en la parte posterior del espejo alineando unos de los ejes de rotación de la IMU con el eje de rotación vertical del espejo.

Aspectos relevantes del montaje

La sujeción sobre el espejo debe ser lo suficientemente fuerte tal que cumpla con las especificaciones de vibración (INT-MEC-03) y más aún resista movimientos fuertes debido a la hostilidad el camino. Es decir, si el conductor circula sobre un camino sinuoso, la IMU debe permanecer en su lugar ya que cualquier modificación trae aparejada una descalibración.

Siendo que la resistencia del pegamento para espejos retrovisores es típicamente de 5.2N/mm^2 y los mismos quedan sujetos al parabrisas bajo las mismas situaciones a las que se estresará la IMU puede establecerse un requerimiento para la fijación de ésta siendo su valor el de resistencia recién mencionado.

11.3.1.1. Tipo de conexionado El conexionado de la IMU debe soportar las mismas fuerzas a las que se somete el adhesivo. Los conectores DF13 soportan fuerzas de hasta 3N/mm^2 y por ello se concluye que deben ser los utilizados para este tipo de conexionado.

11.3.2. Cámara

A su vez, debe estar bien sujeta al parante del automóvil y garantizar de alguna forma que no sea accidentalmente manipulada por imprudencias del usuario.

La solución escogida fue la de crear una carcasa protectora que contenga tanto el circuito como la lente de la cámara, dónde la adhesión de la misma al parante del auto sea del mismo material que el utilizado para colocar el receptor de las alarmas de los automóviles.

Lugar de instalación

Teniendo en cuenta que la dimensión del parabrisas (y por ende del parante) es distinta para cada automóvil, se debió establecer un estándar a ser aplicado a todo vehículo.

Se observó que a una altura de $3/4$ respecto al alto total del parante, la cámara posee un campo visual que logra abarcar ambos ojos del conductor, cuando la altura de este varía entre los 2.1m y 1.55m (alturas consideradas en base a REQ-18)

Esta cualidad se observó en una camioneta SUV (Musso SangYong), en un sedan (Nissan Primera) y en un auto compacto (Hyundai i30).

11.3.3. Sistema

La parte que acarrea un gran cuidado dentro de la integración del sistema es la locación del mismo dentro del automóvil.

La placa no puede ser instalada sin protección mecánica ni montada de forma directa a cualquier superficie.

Lugar de instalación

Este lugar debe ser tal que: Sea el mismo para todo modelo de vehículo o sea particular para cada situación.

El lugar de instalación será de la puerta para que este recubierto y protegido del alcance del usuario y de la intemperie. Se colocará a 20 cm de la bisagra para que esté cerca de la botonera, del espejo y del parante. Siendo que 20cm es una distancia conservadora no hay se asegura que sea una zona despeja de todo lo conserniente al mecanismo de la puerta (ejemplo bisagra, mecanismo de traba, trabas del panel de puerta). A su vez debe ser colocado sobre una superficie para garantizar una sujeción uniforme de toda la placa. Por último, la placa será sujeta por 4 tornillos, uno en cada esquina.

12. Validación del prototipo

12.1. Resultados

Se aplicaron los tests descritos previamente y los resultados fueron los siguientes.

Test ID	Criterio	Resultado
T-INT-01	Post transitorio $V_{in_{MAX}} = 14,7V$ $V_{in_{MIN}} = 9,6V$ $V_{out} = 0V \pm 0,1V$ Indicador de operación = Apagado	Post transitorio ✓ $V_{in_{MAX}} = 12,6V$ ✓ $V_{in_{MIN}} = 11,5V$ ✓ $V_{out} = 0V \pm 0,1V$ ✓ Indicador de operación = Apagado
T-INT-02	$V_{in_{MAX}} = 14,7V$ $V_{in_{MIN}} = 9,6V$ $V_{out} = 0V \pm 0,1V$ Indicador de operación = Apagado	✓ $V_{in_{MAX}} = 14,56V$ ✓ $V_{in_{MIN}} = 9,5V$ ✓ $V_{out} = 0,010V \pm 0,1V$ ✓ Indicador de operación = Apagado
T-INT-03	$V_{in_{MAX}} = 14,7V$ $V_{in_{MIN}} = 9,6V$ $V_{out_{MAX}} = 14,7V$ $V_{out_{MIN}} = 9,6V$ Indicador de operación = Encendido	✓ $V_{in_{MAX}} = 14,56V$ ✓ $V_{in_{MIN}} = 9,5V$ ✓ $V_{out_{MAX}} = 14,56V$ ✓ $V_{out_{MIN}} = 9,5V$ ✓ Indicador de operación = Encendido

Cuadro 32: Tests de Interfaz VIN-VOUT

Test ID	Criterio	Resultado
T-MAN-01	Indicador de operación = Encendido Indicador de falla de reconocimiento de rostro = Encendido	✓ Indicador de operación = Encendido ✓ Indicador de falla de reconocimiento de rostro = Encendido

Cuadro 33: Tests de Mantenibilidad

Test ID	Criterio	Resultado
T-FUNC-01	Indicador de operación = Encendido Manija izquierda trasera, dentro del círculo de precisión Indicador de falla de reconocimiento de rostro = Apagado	✓ Indicador de operación = Encendido ⚡ Manija izquierda trasera, dentro del círculo de precisión (90 % de las veces) ✓ Indicador de falla de reconocimiento de rostro = Apagado
T-FUNC-02	Indicador de operación = Encendido Manija izquierda trasera, dentro del círculo de precisión Indicador de falla de reconocimiento de rostro = Apagado	✓ Indicador de operación = Encendido ✓ Manija izquierda trasera, dentro del círculo de precisión (100 % de las veces) ✓ Indicador de falla de reconocimiento de rostro = Apagado
T-FUNC-03	Indicador de operación = Encendido Manija izquierda trasera, dentro del círculo de precisión Indicador de falla de reconocimiento de rostro = Apagado	✓ Indicador de operación = Encendido ⚡ Manija izquierda trasera, dentro del círculo de precisión (92 % de las veces) ✓ Indicador de falla de reconocimiento de rostro = Apagado

Cuadro 34: Tests de funcionalidad

12.2. Evaluación

En base a los resultados obtenidos en las tablas anteriores se dan las siguientes conclusiones:

- Pese a haber limitado la resolución de la IMU a $0.007^{\circ}/s$ y aplicar un filtro de mediana y de Kalman, en dos muestras consecutivas espaciadas 0.4s la diferencia pudo llegar a resultar de 1° .

- Pese a haber alineado el eje de rotación vertical del espejo con el los ejes de rotación de la IMU el movimiento en una sola dirección del espejo se manifiesta en variaciones en la lectura de los dos ejes de la IMU.
- En situaciones en las que encontrar los ojos resultaba difícil para la parte de procesamiento de imágenes el tiempo de respuesta se limitaba a 10 segundos tomando como punto de control la mitad de la altura del rostro reconocido. Es deseable aumentar la velocidad de procesamiento para que el reconocimiento de los ojos se realice dentro del tiempo planteado en los requerimientos.
- En los tests funcionales 01 y 03, el éxito de la operación no se obtuvo el 100% de las veces. La razón de ello fue que el algoritmo fue acumulando pequeños errores a lo largo de su utilización. Sin embargo, el cúmulo de dichos errores no imposibilitaba su funcionamiento, y al realizar pequeños ajustes manuales, el proyecto volvía a funcionar de forma correcta. Los usuarios del banco de prueba no le dieron mayor importancia por lo que se aprueba, finalmente, su desempeño. Como será mencionado a continuación, se deben considerar futuras mejoras para evitar la corrección del usuario.

Fuera de lo contemplado por los tests

- El mínimo nivel de luminosidad para que el código de detección funcione fue de 0.95 EV100¹, es decir 0.241 $\frac{cd}{m^2}$. Por debajo de ello (0.12 $\frac{cd}{m^2}$ típicamente) la detección aumentaba considerablemente su tasa de fallos. Este análisis no posee un banco de prueba porque no estaba contemplado dentro del proyecto.

12.3. Consideraciones finales hacia el producto final

Teniendo en cuenta las consideraciones realizadas en la evaluación de resultados se concluye que el módulo para el cierre del lazo de control, actualmente realizado por la IMU, debería ser optimizado de alguna de las siguientes maneras:

- En el mercado se haya una gran variedad de potenciómetros lineales cuyo diminuto tamaño permite ser incorporado al receptáculo del espejo.
- Escoger una IMU cuya lectura sea más precisa como por ejemplo ADXL335.
- Aumentar la robustez de los filtros escogidos a los que se le aplican las mediciones tomadas por la IMU

Finalmente en base a la última observación, convendría compilar el software para que sea ejecutado en una unidad de procesamiento gráfica dedicada (GPU).

Para solventar lo ocurrido fuera de lo contemplado por los *tests* se puede incorporar la rutina de detección de poca luminosidad presente en este informe y acoplar una fuente de iluminación externa (puede ser la lámpara ya incorporada en el habitáculo de los vehículos) para funcionar en situaciones de poca luz.

12.4. Estudios de confiabilidad de hardware y software

12.4.1. Software

12.4.1.1. Modelo de predicción Se partió del modelo de MUSA para poder predecir la tasa inicial de fallas.

Este modelo permite estimar la cantidad de fallas esperadas en un tiempo infinito dado por

$$\lambda_0 = k \cdot p \cdot w_0$$

- k es una constante cuantizada para la estructura dinámica del programa y de las máquinas. Al no conocerse expresamente para este caso el valor de k , se toma $4, 2 \times 10^{-7}$.
- p estima el número de ejecuciones por unidad de tiempo y viene dado por $p = \frac{r}{\frac{SLOC}{ER}}$
 - r es el promedio de la velocidad de ejecución de instrucciones, determinado por el fabricante o por benchmarking.
 - SLOC son las líneas de código fuente.

¹Partiendo de un punto arbitrario de luminosidad suficiente para que el sistema se desempeñe correctamente, se dió inicio a disminuir paulatinamente dicho punto hasta que el sistema falle completamente.

En ese instante, se colocó un celular a la altura de la cabeza del conductor a modo de sensar la luminosidad en dicha área.

Para verificar el resultado obtenido, se aumentó la iluminación a modo que el sistema vuelva a funcionar y se corroboró que la luz sensada era, en magnitud, más alta respecto al mínimo encontrado.

- ER es una constante cuyo valor depende del lenguaje de programación : Assembler=1 Macroassembler=1.5, C=2.5, COBOL, FORTRAN=3, ADA=4.5
- w_0 es la estimación del número inicial de fallas en el programa. Por default se puede asumir 6 fallas/ 1000 SLOC. Puede ser calculado utilizando $w_0 = N.B = N \cdot B$
 - N Número total de defectos inherentes. Se estima basándose en anteriores experiencias o utilizando criterio.
 - B Proporción de defectos que se convierten en fallas. Proporción de defectos no corregidos antes de la distribución. Se puede asumir 95 %.

Para el caso de este proyecto, el código de python de OpenCV es compilado en C por lo que se toma ese valor para el factor ER. Conociendo el microcontrolador, se sabe que trabaja a 700MHz.

El código de openCV para detectar rostros cuenta con 1773[6].

La tasa de fallas queda como

$$\lambda_0 = 4,2x10^{-7} \cdot \frac{700x10^6}{\frac{1773}{2,5}} \cdot \frac{6}{1000} = 4,487x10^{-3} f/seg$$

Equivalente a decir una falla cada 222 segundos.

Dos cosas deben ser mencionadas: El software de reconocimiento estará acotado a 10 segundos de uso máximo, entonces se obtendría un error cada 22 accionares. Además, este modelo de predicción es una sobre-estimación del código final. Es decir, una falla cada 222 segundos es el caso para un programa que apenas fue desarrollado e implementado, sin tener en cuenta rutinas de seguridad en caso de errores.

Por ello, este número da idea de la tasa de errores para la versión apenas desarrollada, sin tener en cuenta el software final.

12.4.1.2. Modelo de estimación El objeto de este análisis es el de determinar número de fallas remanentes en el soft justo antes del testeo final; lo que permitiría determinar la longitud de los test a efectuar. Para ello se utilizó el modelo de Shooman.

La razón de ello es que comparte similitudes con el modelo exponencial general donde todos los errores poseen igual grado de severidad, probabilidad de detección y que cada uno es corregido de manera inmediata una vez detectado pero cada falla está normalizada por las líneas de código de esta. El principal uso de la normalización es poder detectar relaciones de un modo más claro y en un sistema donde los datos de entrada atraviesan una serie de procesos anidados, encontrar una relación entre los diversos errores resulta muy importante.

Un claro ejemplo fue cuando se encontró una relación entre la falla de detección de rostros y la de detección de ojos. Ambos fueron resueltos de igual forma.

En la siguiente tabla se demuestra el avance del proyecto. Para referir lo sucedido en cada semana, vea la bitácora de software.

Semanas	Errores totales acumulados	Errores corregidos	Errores residuales
1	13	12	1
2	28	23	5
3	39	32	7
4	41	39	2
5	50	49	1

Cuadro 35: Tabla de errores

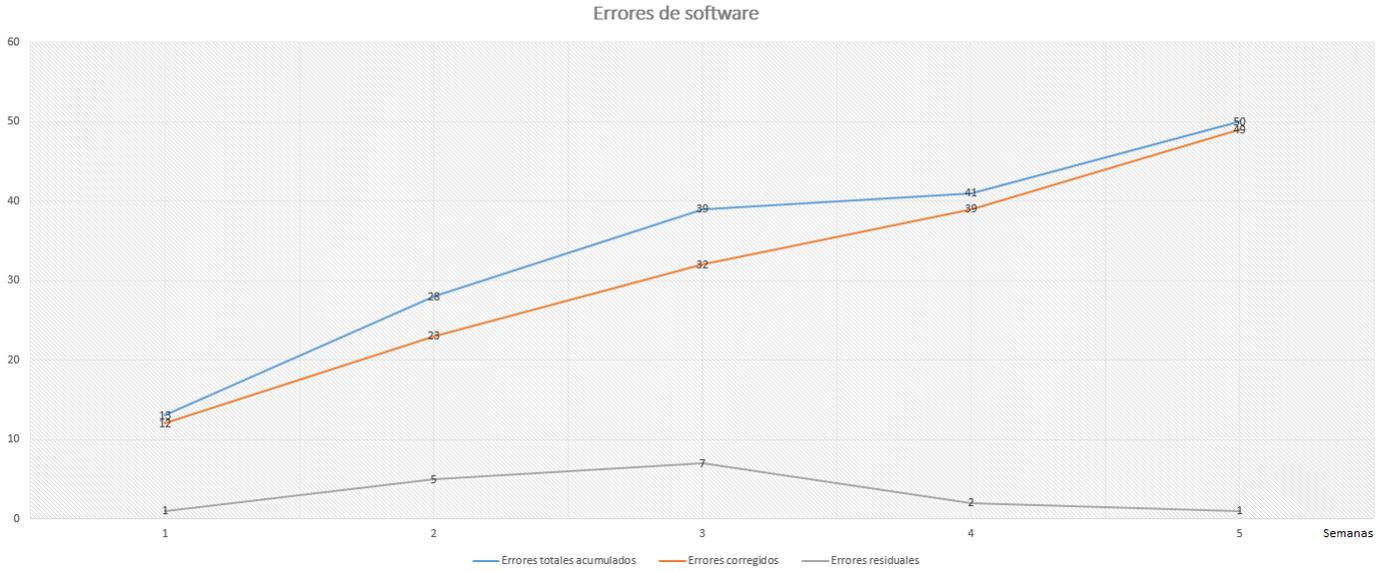


Figura 34: Histórico de errores

Puede ser que hayan más errores, pero hasta la fecha el código a funcionado con 1 falla y es debido a las condiciones lumínicas.

Durante la semana 3 se obtuvieron los siguientes valores acumulados: $H(3 \text{ sem.})=504 \text{ hs.}$ $r(3 \text{ sem.})= 3$ falla catastrófica. Por lo tanto $H/r = 168. \Rightarrow \lambda_1 = 0.005952380$.

Al final del plan de depuración, se obtuvieron los siguientes datos (siempre acumulados):

$H(5 \text{ sem}) = 840 \text{ hs.}$ $r(5 \text{ sem.}) = 1$ falla catastrófica. Por lo tanto $H/r = 840. \Rightarrow \lambda_2 = 0,001190476$.

Con estos datos se procede a calcular los parámetros del sistema al final de las 5 semanas donde se considera que se encuentra estabilizado:

$$\hat{E}_T = -\frac{\left[\frac{\lambda_2}{\lambda_1}\right] \cdot E_c(t_1) - E_c(t_2)}{\frac{\lambda_2}{\lambda_1} - 1} = -\frac{\left[\frac{0,001190476}{0,005952380}\right] \cdot 32 - 49}{\frac{0,001190476}{0,005952380} - 1} = 53$$

53 errores totales estimados frente a 50 errores medidos.

Luego:

$$\frac{k}{I_T} = \frac{\lambda_2}{\hat{E}_T - E_C} = 0,000297619$$

Por lo que λ estimado para nuestro software es $0.000297619 \cdot 2 = 0.000595238$ con un MTTF estimado de 420 hs

Es importante destacar lo mismo que en el inciso anterior: Cada accionar estará comprendido en 10 segundos de uso de software. Si se hace referencia a este último valor, se obtendría un MTTF de 151200 ciclos de uso; a un estimado de un ajuste por día, se trata de aproximadamente 414 años de MTTF.

12.4.2. Hardware

Para calcular la confiabilidad del sistema, se establece que la falla de cualquier componente del mismo producirá una falla catastrófica. Esto quiere decir que todo el sistema deja de funcionar, disminuyendo la vida útil en gran medida. Para el cálculo del MTBF de cada componente, se toma la norma militar *MIL HDBK 217F*. De aquí surgen dos ecuaciones:

$$MTBF_{componente}(1/FIT) = \left(1000 \cdot \lambda_{base} \cdot \prod \pi_i\right)^{-1} \quad (5)$$

$$MTBF_{módulo}(horas) = \left(\sum \left(\frac{1}{MTBF_{componente_i}}\right)\right)^{-1} \quad (6)$$

Con la ecuación 5 se calcula el MTBF de cada componente usando de los datos provistos por los fabricantes de IC. El factor π_i es el valor de los modificadores de cada componente. Luego se calcula el MTBF de cada módulo con la ecuación 6.

Componente		Encapsulado	Cantidad	λ_b	π_Q	π_E	π_C	π_V	π_T	π_{SR}	$\lambda_{ef}(FIT)$
Capacitor	680uF	CT	1	0.0000072	10	20	1.8	1.24	0.79	1	2.54
Capacitor	220uF	CT	1	0.0000072	10	20	1.6	5	0.79	1	9.1

Cuadro 36: Confiabilidad Capacitores

Componente	Valor	Encapsulado	Cantidad	λ_b	π_Q	π_E	π_R	$\lambda_{ef}(FIT)$
Jumper	-	RC	1	0.0017	10	16	1	27.2
Resistencia	10K	RC	6	0.0017	10	16	1	163.2

Cuadro 37: Confiabilidad resistores

Componente	Encapsulado	Cantidad	λ_b	π_Q	π_E	π_S	π_C	π_T	$\lambda_{ef}(FIT)$
1N5824	-	1	.0030	8	9	0.054	1	1	116.64
LM2596DSADJR4G	TO-220	1	0.0020	8	9	1	1	1	144

Componente	Valor	Encapsulado	Cantidad	λ_b	π_Q	π_E	π_S	π_C	π_T	π_A	π_R	C_1	C_2	$\lambda_{ef}(FIT)$
BJT (Puente H doble)	-	-	8	0.00074	8	9	0.39	-	4.5	1.5	0.69	-	-	774.23
IMU - LSM9DS0	-	LGA 24	1	0.16	2	4	-	-	-	-	-	0.51	6.1	3982
Inductancia	33uH	-	1	0.00003	0.3	12	-	1	-	-	-	-	-	0.000108

Cuadro 38: Confiabilidad otros componentes

Utilizando la ecuación 6 calculamos el MTBF total. Como queda plasmado en el cuadro 33, el componente que más influye en el MTBF es la IMU. Cabe destacar que se espera un valor bajo dado que se está considerando que cuando cualquiera de los componentes falla, el sistema deja de funcionar. El valor de MTBF obtenido es de *191hs*.

A pesar de que el valor es bajo, como se esperaba, se puede hacer un análisis de este número. El ciclo de uso de este producto es de unos pocos segundos. Suponiendo un valor extremo de 60s por ciclo, y 2 ciclos por día, estamos ante una vida útil de unos 15 años aproximadamente. Con lo cual, se concluye que el valor obtenido es aceptable para el uso previsto del dispositivo.

13. Conclusiones

13.1. Cumplimiento de Objetivos

Robustez

Luego de haber analizado lo expuesto en la sección evaluación se concluye que el proyecto en el estado actual no cumple con uno de los requerimientos exigidos (que funcione las 24hs del día). Excluyendo las situaciones de baja y excesiva iluminación, el dispositivo pudo resolver dentro del marco de tiempo requerido (10 segundos) en todas las situaciones con mayor o menor resolución. Es decir, son pocos los casos donde se estimó el punto de control en base al centro de la cara y no la posición de los ojos (2 de cada 10).

Versatilidad

El prototipo fue probado en escenarios simulando los tres tipos de vehículos ya mencionados demostrando ser capaz de funcionar en cada uno. Fue necesario realizar una calibración para cada uno de ellos, lo cual era algo esperado.

Sencillez de diseño

La sencillez es algo implícito en cualquier tipo de proyecto que se desee ser lanzado al mercado general.

No se poseía previa idea de las diversas disposiciones de los motores para diferentes espejos ni del espacio disponible en el receptáculo. Sin embargo, la meta estaba fijada en permitir la incorporación del proyecto a cualquier automóvil.

Más allá de las consideraciones antedichas, el objetivo fue alcanzado exitosamente permitiendo acoplar el sistema a cualquier configuración de motor eléctrico, sin la necesidad de cambiar nada del automóvil y solamente adicionando una IMU.

13.2. Lecciones aprendidas

Algunas de las lecciones que vale la pena mencionar son la no subestimación de tiempos. En un principio se creyó que la confección del informe no llevaría tanto tiempo como el que finalmente terminó demandando. En próximos proyectos se tendrá en cuenta que la elaboración de toda la documentación puede llevar incluso más tiempo que la parte práctica, pudiendo demorar todo el proyecto más de lo pensado.

Por otro lado se aprendió a modularizar el trabajo. En un principio, el proyecto fue ideado como un único módulo donde no había una clara distinción entre los límites de cada bloque (alimentación, procesamiento de imágenes, control). Esto llevó a extender la etapa de diseño más allá de lo propuesto en el Gantt dificultando la selección de componentes y por ende dilatando la realización del proyecto. Al separar cada módulo con límites claros abstrayendo uno del otro y especificando claramente sus entradas y salidas, es decir, cuánto sería la tensión máxima de salida del convertidor de tensión, cuáles son las entradas esperadas por el código de compensación del ángulo, etc. el desarrollo del sistema se agilizó y la depuración de cada módulo se realizó de forma más concreta y efectiva.

Otra lección aprendida fue el cálculo de la factibilidad económica para un proyecto de esta envergadura. Descubrimos que no tuvimos en cuenta muchos detalles que terminaron siendo decisivos a la hora de determinar si un proyecto es viable o no (mano de obra, impuestos, requisitos para la creación de una sociedad, cálculo de mercado, licenciamiento y patentamiento, entre otros).

Por último, pero no menos importante, se subrayó la importancia de la investigación previa. Si la misma no es completa y extensiva, pueden surgir problemas durante el desarrollo del proyecto que no fueron tenidos en cuenta. Estos imprevistos pueden demorar el proyecto, y en el peor de los casos, hasta imposibilitarlo. En nuestro caso particular, la forma de crear un lazo de control para el motor, la forma de medir profundidad (que pudo ser solventada), la iluminación del rostro, etc.

13.3. Recomendaciones para futuros diseños

Si bien se pudieron cumplir con muchos de los objetivos planteados al comienzo del proyecto, la tendencia en la industria automotriz indica que todos los vehículos en el futuro contarán con una computadora de abordo o un sistema capaz de correr el programa diseñado sin necesidad de incorporar una placa externa. Por este motivo, no se consideran futuras correcciones a nivel hardware del microcontrolador.

Por otro lado, se pueden mejorar aspectos de software en iteraciones futuras. Estas mejoras incluyen:

- Optimizar las mediciones obtenidas con la IMU de manera tal de poder tener datos más precisos y reducir el error en el posicionamiento.

- Mejoras de software asociadas a un cambio del sensor. Como se mencionó en el desarrollo del informe, el sensor ideal es un encoder. Futuras implementaciones del *soft* serán mucho más versátiles si se dispone de medidas exactas tomadas desde un encoder incorporado en el espejo. De todas formas, esto ya queda fuera del alcance de este proyecto dado que depende íntegramente de las empresas automotrices.

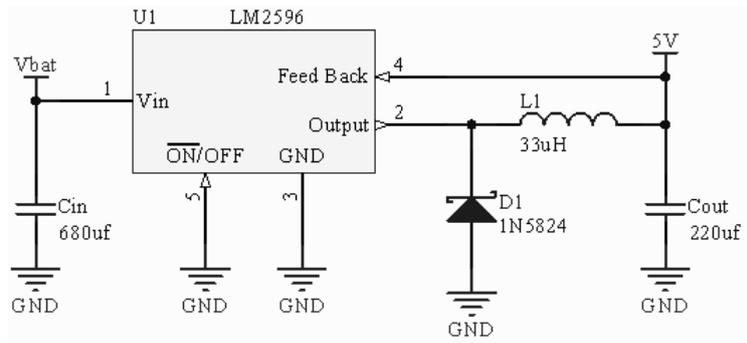


Figura 37: Esquemático Fuente Switching

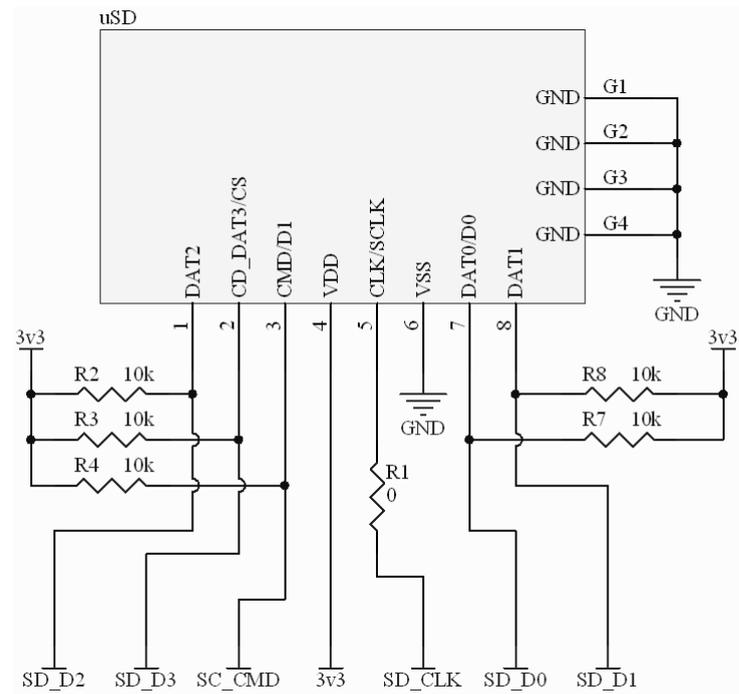


Figura 38: Esquemático uSD

14.2. Planos de PCB

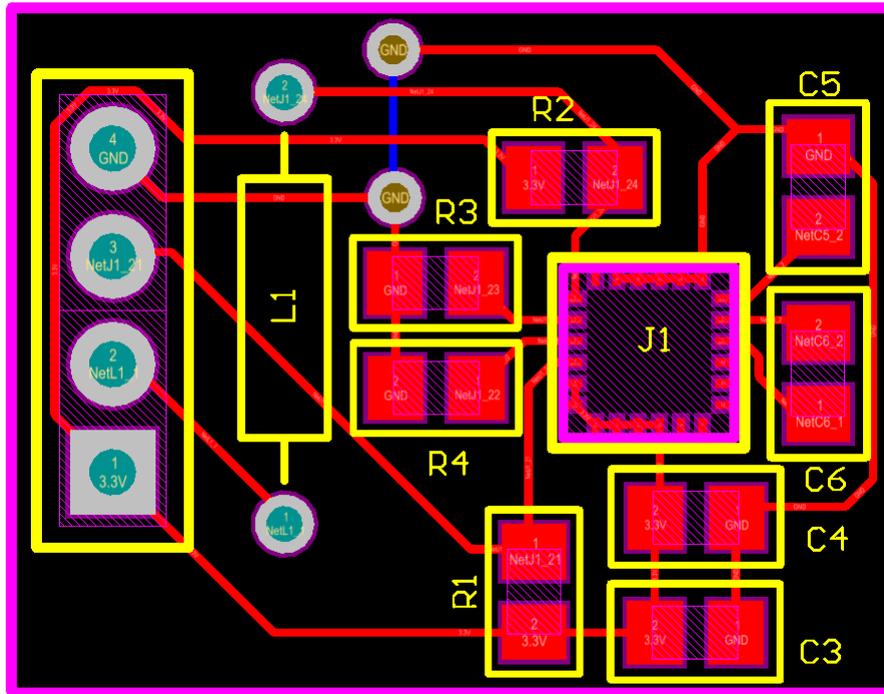


Figura 39: PCB IMU

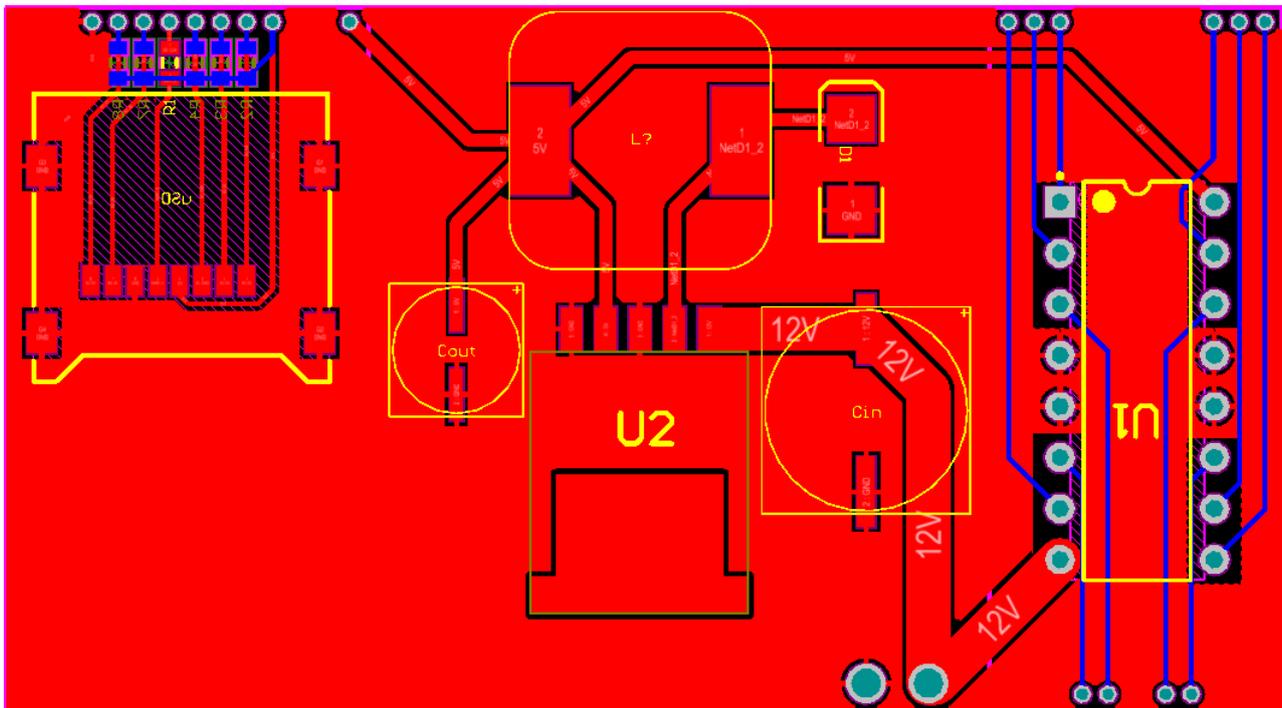


Figura 40: PCB Placa controladora

14.3. Listado de Partes y Componentes (BOM)

Descripción	Fabricante	Part Number del Fabricante	Cantidad
IMU	STMicroelectronics	LSM9DS0TR	1
Resistencia 10 $k\Omega$	Yageo	RC0402JR-0710KL	4
Capacitor 4.7 μF	Murata Electronics North America	GRM21BF51E475ZA01L	1
Capacitor 0.22 μF	Murata Electronics North America	GRM033R60G224ME15D	1
Capacitor 10 μF	Murata Electronics North America	GRM21BR60J106ME19L	1
Capacitor 100 nF	Murata Electronics North America	GRM188F51E104ZA01D	1
Inductor 10 μH	Abracon LLC	AIML-0603-100K-T	1

Cuadro 39: BOM IMU

Descripción	Fabricante	Part Number del Fabricante	Cantidad
CPU	VoCore	VoCore2	1
Capacitor 680 μF	Panasonic Electronic Components	EEE-FK1C681P	1
Capacitor 220 μF	Panasonic Electronic Components	EEE-0JA221WP	1
Diodo Zener 30V	Microsemi	1N5824	1
Inductor 33 μH	Würth Electronics Inc.	7447715330	1
Resistencia 0 Ω	Yageo	RC0603JR-070RL	1
Resistencia 10 $k\Omega$	Yageo	RC0603JR-0710KL	5
Driver	STMicroelectronics	L293D	1
Fuente switching	ON Semiconductor	LM2596DSADJR4G	1
Zócalo uSD	Molex, LLC	0472192001	1

Cuadro 40: BOM placa controladora

14.4. Códigos de Software

14.4.1. Lectura de IMU

```
#!/usr/bin/python
#
# This program reads the angles from the accelerometer, gyroscope
# and magnetometer on a BerryIMU connected to a Raspberry Pi.
#
# This program includes two filters (low pass and median) to improve the
# values returned from BerryIMU by reducing noise.
#
# Copyright (C) 2017 Mark Williams

import sys
import smbus
import time
import math
from LSM9DS0 import *
import time
bus = smbus.SMBus(1)

IMU_upside_down = 0 # Change calculations depending on IMU orientation.
                    # 0 = Correct side up. This is when the skull logo is facing down
                    # 1 = Upside down. This is when the skull logo is facing up

RAD_TO_DEG = 57.29578
M_PI = 3.14159265358979323846
G_GAIN = 0.070 # [deg/s/LSB]
AA = 0.40 # Complementary filter constant
MAG_LPF_FACTOR = 0.4 # Low pass filter constant magnetometer
ACC_LPF_FACTOR = 0.4 # Low pass filter constant for accelerometer
ACC_MEDIANTABLESIZE = 9 # Median filter table
MAG_MEDIANTABLESIZE = 9 # Median filter table

#Kalman filter variables
Q_angle = 0.02
Q_gyro = 0.0015
R_angle = 0.005
y_bias = 0.0
x_bias = 0.0
XP_00 = 0.0
XP_01 = 0.0
XP_10 = 0.0
XP_11 = 0.0
YP_00 = 0.0
YP_01 = 0.0
YP_10 = 0.0
YP_11 = 0.0
KFangleX = 0.0
KFangleY = 0.0
```

```

def kalmanFilterY ( accAngle , gyroRate , DT):
    y=0.0
    S=0.0

    global KFangleY
    global Q_angle
    global Q_gyro
    global y_bias
    global YP_00
    global YP_01
    global YP_10
    global YP_11

    KFangleY = KFangleY + DT * (gyroRate - y_bias)

    YP_00 = YP_00 + ( - DT * (YP_10 + YP_01) + Q_angle * DT )
    YP_01 = YP_01 + ( - DT * YP_11 )
    YP_10 = YP_10 + ( - DT * YP_11 )
    YP_11 = YP_11 + ( + Q_gyro * DT )

    y = accAngle - KFangleY
    S = YP_00 + R_angle
    K_0 = YP_00 / S
    K_1 = YP_10 / S

    KFangleY = KFangleY + ( K_0 * y )
    y_bias = y_bias + ( K_1 * y )

    YP_00 = YP_00 - ( K_0 * YP_00 )
    YP_01 = YP_01 - ( K_0 * YP_01 )
    YP_10 = YP_10 - ( K_1 * YP_00 )
    YP_11 = YP_11 - ( K_1 * YP_01 )

    return KFangleY

```

```

def kalmanFilterX ( accAngle , gyroRate , DT):
    x=0.0
    S=0.0

    global KFangleX
    global Q_angle
    global Q_gyro
    global x_bias
    global XP_00
    global XP_01
    global XP_10
    global XP_11

    KFangleX = KFangleX + DT * (gyroRate - x_bias)

    XP_00 = XP_00 + ( - DT * (XP_10 + XP_01) + Q_angle * DT )
    XP_01 = XP_01 + ( - DT * XP_11 )
    XP_10 = XP_10 + ( - DT * XP_11 )
    XP_11 = XP_11 + ( + Q_gyro * DT )

    x = accAngle - KFangleX
    S = XP_00 + R_angle
    K_0 = XP_00 / S
    K_1 = XP_10 / S

    KFangleX = KFangleX + ( K_0 * x )
    x_bias = x_bias + ( K_1 * x )

    XP_00 = XP_00 - ( K_0 * XP_00 )
    XP_01 = XP_01 - ( K_0 * XP_01 )
    XP_10 = XP_10 - ( K_1 * XP_00 )
    XP_11 = XP_11 - ( K_1 * XP_01 )

    return KFangleX

def writeACC(register , value):
    bus.write_byte_data(ACC_ADDRESS , register , value)
    return -1

def writeMAG(register , value):
    bus.write_byte_data(MAG_ADDRESS, register , value)
    return -1

def writeGRY(register , value):
    bus.write_byte_data(GYR_ADDRESS, register , value)
    return -1

def readACCx():
    acc_l = bus.read_byte_data(ACC_ADDRESS, OUT_X_L_A)
    acc_h = bus.read_byte_data(ACC_ADDRESS, OUT_X_H_A)
    acc_combined = (acc_l | acc_h <<8)
    return acc_combined if acc_combined < 32768 else acc_combined - 65536

```

```

def readACCy():
    acc_l = bus.read_byte_data(ACC_ADDRESS, OUT_Y_L_A)
    acc_h = bus.read_byte_data(ACC_ADDRESS, OUT_Y_H_A)
    acc_combined = (acc_l | acc_h <<8)
    return acc_combined if acc_combined < 32768 else acc_combined - 65536

def readACCz():
    acc_l = bus.read_byte_data(ACC_ADDRESS, OUT_Z_L_A)
    acc_h = bus.read_byte_data(ACC_ADDRESS, OUT_Z_H_A)
    acc_combined = (acc_l | acc_h <<8)
    return acc_combined if acc_combined < 32768 else acc_combined - 65536

def readMAGx():
    mag_l = bus.read_byte_data(MAG_ADDRESS, OUT_X_L_M)
    mag_h = bus.read_byte_data(MAG_ADDRESS, OUT_X_H_M)
    mag_combined = (mag_l | mag_h <<8)
    return mag_combined if mag_combined < 32768 else mag_combined - 65536

def readMAGy():
    mag_l = bus.read_byte_data(MAG_ADDRESS, OUT_Y_L_M)
    mag_h = bus.read_byte_data(MAG_ADDRESS, OUT_Y_H_M)
    mag_combined = (mag_l | mag_h <<8)
    return mag_combined if mag_combined < 32768 else mag_combined - 65536

def readMAGz():
    mag_l = bus.read_byte_data(MAG_ADDRESS, OUT_Z_L_M)
    mag_h = bus.read_byte_data(MAG_ADDRESS, OUT_Z_H_M)
    mag_combined = (mag_l | mag_h <<8)
    return mag_combined if mag_combined < 32768 else mag_combined - 65536

def readGYRx():
    gyr_l = bus.read_byte_data(GYR_ADDRESS, OUT_X_L_G)
    gyr_h = bus.read_byte_data(GYR_ADDRESS, OUT_X_H_G)
    gyr_combined = (gyr_l | gyr_h <<8)
    return gyr_combined if gyr_combined < 32768 else gyr_combined - 65536

def readGYRy():
    gyr_l = bus.read_byte_data(GYR_ADDRESS, OUT_Y_L_G)
    gyr_h = bus.read_byte_data(GYR_ADDRESS, OUT_Y_H_G)
    gyr_combined = (gyr_l | gyr_h <<8)
    return gyr_combined if gyr_combined < 32768 else gyr_combined - 65536

```

```

def readGYRz():
    gyr_l = bus.read_byte_data(GYR_ADDRESS, OUT_Z_L_G)
    gyr_h = bus.read_byte_data(GYR_ADDRESS, OUT_Z_H_G)
    gyr_combined = (gyr_l | gyr_h <<8)
    return gyr_combined if gyr_combined < 32768 else gyr_combined - 65536

#initialise the accelerometer
writeACC(CTRL_REG1_XM, 0b01100111) #z,y,x axis enabled, continuous update,
100Hz data rate
writeACC(CTRL_REG2_XM, 0b00100000) #+/- 16G full scale

#initialise the magnetometer
writeMAG(CTRL_REG5_XM, 0b11110000) #Temp enable, M data rate = 50Hz
writeMAG(CTRL_REG6_XM, 0b01100000) #+/-12gauss
writeMAG(CTRL_REG7_XM, 0b00000000) #Continuous-conversion mode

#initialise the gyroscope
writeGRY(CTRL_REG1_G, 0b00001111) #Normal power mode, all axes enabled
writeGRY(CTRL_REG4_G, 0b00110000) #Continuous update, 2000 dps full scale

gyroXangle = 0.0
gyroYangle = 0.0
gyroZangle = 0.0
CFangleX = 0.0
CFangleY = 0.0
CFangleXFiltered = 0.0
CFangleYFiltered = 0.0
kalmanX = 0.0
kalmanY = 0.0
oldXMagRawValue = 0
oldYMagRawValue = 0
oldZMagRawValue = 0
oldXAccRawValue = 0
oldYAccRawValue = 0
oldZAccRawValue = 0

acc_medianTable1X = [1] * ACC_MEDIANTABLESIZE
acc_medianTable1Y = [1] * ACC_MEDIANTABLESIZE
acc_medianTable1Z = [1] * ACC_MEDIANTABLESIZE
acc_medianTable2X = [1] * ACC_MEDIANTABLESIZE
acc_medianTable2Y = [1] * ACC_MEDIANTABLESIZE
acc_medianTable2Z = [1] * ACC_MEDIANTABLESIZE
mag_medianTable1X = [1] * MAG_MEDIANTABLESIZE
mag_medianTable1Y = [1] * MAG_MEDIANTABLESIZE
mag_medianTable1Z = [1] * MAG_MEDIANTABLESIZE
mag_medianTable2X = [1] * MAG_MEDIANTABLESIZE
mag_medianTable2Y = [1] * MAG_MEDIANTABLESIZE
mag_medianTable2Z = [1] * MAG_MEDIANTABLESIZE

```

```

def imu_control(last_call_s):

    global oldXMagRawValue
    global oldYMagRawValue
    global oldZMagRawValue
    global oldXAccRawValue
    global oldYAccRawValue
    global oldZAccRawValue
    global gyroXangle
    global gyroYangle
    global gyroZangle
    global CFangleX
    global CFangleY

    #Read the accelerometer ,gyroscope and magnetometer values
    ACCx = readACCx()
    ACCy = readACCy()
    ACCz = readACCz()
    GYRx = readGYRx()
    GYRy = readGYRy()
    GYRz = readGYRz()
    MAGx = readMAGx()
    MAGy = readMAGy()
    MAGz = readMAGz()

    ##Calculate loop Period(LP). How long between Gyro Reads
    b = time.time() - last_call_s
    LP = b

    #####
    ##### Apply low pass filter #####
    #####
    MAGx = MAGx * MAG_LPF_FACTOR + oldXMagRawValue*(1 - MAG_LPF_FACTOR);
    MAGy = MAGy * MAG_LPF_FACTOR + oldYMagRawValue*(1 - MAG_LPF_FACTOR);
    MAGz = MAGz * MAG_LPF_FACTOR + oldZMagRawValue*(1 - MAG_LPF_FACTOR);
    ACCx = ACCx * ACC_LPF_FACTOR + oldXAccRawValue*(1 - ACC_LPF_FACTOR);
    ACCy = ACCy * ACC_LPF_FACTOR + oldYAccRawValue*(1 - ACC_LPF_FACTOR);
    ACCz = ACCz * ACC_LPF_FACTOR + oldZAccRawValue*(1 - ACC_LPF_FACTOR);

    oldXMagRawValue = MAGx
    oldYMagRawValue = MAGy
    oldZMagRawValue = MAGz
    oldXAccRawValue = ACCx
    oldYAccRawValue = ACCy
    oldZAccRawValue = ACCz

    #####
    ##### Median filter for accelerometer #####
    #####
    # cycle the table
    for x in range (ACC_MEDIANTABLESIZE-1,0,-1 ):
        acc_medianTable1X[x] = acc_medianTable1X[x-1]
        acc_medianTable1Y[x] = acc_medianTable1Y[x-1]
        acc_medianTable1Z[x] = acc_medianTable1Z[x-1]

    # Insert the latest values
    acc_medianTable1X[0] = ACCx
    acc_medianTable1Y[0] = ACCy
    acc_medianTable1Z[0] = ACCz

```

```

# Copy the tables
acc_medianTable2X = acc_medianTable1X [:]
acc_medianTable2Y = acc_medianTable1Y [:]
acc_medianTable2Z = acc_medianTable1Z [:]

# Sort table 2
acc_medianTable2X.sort ()
acc_medianTable2Y.sort ()
acc_medianTable2Z.sort ()

# The middle value is the value we are interested in
ACCx = acc_medianTable2X [int (ACC_MEDIANTABLESIZE / 2)];
ACCy = acc_medianTable2Y [int (ACC_MEDIANTABLESIZE / 2)];
ACCz = acc_medianTable2Z [int (ACC_MEDIANTABLESIZE / 2)];

#####
#### Median filter for magnetometer ####
#####
# cycle the table
for x in range (MAG_MEDIANTABLESIZE-1,0,-1 ):
    mag_medianTable1X [x] = mag_medianTable1X [x-1]
    mag_medianTable1Y [x] = mag_medianTable1Y [x-1]
    mag_medianTable1Z [x] = mag_medianTable1Z [x-1]

# Insert the latest values
mag_medianTable1X [0] = MAGx
mag_medianTable1Y [0] = MAGy
mag_medianTable1Z [0] = MAGz

# Copy the tables
mag_medianTable2X = mag_medianTable1X [:]
mag_medianTable2Y = mag_medianTable1Y [:]
mag_medianTable2Z = mag_medianTable1Z [:]

# Sort table 2
mag_medianTable2X.sort ()
mag_medianTable2Y.sort ()
mag_medianTable2Z.sort ()

# The middle value is the value we are interested in
MAGx = mag_medianTable2X [int (MAG_MEDIANTABLESIZE / 2)];
MAGy = mag_medianTable2Y [int (MAG_MEDIANTABLESIZE / 2)];
MAGz = mag_medianTable2Z [int (MAG_MEDIANTABLESIZE / 2)];

#Convert Gyro raw to degrees per second
rate_gyr_x = GYRx * G_GAIN
rate_gyr_y = GYRy * G_GAIN
rate_gyr_z = GYRz * G_GAIN

#Calculate the angles from the gyro.
gyroXangle+=rate_gyr_x*LP
gyroYangle+=rate_gyr_y*LP
gyroZangle+=rate_gyr_z*LP

##Convert Accelerometer values to degrees
AccXangle = (math.atan2 (ACCy,ACCz)+M_PI)*RAD_TO_DEG
AccYangle = (math.atan2 (ACCz,ACCx)+M_PI)*RAD_TO_DEG

```

```

#Complementary filter used to combine the accelerometer and gyro values.
CFangleX=AA*(CFangleX+rate_gyr_x*LP) +(1 - AA) * AccXangle
CFangleY=AA*(CFangleY+rate_gyr_y*LP) +(1 - AA) * AccYangle

#Kalman filter used to combine the accelerometer and gyro values.
kalmanY = kalmanFilterY (AccYangle, rate_gyr_y,LP)
kalmanX = kalmanFilterX (AccXangle, rate_gyr_x,LP)

if IMU_upside_down :
    MAGy = -MAGy

#Calculate heading
heading = 180 * math.atan2 (MAGy,MAGx)/M_PI

#Only have our heading between 0 and 360
if heading < 0:
    heading += 360

#Normalize accelerometer raw values.
accXnorm = ACCx/math.sqrt (ACCx * ACCx + ACCy * ACCy + ACCz * ACCz)
accYnorm = ACCy/math.sqrt (ACCx * ACCx + ACCy * ACCy + ACCz * ACCz)

#Calculate pitch and roll
if IMU_upside_down :
    accXnorm = -accXnorm          #flip Xnorm as the IMU is upside down
    accYnorm = -accYnorm          #flip Ynorm as the IMU is upside down
    pitch = math.asin (accXnorm)
    roll = math.asin (accYnorm/math.cos (pitch))
else :
    pitch = math.asin (accXnorm)
    roll = -math.asin (accYnorm/math.cos (pitch))

#Calculate the new tilt compensated values
magXcomp = MAGx*math.cos (pitch)+MAGz*math.sin (pitch)
magYcomp = MAGx*math.sin (roll)*math.sin (pitch)+MAGy*math.cos (roll)-MAGz*math.sin (roll)*math.sin (pitch)

#Calculate tilt compensated heading
tiltCompensatedHeading = 180 * math.atan2 (magYcomp,magXcomp)/M_PI

#Only have our heading between 0 and 360
if tiltCompensatedHeading < 0:
    tiltCompensatedHeading += 360

return kalmanX, kalmanY

```

14.4.2. Inicialización

```
from imu_control import imu_control
from threading import Thread
from math import tan, atan
import RPi.GPIO as GPIO
import numpy as np
import pickle
import time
import cv2
import pdb

START_GPIO = 32

#X is along the dashboard
#Y is depth
#Z is height

class Control():
    def __init__(self):
        self.face_cascade = cv2.CascadeClassifier('/usr/share/opencv/haarcascades/haarcascade_profileface.xml')
        self.eye_cascade = cv2.CascadeClassifier('/usr/share/opencv/haarcascades/haarcascade_eye.xml')

        #Interrupt for functioning
        GPIO.setmode(GPIO.BOARD)
        GPIO.setup(START_GPIO, GPIO.IN, pull_up_down=GPIO.PUD_UP)

        self.light_gpio = 32
        GPIO.setup(self.light_gpio, GPIO.OUT)

        self.Motor1A = 36
        self.Motor1B = 38
        self.Motor1E = 40
        GPIO.setup(self.Motor1A, GPIO.OUT)
        GPIO.setup(self.Motor1B, GPIO.OUT)
        GPIO.setup(self.Motor1E, GPIO.OUT)
        self.disableHorizontal()

        self.Motor2A = 33
        self.Motor2B = 35
        self.Motor2E = 37
        GPIO.setup(self.Motor2A, GPIO.OUT)
        GPIO.setup(self.Motor2B, GPIO.OUT)
        GPIO.setup(self.Motor2E, GPIO.OUT)
        self.disableVertical()

        self.stopped = False
        Thread(target=self._update, args=()).start()

    def __del__(self):
        self.stopped = True
```

```

def _update(self):
    previous_call = time.time()
    while True:
        last_call = time.time()
        if self.stopped:
            return
        self._y_imu, self._z_imu = imu_control(previous_call)
        previous_call = last_call

def _read_imu(self):
    time.sleep(0.5)
    return self._y_imu, self._z_imu

def _load_calibration(self):
    try:
        f = open('calibration.pckl', 'rb')
        self.y_delta_deg_calib, self.z_delta_deg_calib, self.x_pos_calib,
        self.y_pos_calib, self.z_pos_calib = pickle.load(f)
        f.close()
        # self._move_to_setpoint(self.y_delta_deg_calib, self.z_delta_deg_calib)
    except:
        #indicar falta de calibracion
        pass

def left(self):
    GPIO.output(self.Motor1A,GPIO.HIGH)
    GPIO.output(self.Motor1B,GPIO.LOW)

def right(self):
    GPIO.output(self.Motor1A,GPIO.LOW)
    GPIO.output(self.Motor1B,GPIO.HIGH)

def enableHorizontal(self):
    self.disableVertical()
    GPIO.output(self.Motor1E,GPIO.HIGH)

def disableHorizontal(self):
    GPIO.output(self.Motor1E,GPIO.LOW)

def up(self):
    GPIO.output(self.Motor2A,GPIO.HIGH)
    GPIO.output(self.Motor2B,GPIO.LOW)

def down(self):
    GPIO.output(self.Motor2A,GPIO.LOW)
    GPIO.output(self.Motor2B,GPIO.HIGH)

def enableVertical(self):
    self.disableHorizontal()
    GPIO.output(self.Motor2E,GPIO.HIGH)

def disableVertical(self):
    GPIO.output(self.Motor2E,GPIO.LOW)

```

14.4.3. Homing

```
def _homing(self):
    max_yDeg, _ = self._movement_routine(direction='vertical', func=self.up)
    min_yDeg, _ = self._movement_routine(direction='vertical', func=self.down)
    yDeg_middle = (max_yDeg + min_yDeg) / 2
    self.up()
    self.enableVertical()
    yDeg_measured, _ = self._read_imu()
    while yDeg_measured < yDeg_middle:
        yDeg_measured, _ = self._read_imu()
    self.disableVertical()

    _, min_zDeg = self._movement_routine(direction='horizontal', func=self.left)
    _, max_zDeg = self._movement_routine(direction='horizontal', func=self.right)
    zDeg_middle = (max_zDeg + min_zDeg) / 2
    self.left()
    self.enableHorizontal()
    _, zDeg_measured = self._read_imu()
    while zDeg_measured < zDeg_middle:
        _, zDeg_measured = self._read_imu()
    self.disableHorizontal()

    self.ST_y_deg, self.ST_z_deg = self._read_imu()

def _movement_routine(self, direction, func):
    deg_tol = 0.2
    func()
    if direction == 'horizontal':
        self.enableHorizontal()
    else:
        self.enableVertical()
    new_yDeg, new_zDeg = self._read_imu()
    old_yDeg, old_zDeg = -new_yDeg, -new_zDeg
    while abs(new_yDeg - old_yDeg) > deg_tol or
        abs(new_zDeg - old_zDeg) > deg_tol:
        old_yDeg, old_zDeg = new_yDeg, new_zDeg
        new_yDeg, new_zDeg = self._read_imu()
    if direction == 'horizontal':
        self.disableHorizontal()
    else:
        self.disableVertical()
    return new_yDeg, new_zDeg
```

14.4.4. Movimiento

```
def _move_to_setpoint(self, y_delta_deg, z_delta_deg):
    init_y_deg, init_z_deg = self._read_imu()
    local_yDeg, local_zDeg = init_y_deg, init_z_deg
    self.up() if y_delta_deg > 0 else self.down()
    self.enableVertical()
    while abs(init_y_deg - local_yDeg) < abs(y_delta_deg):
        local_yDeg, local_zDeg = self._read_imu()
        print(local_yDeg, local_zDeg)
    self.disableVertical()

    self.right() if z_delta_deg > 0 else self.left()
    self.enableHorizontal()
    while abs(init_z_deg - local_zDeg) < abs(z_delta_deg):
        local_yDeg, local_zDeg = self._read_imu()
        print(local_yDeg, local_zDeg)
    self.disableHorizontal()
```

14.4.5. Toma de imagen

```
def _take_pic(self):
    cap = cv2.VideoCapture(0)
    ret, frame = cap.read()
    return cv2.cvtColor(frame, cv2.COLOR_RGB2GRAY)
```

14.4.6. Reconocimiento de centro de ojos

```
def get_face_center(self, img):
    # img = self._take_pic()
    face_coords = self._detect_face(img)
    face = self._get_face(img, face_coords)
    eyes_coords = self._detect_eyes(face)
    self.y_center_pos, self.z_center_pos = self._get_face_center(face_coords,
                                                                eyes_coords)

def _detect_face(self, img):
    start = time.time()
    faces = ()
    rescale_factor = 6
    while len(faces) != 1 and rescale_factor != 0:
        hood = 0
        while len(faces) != 1 and hood < 10:
            faces = self.face_cascade.detectMultiScale(img, 1+rescale_factor/10,
                                                         hood, minSize=(200,200),
                                                         maxSize=(400,400))

            hood += 1
        if rescale_factor > 1:
            rescale_factor -= 1
        else:
            rescale_factor -= 0.1
    print(time.time()-start)
    return faces

def _get_face(self, img, coords):
    for (y,z,w,h) in coords:
        roi = img[z:z+int(h/2), y:y+w]
    return roi

def _detect_eyes(self, roi):
    start = time.time()
    eyes = ()
    rescale_factor = 6
    while len(eyes) != 2 and rescale_factor != 0:
        hood = 0
        while len(eyes) != 2 and hood != 10:
            eyes = self.eye_cascade.detectMultiScale(roi, 1 + rescale_factor/10,
                                                         hood, minSize=(20,20),
                                                         maxSize=(70,70))

            hood += 1
        if rescale_factor > 1:
            rescale_factor -= 1
        else:
            rescale_factor -= 0.1
    print(time.time()-start)
    return eyes

def _get_face_center(self, face_coords, eyes_coords):
    y_center_pos = face_coords[0][0] +
                    int(((2*eyes_coords[0][0] + eyes_coords[0][2])/2 +
                        (2*eyes_coords[1][0] + eyes_coords[1][2])/2)/2)
    z_center_pos = face_coords[0][1] +
                    int(((2*eyes_coords[0][1] + eyes_coords[0][3])/2 +
                        (2*eyes_coords[1][1] + eyes_coords[1][3])/2)/2)
    return y_center_pos, z_center_pos
```

14.4.7. Cálculo de desplazamiento

```
def calc_y_axis(self, y):
    return 0.5 * ( atan(self.x_pos_calib / y) -
                 atan(self.x_pos_calib / self.y_pos_calib) ) * 180 / np.pi

def calc_z_axis(self, y, z):
    return 0.5 * ( atan(z / y) -
                 atan(self.z_pos_calib / self.y_pos_calib) ) * 180 / np.pi
```

14.4.8. Calibración

```
def _calibrate(self, img1, img2):
    self.get_face_center(img1)
    self.y_delta_deg_calib, self.z_delta_deg_calib = self._read_imu()

    self.z_delta_deg_calib -= self.ST_z_deg
    self.y_delta_deg_calib -= self.ST_y_deg

    self.y_pos_calib = self.y_center_pos
    self.z_pos_calib = self.z_center_pos

    self.get_face_center(img2)
    img2_y_deg, _ = self._read_imu()
    img2_y_pos = self.y_center_pos

    self.x_pos_calib = self._xCalibration(self.y_delta_deg_calib,
                                         self.y_pos_calib,
                                         img2_y_deg, img2_y_pos)
```

14.4.9. Detección de luminosidad

```
def check_luminosity(self, img):
    desiredMeanValue = 128.0
    if img.mean() < desiredMeanValue:
        self._illuminate()

def _illuminate(self):
    GPIO.output(self.light_gpio, GPIO.HIGH)
```

14.5. Hojas de Datos de Componentes

14.5.1. Microcontrolador

<http://vonger.cn/misc/vocore2/datasheet.zip>

14.5.2. IMU

<https://cdn-shop.adafruit.com/datasheets/LSM9DS0.pdf>

14.5.3. Driver

<http://www.ti.com/lit/ds/symlink/l293.pdf>

14.5.4. Fuente

<http://www.ti.com/lit/ds/symlink/lm2596.pdf>

14.6. Bitácora

A continuación se detalla el proceso que llevó al producto final de forma cronológica.

Selección de proyecto

La idea original del proyecto se planteó como un lazo cerrado de control sobre los espejos laterales y retrovisor del automóvil, los cuales estuvieran en constante modificación para garantizar el óptimo confort del conductor.

Selección de prototipo

Para poder comenzar con el estudio del proyecto se debía investigar aquel sistema que funcione como base del mismo y tenga suficiente versatilidad como para utilizar de forma sencilla diversos lenguajes de programación hasta seleccionar el más indicado para ejecutar el programa final, tenga facilidad con el uso de cámaras, permita resolver de forma eficiente algoritmos de procesamiento de imágenes y cuyo costo (por ser solamente un prototipo) no sea elevado.

Por todo ello es que se decidió partir desde una *RaspberryPi* modelo 2B+.

Otras opciones fueron consideradas y descartadas como:

1. Arduino UNO. Descartado por ser utilizado con lenguaje de programación Arduino como base y necesitar demasiada configuración para funcionar del modo deseado.
2. *BeagleBone Black*. Descartado por presentar una leve mejora frente a la *Raspberry* a un costo más elevado y carecer de un soporte tan grande como el de esta.
3. FPGA. Descartado por tener un costo demasiado elevado a costa de ejecutar operación más rápido y ser programa en VHDL.

Selección del equipo

Para poder constituir el sistema más preciso de reconocimiento de posición del conductor se intentó desarrollar el sistema de reconocimiento del ser humano mediante dos imágenes. Por ello, la primer idea fue la de utilizar dos cámaras (una en cada parante delantero del automóvil) y reconstruir el espacio de la cabina.

Selección de algoritmo

Siendo que reconstruir todo el espacio de la cabina carecía de sentido se optó por un nuevo enfoque. Mediante las sombras proyectadas, poder estimar la posición del conductor.

A su vez, con la cámara del lado del conductor se detectarían las pupilas de este.

Selección de proyecto

Debido a la complejidad y a través de una encuesta realizada que relevó que el conductor se sentiría incómodo con los espejos en constante movimiento, el fin del proyecto fue modificado al punto de mover los espejos solo cuando el usuario lo requiriese.

Selección de equipo

En base a todos los requerimientos que serían necesarios acoplar para que el sistema sea infalible (como intentar generar siempre el mismo grado de luminosidad, mismo punto de generación de luz, ignorar fuentes de luz externas), es que se decidió modificar la constitución del proyecto para que solo requiriese 1 cámara del lado del conductor para identificar cambios en la altura del sujeto, y un sensor en la corredera del asiento para identificar cambios en la profundidad del sujeto.

Selección del equipo

Detectar la posición del asiento genera una gran incertidumbre respecto de la posición del conductor. Por ejemplo, la persona puede conducir reclinada, encorvada, puede cambiar su posición a lo largo del camino, etc. Debido a ello, un sensor IR que apunte al cabezal fue la opción más sencilla y menos invasiva hacia el sujeto. De este modo, detectar la profundidad posee una medición más robusta.

Selección de equipo y algoritmo

Pese a lo dicho anteriormente, seleccionar un punto donde el sensor de profundidad IR nunca tuviese oclusiones resultó imposible. Además, el único ángulo que garantiza apuntar siempre al cabezal del respaldo es de frente.

Dicho esto, se comprobó a su vez que medir posición con la cámara predecía movimiento vertical y en profundidad. Por ello el equipo fue finalmente construido por tan solo una cámara.

Selección de equipo

Para empezar, se intentó considerar al espejo lateral como un sistema cerrado y sin acceso. Por ello, se pensó en calibrar movimiento en un eje del espejo por un lapso de tiempo a una tensión y corriente fija.

Selección de equipo

El enfoque anterior resultó descartado casi de forma inmediata ya que las mediciones resultaron inconclusas y una calibración de forma tan sencilla nunca pudo ser lograda. Por ende, se procedió a generar un lazo cerrado de control sobre el espejo, ello es, acceder al espejo, agregar un subsistema adicional y asegurarse la correcta utilización.

El subsistema de más fácil lectura en cuanto a variaciones que permite una sencilla conectividad con la *Raspberry* resultó el de un potenciómetro.

Selección de equipo

Por la disponibilidad del espacio dentro del espejo es que se descartó la utilización de un potenciómetro. La finalidad de este proyecto no tiene como objetivo, además, modificar las auto-partes. Es decir, modificar la estructura que contiene los espejos.

El nuevo subsistema debería permitir una fácil colocación, una lectura de variaciones de posición precisa y debía ser lo suficientemente pequeño como para *ser introducido* sin modificar la estructura. El subsistema que cumple con todos estos requisitos es una IMU.

Mediante una correcta calibración inicial, la IMU permita establecer con 9 grados de libertad la orientación y movimiento de un objeto sobre la cual es colocado.

Selección de algoritmo

Originalmente se pensó en grabar de forma continua las mediciones de la IMU en cuanto a sus 9 ejes, de esta forma, incluso con el vehículo apagado, siempre se podría estimar la última posición de uso.

Esto resultó ser falso ya que al ser apagado se pierden 3 grados de libertad del giróscopo, el acelerómetro no interviene e intentar reconstruir la última medición requiere de un movimiento en los 2 planos.

Por ellos se decidió realizar una rutina de homing únicamente cuando se enciende el auto y comienza la utilización del dispositivo. La rutina puede tardar unos pocos segundos, da tiempo a inicializar los módulos del programa y garantiza menos error en la estimación del movimiento del espejo al llevar su posición a la de calibración y no acumular errores de uso.

Selección de equipo

Naturalmente no se puede esperar que un GPIO pueda controlar un motor de corriente continua, ya que la tensión de salida es de 3v3 y la máxima corriente de unos pocos amperes. Dicho es, se investigó que subsistema convendría integrar para poder controlar los espejos desde los GPIOs.

Por cumplir con las especificaciones del motor en cuánto a consumo y por utilizado en la gran mayoría de proyectos relacionados con GPIOs y motores es que se decidió incorporar un módulo L293D.

Factibilidad económica

Al considerar la mano de obra se tuvo que recalcular el valor de venta del producto para que el proyecto siga siendo viable en los tiempos estimados. Esto significó un aumento insignificante a visto del lado de una automotriz.

14.6.1. Bitácora de software

14.6.1.1. Semana 1: IMU: Reconocimiento por puerto I2C y lectura desde *Python*. Error: Habilitar dos puertos I2C.

Reconocimiento de rostro: Código en *Python* para reconocimiento. Todo error solucionado.

14.6.1.2. Semana 2: IMU: Todo error solucionado.

Reconocimiento de rostro: Todo error solucionado.

Reconocimiento de ojos: Código en *Python* para reconocimiento. Errores: No siempre halla dos ojos, en la posición correcta y el tiempo de resolución es excesivo.

14.6.1.3. Semana 3: IMU: Rutina de *Python*. Errores: Dispersión de valores muy grandes, los filtros no suelen funcionar correctamente, no se decide qué parámetros utilizar (valores crudos del sensor, *tip/tilt*, *heading*, etc.)

Reconocimiento de rostro: Se halló un caso donde el rostro es muy grande y no convergió a una solución.

Reconocimiento de ojos: Se mejoraron las condiciones iniciales de búsqueda pero no siempre converge correctamente.

14.6.1.4. Semana 4: IMU: Se decidió implementar un filtro de Kalman para los valores de los giróscopos. El resultado y resolución son aceptables.

Reconocimiento de rostro: El tiempo de convergencia es excesivo.

Reconocimiento de ojos: El tiempo de convergencia es excesivo.

14.6.1.5. Semana 5: IMU: Comportamiento aceptable.

Reconocimiento de rostro: Se limitó el tamaño del rasgo a detectar. Comportamiento robusto y veloz.

Reconocimiento de ojos: Comportamiento robusto y veloz.

Rutina de homing: Muchos inconvenientes acerca de cómo encontrar los límites de carrera hasta que finalmente se encontró la solución.

Control de movimiento: Problemas en la lectura de los valores, signos en el movimiento, robustez en las rutinas para toda contingencia. Todo error solucionado.

Referencias

- [1] Yi-Qing Wang, *An Analysis of the Viola-Jones Face Detection Algorithm*, Image Processing On Line, 2014, <http://dx.doi.org/10.5201/ipol.2014.104>
- [2] Instituto Nacional de Tecnología Industrial <https://www.inti.gob.ar/certificaciones/c-seguridadElectrica.html>
- [3] Solicitud de certificación INTI https://www.inti.gob.ar/certificaciones/pdf/FCSolicitud_de_Certificacion.doc
- [4] Instituto Nacional de la Propiedad Industrial <http://www.inpi.gov.ar/index.php?Id=107>
- [5] Código aduanero <http://servicios.infoleg.gob.ar/infolegInternet/anexos/110000-114999/110121/norma.htm>
- [6] OpenCV, cascadedetect, <https://github.com/opencv/opencv/blob/master/modules/objdetect/>
- [7] <http://www.ti.com/lit/an/szza0/szza009.pdf>
- [8] <http://www.olliw.eu/2013/imu-data-fusing/>
- [9] <http://www.ipc.org/TOC/IPC-A-600F.pdf>
- [10] <http://www.ipc.org/TOC/IPC-A-610F.pdf>
- [11] <http://tf.boulder.nist.gov/general/pdf/1133.pdf>
- [12] <http://www.afac.org.ar/paginas/noticia.php?id=1609>
- [13] <http://autoblog.com.ar/wp-content/uploads/2015/01/Patentamientos-totales-2014.pdf>
- [14] <https://www.ineltro.ch/media/downloads/SAAItem/45/45958/36e3e7f3-2049-4adb-a2a7-79c654d92915.pdf>
- [15] https://en.wikipedia.org/wiki/Bang%E2%80%93bang_control
- [16] <https://es.wikipedia.org/wiki/Estatura>
- [17] <https://www.oecd.org/dev/Impuestos-sobre-los-salarios-en-America-Latina-y-el-Caribe-RESUMEN.pdf>
- [18] <http://argentina.workingdays.org/>
- [19] https://en.wikipedia.org/wiki/Pinhole_camera_model