



Powered by
Arizona State University

INGENIERÍA AUTOMOTRIZ

**Proyecto previo a la Obtención del Título de Ingeniero
Automotriz**

Autor: Enrique Ricardo Salazar Yépez

Tutor: Ing. Alex Fernando Llerena Mena

**Desarrollo de Prototipo de Asistencia de Cambio de Carril para
Conductores de Vehículos Tipo M1.**

Certificado de Autoría

Yo, Enrique Ricardo Salazar Yépez, declaro bajo juramento, que el trabajo aquí descrito es de mi autoría; que no ha sido presentado anteriormente para ningún grado o calificación profesional y que se ha consultado la bibliografía detallada. Cedo mis derechos de propiedad intelectual a la Universidad Internacional del Ecuador, para que sea publicado y divulgado en internet, según lo establecido en la Ley de Propiedad Intelectual, reglamento y leyes.

Enrique Ricardo Salazar Yépez

C.I. 0953341518

Aprobación del Tutor

Yo, Alex Fernando Llerena Mena certifico que conozco al autor del presente trabajo siendo responsable exclusivo tanto de su originalidad y autenticidad, como de su contenido.

Ing. Alex Fernando Llerena Mena, MSc.

Director de Proyecto

Dedicatoria

El presente proyecto lo dedico con mucho amor a mi hija Camila y a mi esposa Carla quienes han sido el motor que me ha motivado para seguir adelante en todo aquello que me proponga, a mi padre por ser guía y modelo a seguir, a mi madre por enseñarme a ser mejor hombre, mejor padre y sobre todo a mi Dios que, sin él, nada de esto existiría.

Salazar Yépez Enrique Ricardo

Agradecimientos

Mi agradecimiento a todos los docentes de la carrera de Ingeniería Automotriz, en especial a mi tutor Ing. Fernando Llerena Mena, por sus sabios consejos y acertada dirección.

También agradezco a todas las personas que colaboraron en la investigación.

Muchas gracias

Índice General

Certificado de Autoría.....	iii
Aprobación del Tutor.....	iv
Dedicatoria.....	v
Agradecimientos	vi
Índice General.....	vii
Índice de Figuras.....	xi
Índice de Tablas	xiii
Resumen.....	xiv
Abstract	xv
Capítulo I.	1
Antecedentes	1
1.1 Título de Investigación.....	1
1.2 Planteamiento, Formulación y Sistematización del Problema.....	1
1.2.1 <i>Planteamiento del Problema</i>	1
1.2.2 <i>Formulación del Problema</i>	2
1.2.3 <i>Sistematización del Problema</i>	2
1.3 Objetivos de la Investigación	2
1.3.1 <i>Objetivo General</i>	2
1.3.2 <i>Objetivos Específicos</i>	2
1.4 Justificación y Delimitación de la Investigación	2
1.4.1 <i>Justificación Teórica</i>	2
1.4.2 <i>Justificación Práctica</i>	3
1.4.3 <i>Justificación Metodológica</i>	4
1.4.4 <i>Delimitación Temporal</i>	4

1.4.5	<i>Delimitación Geográfica</i>	4
1.4.6	<i>Delimitación del Contenido</i>	4
	Capítulo II.	5
	Marco Teórico.....	5
2.1	Vehículos con Asistencia de Conducción de Nivel 1	5
2.2	Raspberry Pi.....	6
2.3	Accidentes de Tránsito.....	7
2.4	Asistencia al Conductor	8
2.4.1	<i>Asistencia de Mantenimiento de Carril</i>	9
2.4.2	<i>Control de Crucero Adaptativo</i>	9
2.4.3	<i>Frenado de Emergencia Automático</i>	9
2.4.4	<i>Alerta de Colisión</i>	9
2.4.5	<i>Asistencia de Estacionamiento</i>	9
2.5	Automóviles con Asistencia al Conductor.....	10
2.5.1	<i>Tesla</i>	12
2.5.2	<i>BMW</i>	12
2.5.3	<i>Mercedes-Benz</i>	12
2.5.4	<i>Audi</i>	13
2.5.5	<i>Volvo</i>	13
2.6	Lenguajes de Programación	14
2.6.1	<i>Python</i>	14
2.6.2	<i>C</i>	15
2.6.3	<i>C++</i>	15
2.6.4	<i>Java</i>	15
2.6.5	<i>JavaScript</i>	15

2.6.6	<i>Scratch</i>	15
	Capítulo III.....	16
	Implementación del Prototipo de Asistencia al Conductor.....	16
3.1	Metodología para la Implementación.....	16
3.2	Componentes del Prototipo	17
3.2.1	<i>Cámara</i>	17
3.2.2	<i>Altavoces</i>	19
3.2.3	<i>Pantalla</i>	21
3.2.4	<i>Microcontrolador</i>	23
3.2.5	<i>Fuente de Energía</i>	25
3.2.6	<i>Mouse y Teclado</i>	26
3.3	Elementos de Gasto.....	27
3.4	Desarrollo del Algoritmo	28
3.4.1	<i>Librerías Utilizadas en la Elaboración del Código</i>	28
3.4.2	<i>Open Source Computer Vision Library</i>	29
3.4.3	<i>NumPy</i>	29
3.4.4	<i>Pygame</i>	30
3.4.5	<i>Time</i>	30
3.4.6	<i>Math</i>	31
3.4.7	<i>Diagrama de Flujo del Funcionamiento del Algoritmo Implementado</i>	32
3.5	Esquema de Conexión de los Componentes Seleccionados	32
3.6	Implementación del Prototipo	34
	Capitulo IV.....	39
	Pruebas y Resultados de la Implementación.....	39
4.1	Pruebas del Prototipo.	40

4.1.1	<i>Pruebas de Campo</i>	40
4.1.2	<i>Precisión de Detección y Alertas</i>	45
4.1.3	<i>Tiempo de Respuesta del Prototipo</i>	46
4.1.4	<i>Falsos Positivos y Falsos Negativos</i>	47
4.2	Resultados Obtenidos de la Implementación del Prototipo	48
	Conclusiones	50
	Recomendaciones	51
	Bibliografía	52
	Anexos	53

Índice de Figuras

Figura 1	<i>La Mini Pc Mas Famosa del Mundo</i>	7
Figura 2	<i>Principales Causas de Muertes por Accidentes de Transito</i>	8
Figura 3	<i>Sistemas de Asistencia al Conductor en Carros Nuevos</i>	10
Figura 4	<i>Vehículos Autónomos</i>	13
Figura 5	<i>Cámara Web</i>	18
Figura 6	<i>Altavoces</i>	20
Figura 7	<i>Pantalla</i>	22
Figura 8	<i>Raspberry Pi 4</i>	24
Figura 9	<i>Fuente de Alimentación</i>	26
Figura 10	<i>Teclado con Touchpad</i>	27
Figura 11	<i>Esquema de Conexión</i>	32
Figura 12	<i>Instalación Cámara Web</i>	35
Figura 13	<i>Instalación Cámara Web Vista Frontal</i>	35
Figura 14	<i>Instalación Pantalla y Raspberry</i>	36
Figura 15	<i>Instalación Pantalla y Raspberry vista posterior</i>	36
Figura 16	<i>Instalación Altavoces</i>	37
Figura 17	<i>Fuente de Alimentación</i>	37
Figura 18	<i>Fuente de Alimentación Instalada</i>	38
Figura 19	<i>Instalación Componentes Raspberry</i>	38
Figura 20	<i>Ruta Vial 10 La joya – La Rioja</i>	40
Figura 21	<i>Trayecto de Ruta (Reconocimiento de Líneas Guías)</i>	41
Figura 22	<i>Trayecto de Ruta (Reconocimiento de Líneas Guías y Línea Continua)</i>	42
Figura 23	<i>Trayecto de Ruta (Intento de Descarrilamiento)</i>	42
Figura 24	<i>Trayecto de Ruta (Prototipo Emite Alerta)</i>	43

Figura 25	<i>Trayecto de Ruta (Reincorporación a Carril, Deja de Emitir Pitido)</i>	43
Figura 26	<i>Prueba de Ruta (Intento 2 de Descarrilamiento)</i>	44
Figura 27	<i>Prueba de Ruta (Momento Exacto Donde Emite la Alerta)</i>	44
Figura 28	<i>Prueba de Ruta (Reincorporación de Carril, Deja de Emitir Alerta Audible)</i>	

Índice de Tablas

Tabla 1	<i>Comparativa de Cámaras Web</i>	18
Tabla 2	<i>Comparativa de Altavoces</i>	20
Tabla 3	<i>Comparativa de Pantallas</i>	21
Tabla 4	<i>Comparativa de Raspberry</i>	23
Tabla 5	<i>Resumen del Presupuesto</i>	27
Tabla 6	<i>Resultados de las Pruebas sobre Detección y Alertas</i>	45
Tabla 7	<i>Resultados de Tiempo de Respuesta del Prototipo</i>	46
Tabla 8	<i>Resultados de Pruebas de Falsos Positivos y Falsos Negativos</i>	47

Resumen

Este proyecto tiene como objetivo principal la implementación de un prototipo de asistencia vehicular de cruce de carril diseñado específicamente para tipos de vehículo M1. A través de un enfoque técnico integral, se busca mejorar la seguridad y la eficiencia en situaciones de cambio de carril, reduciendo el riesgo de colisiones y brindando una experiencia de conducción más segura y confiable.

La metodología se basa en una cuidadosa selección de componentes, considerando aspectos como resolución de cámaras, potencia de procesamiento y capacidad de detección de colisiones. La integración de estos componentes se realiza de manera estratégica, aprovechando la potencia de una Raspberry Pi para ejecutar algoritmos de detección y alerta en tiempo real.

El desarrollo del proyecto abarca desde la creación del algoritmo fundamental hasta las pruebas prácticas en condiciones de tráfico reales. Se ha dado prioridad a la precisión de la detección, la efectividad de las alertas y la adaptabilidad del sistema a diversas situaciones, incluidas variaciones en la iluminación y cambios de carril bruscos.

El resultado es un prototipo que supera los retos técnicos y funcionales, demostrando un alto grado de precisión en la detección de carriles y vehículos, así como en la generación de alertas en situaciones de colisión potencial. La capacidad del sistema para integrarse de manera exitosa en diferentes tipos de vehículos refuerza su utilidad y versatilidad en el mundo real.

Palabras Clave: Asistente vehicular, Vehículos tipo M1, Algoritmo, Raspberry

Abstract

The main objective of this project is to implement a prototype for cross lane vehicle assistance specifically designed for M1 vehicle types. Through a comprehensive technical approach, the aim is to enhance safety and efficiency in lane-changing situations, reducing the risk of collisions and providing a safer and more reliable driving experience.

The methodology is based on a careful selection of components, considering aspects such as camera resolution, processing power, and collision detection capabilities. The integration of these components is strategically executed, harnessing the power of a Raspberry Pi to run real-time detection and alert algorithms.

The project's development encompasses everything from the creation of the fundamental algorithm to practical testing under real traffic conditions. Emphasis has been placed on the accuracy of detection, effectiveness of alerts, and adaptability of the system to various scenarios, including variations in lighting and abrupt lane changes.

The result is a prototype that overcomes technical and functional challenges, demonstrating a high degree of accuracy in lane and vehicle detection, as well as in generating alerts in potential collision situations. The system's ability to seamlessly integrate into different vehicle types reinforces its utility and versatility in the real world.

Keywords: Vehicle assistance, M1-type vehicles, Algorithm, Raspberry Pi

Capítulo I.

Antecedentes

1.1 Título de Investigación

Desarrollo de Prototipo de Asistencia de Cambio de Carril para Conductores de Vehículos Tipo M1.

1.2 Planteamiento, Formulación y Sistematización del Problema

1.2.1 Planteamiento del Problema

En nuestro país, existen numerosos accidentes vehiculares que ponen en riesgo la vida del conductor y sus pasajeros. Una de las causas, es debido a la falta de concentración en situaciones de conducción monótonas, lo que conlleva, a un posible abandono de carril. Motivo por el cual, se propone implementar un sistema que brinde asistencia al conductor durante su largo trayecto.

Existen cifras que superan los 21 mil siniestros de tránsito y 1 mil heridos en las carreteras ecuatorianas (ANT, 2022). Según la plataforma Seat (Sociedad Española de Automóviles de Turismo), si los coches contaran con un programa especializado en mantenimiento de carriles, estas cifras se reducirían a la mitad.

Empresas como Audi y Volkswagen, tienen vehículos de alta gama que poseen el “Lane Active Assist” como parte integrada del automotor, lo que significa un gran avance en cuanto a la tecnología vehicular; Un sistema Cross Lane Assist sirve para brindar asistencia vehicular nivel 1, en el cual, mediante sensores, monitorea las líneas guías de la carretera; Si el conductor pierde su carril este sistema indicará mediante una alerta visual que se ha desviado del mismo y que debe de corregir su trayecto para evitar posibles accidentes.

Ecuador, debe de tener entidades que se dediquen a la elaboración de sistemas de asistencia vehicular, con el fin de reducir accidentes y muertes que se registran cada año en las vías. Por ello, es importante traer este tipo de tecnología, para así impulsar un nuevo concepto

de soporte al conductor durante su trayecto. El país aspira prevenir de manera significativa los siniestros de tránsito hasta el 2030. (Machado, 2022)

1.2.2 *Formulación del Problema*

¿El desarrollo de este prototipo de asistencia para conductores de vehículos tipo M1, podrá asistir al conductor y prevenir algún siniestro de tránsito?

1.2.3 *Sistematización del Problema*

- ¿Cuáles son los beneficios al desarrollar un sistema de asistencia de cambio de carril?
- ¿Cómo es el funcionamiento del sistema de asistencia de cruce de carril vehicular?
- ¿Qué pruebas se pueden realizar para determinar el buen funcionamiento del prototipo?

1.3 *Objetivos de la Investigación*

1.3.1 *Objetivo General*

Implementar un prototipo de asistencia vehicular de cruce de carril para tipos de vehículo M1.

1.3.2 *Objetivos Específicos*

- Seleccionar los componentes requeridos en base a parámetros de diseño.
- Desarrollar el algoritmo que permita el funcionamiento del sistema de asistencia de cruce de carril en una Raspberry pi.
- Verificar el funcionamiento del sistema de asistencia vehicular mediante pruebas de campo.

1.4 *Justificación y Delimitación de la Investigación*

1.4.1 *Justificación Teórica*

La seguridad vial es una preocupación primordial en la sociedad actual. La implementación de sistemas de asistencia a la conducción de un vehículo se alinea con la teoría

de prevención de accidentes al proporcionar a los conductores alertas y herramientas para prevenir colisiones. La aplicación de estas tecnologías tiene el potencial de reducir drásticamente los accidentes y, por ende, salvar vidas y disminuir lesiones graves.

La teoría de la conducción autónoma aboga por la progresiva automatización de tareas de conducción. El proyecto se alinea con esta teoría al desarrollar un sistema que introduce una forma limitada de automatización al asistir al conductor en una tarea específica: el cruce de carril. Esto representa un paso hacia la implementación gradual de tecnologías autónomas.

La teoría de la legislación vial resalta la importancia de la implementación de tecnologías que cumplan con las normativas de seguridad establecidas por las autoridades reguladoras. El proyecto contribuye a cumplir con estas normativas al ofrecer una solución innovadora que aborda preocupaciones de seguridad vial.

1.4.2 Justificación Práctica

La práctica de la conducción segura es esencial para prevenir accidentes y mantener la seguridad en carretera. Este proyecto busca proporcionar a los conductores una herramienta práctica que los ayude a tomar decisiones más informadas y seguras al cambiar de carril, evitando maniobras riesgosas.

La implementación de la asistencia de cruce de carril puede mejorar significativamente la experiencia de conducción de los usuarios. Al proporcionar alertas claras y útiles, los conductores experimentarán un mayor nivel de comodidad y confianza al manejar, lo que puede influir positivamente en la percepción general del vehículo.

En un contexto práctico, la implementación de sistemas de asistencia de cruce de carril representa un avance en la tecnología automotriz y una contribución a la innovación en la industria. Esto puede tener un impacto positivo en la reputación de los fabricantes de vehículos que priorizan la seguridad y la vanguardia tecnológica.

1.4.3 Justificación Metodológica

Se basa en la necesidad de abordar los desafíos técnicos de manera organizada, optimizar los recursos, garantizar la calidad y eficacia del proyecto en todas sus etapas. La elección de una metodología adecuada contribuirá al éxito y la viabilidad de la implementación del prototipo de asistencia vehicular de cruce de carril.

1.4.4 Delimitación Temporal

El trabajo se desarrollará desde el mes de Mayo del 2023, hasta Agosto del 2023, lapso que permitirá realizar la implementación del prototipo y sus pruebas de campo.

1.4.5 Delimitación Geográfica

El trabajo se desarrollará dentro de la ciudad de Guayaquil y su implementación se llevará a cabo en la Urbanización La Joya.

1.4.6 Delimitación del Contenido

En la primera fase, se realizará el análisis para seleccionar el controlador apropiado.

En la segunda fase, se determinará los componentes de este prototipo para poder brindar la asistencia de cruce de carril junto con el algoritmo de control.

En la tercera fase, se procederá a implementar el prototipo para luego realizar las pruebas en un vehículo tipo M1.

Capítulo II.

Marco Teórico

2.1 Vehículos con Asistencia de Conducción de Nivel 1

Los vehículos de nivel 1 son aquellos que poseen determinadas características como controles de velocidad, mantenimiento de carril. Este nivel precisamente está enfocado en la comodidad y seguridad de los conductores durante su trayecto en carreteras. (García, 2016).

Los vehículos tipo M1 se refieren a los vehículos de pasajeros diseñados y construidos para el transporte de personas. Esta categoría incluye automóviles de pasajeros, vehículos utilitarios deportivos (SUV), furgonetas y minibuses. Los vehículos tipo M1 están diseñados para transportar a menos de 9 personas, incluido el conductor, y están destinados principalmente al transporte privado y comercial de pasajeros.

Estos vehículos cumplen con una serie de regulaciones y estándares de seguridad establecidos por diferentes autoridades y organismos de cada país. Estos requisitos incluyen características como cinturones de seguridad, bolsas de aire, sistemas de frenos antibloqueo (ABS), control de estabilidad y otras medidas de seguridad diseñadas para proteger a los ocupantes en caso de accidente.

En resumen, los vehículos tipo M1 son aquellos diseñados para el transporte de pasajeros y se refieren a automóviles, SUV, furgonetas y minibuses destinados a transportar a menos de 9 personas, incluido el conductor.

Este tipo de automóviles puede tener un sistema o máximo dos que controlen la dirección o velocidad, pero no ambos al mismo tiempo. Por ejemplo, algunos cuentan con piloto automático, el cual se encarga de automáticamente frenar o acelerar de acuerdo con el tráfico, y otros, cuentan con un sistema de control de carril, evitando que un vehículo se salga de su carril. (Galván, 2017).

2.2 Raspberry Pi

Un Raspberry Pi, es un ordenador de bolsillo, del tamaño de una tarjeta de crédito. Consta de una placa con varios componentes de una computadora, como ARM, chip gráfico, y memoria RAM como se visualiza en la figura 1. Además de poseer puertos, entradas que facilitan la instalación de dispositivos periféricos; Algunos modelos brindan conexión mediante bluetooth o wifi y permiten instalar sistemas operativos libres a través de una microSD.

Raspberry Pi Cam. Este módulo de cámara de 8 Mp es capaz de capturar video de 1080 p e imágenes; se puede conectar directamente a la Raspberry Pi, como se observa en la figura 6. Compatible con el sistema operativo Raspbian listo para conectar y usar, muy adecuado para fotografiar por lapsos, grabar video o para usarlo en aplicaciones de seguridad y en detección de movimientos. Solo hay que conectar el cable plano incluido al puerto CSI Camera Serial Interface- de la Raspberry Pi. (Novillo Regalado, 2021)

Este ordenador permite varias aplicaciones al usuario, entre esas se encuentra navegar en internet mediante el sistema office, crear un centro multimedia utilizando su puerto HDMI, convertir un televisor en un Smart Tv, guardar archivos de manera segura en una nube, instalar un emulador de videojuegos clásicos, conectar motores, pantallas LED y cámaras de vigilancia.

El Raspberry Pi, no solo se lo puede utilizar con fines laborales, sino que también puede ser de uso doméstico; Como medidor de calidad de aire, que indica si existe humedad en el ambiente o brinda datos en cuanto a la concentración de partículas; Como sistema de riego para plantas, utilizando sensores de humedad conectados a una bomba; Como medidor de madurez de frutas y verduras a través de un sensor de luz que detecta el color de los alimentos; Como sistema central para convertir una casa convencional en una inteligente.

En fin, son un sin número de funciones útiles que tiene este dispositivo que brinda la facilidad de emplearlo en diversos campos. Sin embargo, las limitaciones existen incluso con

este ordenador, pues su velocidad no es extremadamente rápida, al contrario, posee una velocidad similar a la de uno de los primeros ordenadores existentes. Motivo por el cual, se debe considerar dos veces si se lo va a utilizar para navegar en red, cargar videojuegos. Además de la limitación mencionada, el ordenador no consta de un sistema de prendido o apagado y pese a que tiene un consumo bajo, debe ser conectado a la corriente. (Calvo, 2022).

Figura 1

La Mini Pc Mas Famosa del Mundo



Fuente: Eloutput, 2022.

2.3 Accidentes de Tránsito

El INEC menciona que los accidentes de tránsito pertenecen a las diez causas principales de muerte en el país. (Machado, 2022)

Según el banco mundial, las muertes ocasionadas por accidentes de tránsito en Ecuador son de 20 decesos cada 100 mil habitantes. Es decir, el país se encuentra ubicado en la quinta posición de tasas de mortalidad más altas, en vías, en todo América del Sur. Sin embargo, el director de la ANT, Adrián Castro, indica que, los siniestros de tránsito son prevenibles, debido a que se relacionan con irrespetar leyes establecidas de conducción.

Juan Carlos de la Hoz, representante del banco mundial, indica que, en Sudamérica, 5

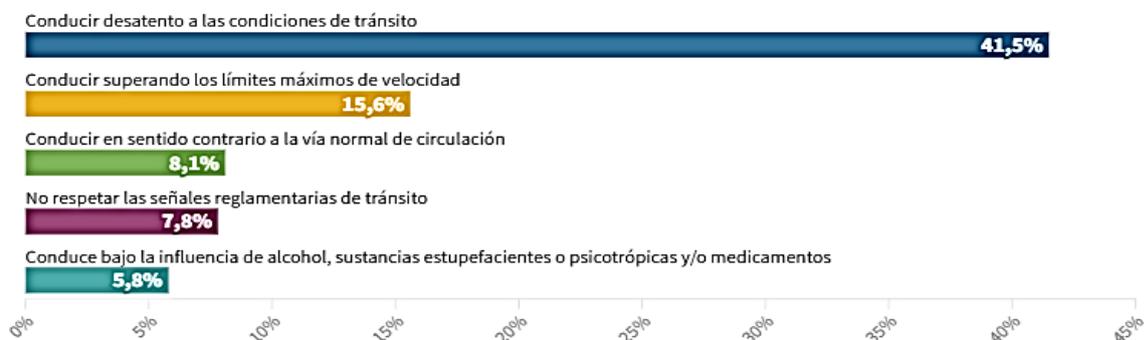
millones de personas necesitan atención médica a causa de accidentes automovilísticos; Por lo que, apoyan la Es-Segura 2022-2023, con el fin de reducir los siniestros, evitar muertes y al mismo tiempo ahorrar dinero de acuerdo con la vida que se le asigna a las personas.

Marcelo Cabrera, ministro de transporte del Ecuador, explica los objetivos que se llevarán a cabo junto la estrategia nacional de movilidad segura. Entre estos se encuentra la gestión de movilidad, encargada de establecer vínculos con empresas para la correcta obtención de datos en el país sobre este tema; Mejorar la infraestructura vial para obtener vías más seguras; Promover vehículos con tecnología avanzada; Concientizar a los conductores a ser más responsables, obteniendo como resultado usuarios más seguros y confiados al volante. Y como último punto, mejorar la capacidad de respuesta ante accidentes de tránsito.

Figura 2

Principales Causas de Muertes por Accidentes de Tránsito

Datos de enero a abril de 2022



Fuente: ANT, 2022.

2.4 Asistencia al Conductor

Los sistemas de asistencia al conductor tienen como objetivo reducir cualquier impacto de colisión entre coches o evitar un accidente de tránsito. Estos sistemas pueden ser de varios tipos y son de gran ayuda para el conductor. (Soria, 2012).

Algunos de los tipos comunes de asistencia vehicular al conductor incluyen:

2.4.1 Asistencia de Mantenimiento de Carril.

Esta función utiliza cámaras y sensores para detectar las marcas del carril y ayuda al conductor a mantenerse dentro de los límites del carril. Si el vehículo se desvía involuntariamente del carril, el sistema puede emitir una advertencia sonora o aplicar ligeramente los frenos para corregir la trayectoria.

2.4.2 Control de Crucero Adaptativo.

También conocido como control de velocidad de crucero inteligente, este sistema utiliza sensores para detectar la distancia y la velocidad de los vehículos que se encuentran delante. Ajusta automáticamente la velocidad del vehículo para mantener una distancia segura y constante con el tráfico delantero.

2.4.3 Frenado de Emergencia Automático.

Este sistema detecta obstáculos o vehículos detenidos en la trayectoria del vehículo y, si el conductor no responde a tiempo, puede activar automáticamente los frenos para evitar una colisión o reducir su gravedad.

2.4.4 Alerta de Colisión.

Esta función utiliza sensores para monitorear el entorno del vehículo y detectar la proximidad de otros vehículos u objetos. Si se detecta un riesgo de colisión, el sistema emite una advertencia visual o sonora para alertar al conductor. Una alerta de colisión es una notificación o aviso emitido por un sistema de seguridad en un vehículo (ya sea autónomo o conducido por una persona) cuando se detecta un alto riesgo de colisión inminente con otro vehículo, peatón u obstáculo en la carretera. El objetivo principal de estas alertas es proporcionar a los conductores o al sistema de conducción autónoma una advertencia temprana para que puedan tomar medidas evasivas o frenar a tiempo y evitar un accidente

2.4.5 Asistencia de Estacionamiento.

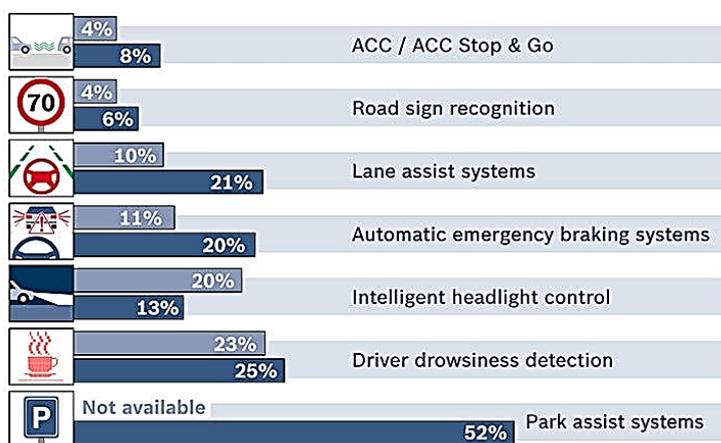
Este sistema utiliza sensores para detectar obstáculos alrededor del vehículo durante las

maniobras de estacionamiento. Puede proporcionar indicaciones visuales o sonoras para guiar al conductor y, en algunos casos, incluso puede asumir el control del volante para realizar estacionamientos automáticos.

Estos son solo algunos ejemplos de la amplia gama de asistencias vehiculares al conductor disponibles en los vehículos modernos. La tecnología continúa avanzando y se están desarrollando nuevas funciones para mejorar la seguridad y la experiencia de conducción.

Figura 3

Sistemas de Asistencia al Conductor en Carros Nuevos



Fuente: Bosch, 2013.

2.5 Automóviles con Asistencia al Conductor

El grupo de defensa del consumidor, Consumer Reports, realizó varias pruebas con más de diecisiete sistemas de asistencia al conductor y se enfocó en dos características específicas.

El control crucero mantiene una velocidad constante, pero consta de la opción de acelerar o frenar de acuerdo con la preferencia del usuario; El control de carril, posee cámaras, sensores que se enfocan en las líneas de la carretera y el tráfico vehicular para enviar presión al volante y mantener centrado el vehículo en su carril, una vez que se activa la señal.

Este grupo al terminar su estudio concluyó con que, el automóvil con mejor guía al conductor es el Super Cruise de Cadillac debido a que en comparación con otros vehículos, éste posee una cámara infrarroja que asegura la vista del conductor hacia la carretera.

Esta característica novedosa, es considerada nivel 3, ya que, si el usuario descuida la vista o se queda dormido, el monitor del vehículo enciende una advertencia visual y audible; En caso de que el conductor no reaccione, el sistema opta por reducir la velocidad del vehículo, activa peligros, se detiene y pide ayuda.

Desde los años 90, se han desarrollado diversos sistemas de detección de carriles que se han diseñado para adaptarse a diferentes condiciones de la carretera, ya sea en autopistas, calles urbanas o rurales. Sin embargo, las investigaciones recientes han revelado que estos sistemas mayormente identifican únicamente una línea de carril específica en tiempo real y no tienen la capacidad de detectar carriles adicionales, lo que limita su utilidad en situaciones como cierres de carriles, obras en la carretera y accidentes que puedan bloquear el carril de circulación. (C. D'Cruz and Ju Jia Zou)

Desde el año 1925, se ha buscado innovar en el ámbito automotriz, por lo que introdujeron por primera vez un prototipo de vehículo autónomo. (Stayton, 2011). La tecnología vehicular con el paso de los años ha avanzado de manera exponencial, por lo que cada vez se busca implementar nuevos sistemas, que sirvan de soporte al conductor.

En la actualidad, existen empresas interesadas en la mejoría y avance de los coches. Google, por ejemplo, se enfoca en sistemas de mapeo, gps a precisión; Mercedes, pone su atención en cámaras y radares; Tesla se enfoca en actualizaciones de software que brindan nuevas funciones al usuario; Volvo brinda asistencia al conductor y sistemas actualizados de control; Kia y Hyundai por su lado están implementando la fabricación de coches eléctricos con sistemas de asistencia vehicular lo que conlleva a una gran disminución al uso de fuentes fósiles como fuente de combustible para los motores de combustión interna.

Cuando se escucha acerca del vehículo eléctrico la sociedad en general tiende a pensar que se trata de un invento tecnológico reciente, no obstante, su investigación y desarrollo, aunque lento, datan de hace más de un siglo; de hecho, este se desarrolló más pronto que el

automóvil de combustión interna que predomina actualmente. (Alex Fernando Llerena Mena, 2020).

Los vehículos con asistencia al conductor son aquellos equipados con tecnologías avanzadas que ayudan al usuario en la conducción y la toma de decisiones. Estas características pueden incluir sistemas de frenado automático, control de crucero adaptativo, alerta de cambio de carril y estacionamiento automático. La asistencia vehicular tiene como objetivo mejorar la seguridad y la comodidad en la carretera al reducir el riesgo de colisiones y facilitar tareas de manejo. Sin embargo, es importante recordar que, a pesar de estas funciones, los conductores deben mantener su atención en la carretera en todo momento.

Numerosas marcas de vehículos ofrecen sistemas de asistencia al conductor en sus modelos más nuevos y avanzados. Lo que con lleva a analizar que cada día los avances en la industria automotriz son cada vez más grandes además estos sistemas pueden variar en términos de disponibilidad y nivel de sofisticación según el modelo y la línea de vehículos. Algunas de las marcas automotrices conocidas por incluir sistemas de asistencia al conductor en sus vehículos son:

2.5.1 *Tesla.*

Es conocida por su sistema de asistencia al conductor denominado Autopilot. Ofrece características como control de crucero adaptativo, cambio de carril y estacionamiento automáticos.

2.5.2 *BMW.*

Cuenta con su sistema de asistencia al conductor llamado Driving Assistant. Incluye funciones como control de crucero adaptativo, asistente de mantenimiento de carril y advertencia de colisión frontal.

2.5.3 *Mercedes-Benz.*

Ofrece varias tecnologías de asistencia al conductor bajo el nombre de sistema de

asistencia a la conducción inteligente (Intelligent Drive). Incluye funciones como control de crucero adaptativo, asistente de mantenimiento de carril, frenado de emergencia automático y asistencia de estacionamiento.

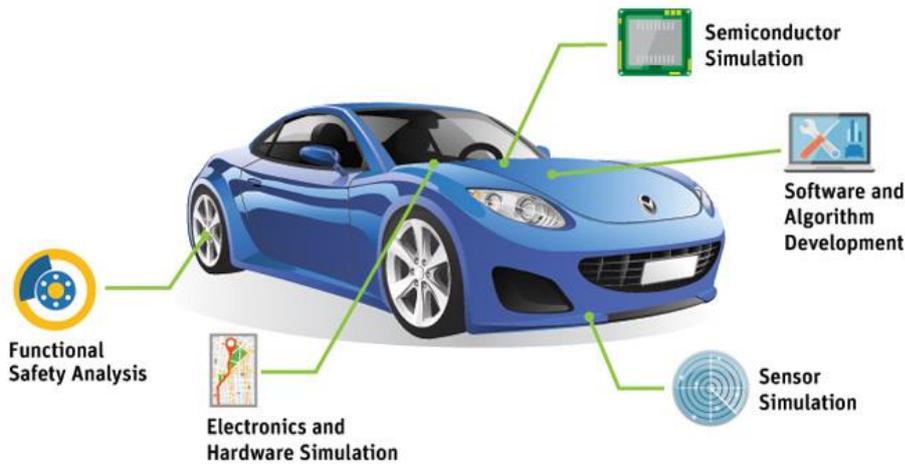
2.5.4 Audi.

Cuenta con su sistema de asistencia al conductor llamado Audi Pre Sense, que incluye características como control de crucero adaptativo, asistente de mantenimiento de carril y frenado de emergencia automático.

2.5.5 Volvo.

Es conocida por su enfoque en la seguridad, y ofrece tecnologías de asistencia al conductor bajo el nombre de Volvo IntelliSafe. Incluye características como control de crucero adaptativo, asistente de mantenimiento de carril, frenado de emergencia automático y asistente de estacionamiento.

Estas marcas de automóviles mencionadas anteriormente son solo una pequeña muestra de la amplia gama de fabricantes que han implementado sistemas de asistencia al conductor en sus vehículos. Para obtener una visión completa y actualizada de las opciones disponibles en términos de sistemas de ayuda al conductor en los diversos modelos de automóviles, es crucial realizar una investigación exhaustiva y consultar tanto a concesionarios como a las páginas web oficiales de los fabricantes. Esto te permitirá acceder a información detallada y precisa sobre las últimas tecnologías y características que están disponibles en el mercado automovilístico actual.

Figura 4*Vehículos Autónomos*

Fuente: ESSS, 2018.

2.6 Lenguajes de Programación

Los lenguajes de programación son herramientas fundamentales en la interacción entre humanos y computadoras. Cada lenguaje posee su propia estructura y reglas, permitiendo a los desarrolladores comunicar instrucciones y crear software de manera efectiva. Desde los lenguajes de alto nivel como Python, hasta los lenguajes de bajo nivel como C++, cada uno tiene sus ventajas y desafíos.

La Raspberry Pi es una computadora de placa única (SBC) muy popular y versátil que se utiliza en una amplia gama de proyectos. Puedes utilizar varios lenguajes de programación para desarrollar aplicaciones en la Raspberry Pi. Algunos de los lenguajes de programación comunes utilizados en las Raspberry Pi son los siguientes:

2.6.1 Python.

Es uno de los lenguajes más populares para programar en la Raspberry Pi. Es fácil de aprender y cuenta con una amplia variedad de bibliotecas y módulos que son compatibles con la Raspberry Pi.

2.6.2 C.

El lenguaje de programación C es ampliamente utilizado para programar sistemas embebidos, incluyendo la Raspberry Pi. Proporciona un control de bajo nivel y un acceso directo al hardware de la Raspberry Pi.

2.6.3 C++.

Es una extensión del lenguaje C que agrega características de programación orientada a objetos. Al igual que C, es utilizado para programar en la Raspberry Pi y ofrece un control de bajo nivel.

2.6.4 Java.

Es otro lenguaje de programación ampliamente utilizado que es compatible con la Raspberry Pi. Puedes desarrollar aplicaciones Java utilizando la plataforma Java Standard Edition (Java SE) en la Raspberry Pi.

2.6.5 JavaScript.

Es un lenguaje de programación utilizado principalmente para desarrollo web, pero también se puede utilizar en la Raspberry Pi. Con la ayuda de bibliotecas como Node.js, puedes ejecutar aplicaciones JavaScript en la Raspberry Pi.

2.6.6 Scratch.

Es un lenguaje de programación visual y educativo utilizado para enseñar a los niños los conceptos básicos de la programación. También se puede utilizar en la Raspberry Pi para desarrollar proyectos interactivos y educativos.

Estos son solo algunos de los lenguajes de programación utilizados en la Raspberry Pi. Sin embargo, la Raspberry Pi es compatible con muchos más lenguajes de programación.

Capítulo III.

Implementación del Prototipo de Asistencia al Conductor

La implementación del prototipo es de utilidad para brindar soporte durante la conducción en tiempo real, las partes del prototipo son una Raspberry modelo Pi4, la que llevará acabado la codificación y recepción de datos durante la conducción, adicional, está compuesto por una pantalla donde transmitirá y mostrarán los datos, lleva también una cámara web para la recepción de imagen y unos parlantes para reproducir sonido estéreo.

El conjunto de estas piezas lograra brindar soporte durante la conducción evitando que el conductor se descarrile involuntariamente y así evitar accidentes de tránsito.

3.1 Metodología para la Implementación

El desarrollo exitoso del prototipo requerirá la adopción de una metodología estructurada que abarque desde la selección de componentes hasta la implementación práctica del sistema. A continuación, se presenta una metodología detallada que incorpora los aspectos técnicos y las consideraciones específicas discutidas:

Comprender y documentar los requisitos clave del asistente de cruce de línea, considerando los objetivos del sistema, la detección de carriles y la alerta de colisión. Establecer los parámetros técnicos esenciales para la selección de componentes, como cámaras, microcontroladores y periféricos.

Investigar las opciones disponibles para las cámaras, microcontroladores, altavoces y otros componentes esenciales del sistema. Basándose en los criterios técnicos previamente identificados, seleccionar los componentes más adecuados para el prototipo.

Realizar el diseño de circuitos y conexiones necesarios para interconectar los componentes seleccionados, considerando la alimentación, comunicación y transferencia de datos entre los dispositivos.

Desarrollar el software necesario para la detección de carriles y alerta de colisión. Utilizar el microcontrolador seleccionado para procesar la información de la cámara, implementar algoritmos de detección y seguimiento, y generar alertas en caso de colisión inminente.

Integrar físicamente los componentes seleccionados. Asegurarse de que la cámara, el microcontrolador, los altavoces y otros periféricos estén conectados correctamente y funcionen en conjunto.

Realizar pruebas exhaustivas para verificar el funcionamiento del prototipo en diferentes escenarios de detección de carriles y alerta de colisión. Realizar ajustes en el software y el hardware según sea necesario para mejorar el rendimiento y la precisión.

3.2 Componentes del Prototipo

La elección y la integración adecuadas de los componentes del prototipo de asistencia de cruce de línea son fundamentales para garantizar el rendimiento y la funcionalidad óptimos del sistema. Cada componente, desde la cámara de detección hasta el microcontrolador y los periféricos, cumple un papel crucial en la ejecución coherente y precisa de las tareas asignadas al sistema. La calidad y compatibilidad de estos componentes influyen directamente en la capacidad del prototipo para detectar carriles, realizar un seguimiento de vehículos y emitir alertas de colisión de manera efectiva. Asimismo, una elección informada de los componentes contribuye a la robustez y la confiabilidad del sistema.

3.2.1 Cámara

La cámara web (Figura 5) es la encargada de realizar el monitoreo en tiempo real durante la conducción. Para determinar la cámara más idónea para la implementación del prototipo, se procede a evaluar las opciones presentadas en la tabla 1, centrándonos en un análisis exhaustivo de los aspectos técnicos siguientes:

- Resolución.

- Velocidad de captura, FPS.
- Sensibilidad a la luz.
- Diseño: compacto, tipo conexión.

Tabla 1*Comparativa de Cámaras Web*

	Genius	RaspiCam	WebCam Q6
Resolución	Malo - 720px	Bueno – 1080px	Bueno – 1080px
Costo	Alto	Alto	Bajo
Diseño	Malo	Regular	Bueno
Sensibilidad	Bueno	Bueno	Bueno

Figura 5*Cámara Web*

Al examinar en detalle los distintos parámetros y características técnicas de los componentes de cada cámara, se ha llegado a la conclusión de que la elección más adecuada es la WebCam Q6. La razón detrás de esta elección radica en su óptima relación entre la resolución de imagen que ofrece y su precio.

La comparación exhaustiva de las cámaras disponibles involucró una evaluación minuciosa de aspectos cruciales como la calidad de imagen, la velocidad de captura, la sensibilidad a la luz, el rango dinámico y otros atributos técnicos que influyen en el rendimiento

general de cada cámara. En este análisis detallado, la WebCam Q6 se destacó al proporcionar una resolución de imagen muy satisfactoria en relación con su costo.

La elección de una cámara para cualquier aplicación de visión es una decisión estratégica, ya que la calidad y claridad de las imágenes capturadas tienen un impacto directo en la precisión y confiabilidad del sistema de análisis. La WebCam Q6 ha demostrado ser una solución que equilibra eficazmente estas consideraciones clave, al ofrecer una resolución que permite capturar detalles importantes en las imágenes, al tiempo que se ajusta a un presupuesto razonable.

Este enfoque en la relación resolución-precio no solo garantiza que se cumplan las necesidades técnicas y operativas del proyecto, sino que también optimiza la inversión al proporcionar resultados de alta calidad sin comprometer el aspecto económico. La elección de la WebCam Q6 subraya la importancia de tomar decisiones informadas basadas en un análisis cuidadoso y un entendimiento profundo de los requisitos específicos de la aplicación.

En resumen, después de un análisis detenido y ponderado de las características técnicas de las diferentes cámaras, la elección final de la WebCam Q6 se respalda en su capacidad para brindar una excelente resolución de imagen a un costo favorable, lo que la convierte en la opción óptima para cumplir con los objetivos y requisitos del proyecto en cuestión.

3.2.2 Altavoces

Los altavoces (Figura 6) son los encargados de emitir el sonido audible como alerta en el prototipo. Con el propósito de elegir los altavoces más apropiados para la realización del prototipo, se lleva a cabo una evaluación de los tres altavoces detallados en la tabla 2, poniendo especial énfasis en el examen minucioso de los elementos técnicos que se describen a continuación:

- Reproducir sonido.
- Diseño compacto, entrada Jack 3.5mm y puerto USB con alimentación de 5v

- Conexión a la Raspberry mediante puerto USB
- Entrada de audio por el puerto Jack

Tabla 2*Comparativa de Altavoces*

	Havit	Genius	RadioShack
Diseño	Bueno	Regular	Malo
Costo	Bueno	Malo	Malo
Potencia	Bueno	Regular	Bueno
Portabilidad	Bueno	Bueno	Bueno

Figura 6*Altavoces*

Luego de realizar un análisis minucioso de los detalles recopilados en la tabla 2, es posible llegar a una decisión informada: los altavoces Havit emergen como la opción óptima. Esta elección se basa en la observación de cómo estos altavoces cumplen con todos los parámetros definidos previamente para su evaluación.

La capacidad de reproducir sonido de manera efectiva es esencial en cualquier conjunto de altavoces, y los altavoces Havit demuestran un rendimiento confiable en esta área. Además, su diseño compacto, junto con la disponibilidad de una entrada Jack 3.5mm y un puerto USB

con alimentación de 5v, confirma su adaptabilidad a diversas situaciones y fuentes de alimentación.

Es fundamental que los altavoces estén conectados a la Raspberry Pi de manera fluida, y los altavoces Havit logran esto de manera efectiva a través de su conexión USB. La posibilidad de utilizar la entrada de audio por el puerto Jack agrega versatilidad y permite su integración con diferentes dispositivos y sistemas de audio.

La elección de los altavoces adecuados es un aspecto crítico en el desarrollo de este proyecto, ya que afecta directamente la calidad, la experiencia auditiva y será el principal medio de alerta al conductor.

3.2.3 Pantalla

La pantalla de 7 pulgadas (Figura 7) muestra imágenes, contenido con claridad y nitidez. Su tamaño compacto la hace ideal para diversas aplicaciones. Para establecer la cámara más adecuada para la ejecución del prototipo, se lleva a cabo una evaluación de las alternativas mencionadas en la tabla 3, focalizando nuestra atención en un examen detallado de los siguientes elementos técnicos:

- Tamaño Monitor portátil.
- Resolución.
- Tipo de entradas/salidas de video
- Alimentación de 5v mediante micro USB

Tabla 3

Comparativa de Pantallas

	HMTECH	ROADOM	INGCOOL
Resolución	Bueno – 1080px	Bueno – 1080px	Bueno – 1080px
Costo	Bueno	Malo	Malo
Panel	Bueno - IPS	Bueno - OLED	Bueno - LCD
Construcción	Bueno	Regular	Bueno

Figura 7*Pantalla*

Tras un análisis exhaustivo de los resultados de la comparativa, se destaca un factor clave que orienta la elección: la resolución de la pantalla y tipo de panel, con el precio actuando como un factor secundario pero relevante. Con esta perspectiva en mente, se concluye que la opción óptima es la pantalla HMTECH. Esta elección se fundamenta en la resolución superior de su pantalla en relación con los otros competidores, y aunque también se valora su costo más bajo, es su rendimiento visual lo que lo distingue.

La resolución de la pantalla desempeña un papel central en la calidad de imagen que una cámara puede ofrecer. En este análisis, la pantalla HMTECH sobresale al proporcionar una resolución que es significativamente mejor que la de sus contrapartes. Esta característica es de gran importancia en aplicaciones que requieren una alta definición y precisión visual.

Esta elección resalta la importancia de considerar los atributos técnicos esenciales en primer lugar, y luego evaluar cómo esos atributos se equilibran con el costo. La HMTECH ejemplifica cómo es posible obtener una resolución de pantalla superior sin incurrir en un gasto excesivo.

3.2.4 *Microcontrolador*

El microcontrolador (Figura 8) es el encargado de procesar toda la información recibida de los periféricos de entrada conectados entregar información por los puertos o periféricos de salida. Se analizan tres microcontroladores los cuales están mostrados en la tabla 4 y se toman en cuenta las siguientes características técnicas:

- Procesador.
- Modelo.
- Memoria RAM.
- Cantidad de puertos de conexión para los periféricos
- Fuente de alimentación.
- La interfaz de comunicación
- Sistema operativo.

Tabla 4

Comparativa de Raspberry

	Arduino	Raspberry Pi3	Raspberry Pi4
Procesador	Malo – Atmega 328P	Regular – ARM Cortex A53	Bueno – Broadcom Quad-Core Cortex A72
RAM	Malo – 1Kb Ram	Regular – 1 Gb Ram	Bueno – 8Gb Ram
Puertos	Bueno	Malo	Bueno
Conectividad	Bueno	Bueno	Bueno

Al llevar a cabo un análisis exhaustivo de los tres microcontroladores disponibles, a saber, Arduino, Raspberry Pi 3 y Raspberry Pi 4, y considerando atentamente los criterios técnicos mencionados, se llega a la conclusión de que la Raspberry Pi 4 (figura 8) es la elección más apropiada para el prototipo en cuestión.

La decisión se basa en un enfoque meticuloso de los diferentes parámetros que determinan la idoneidad de cada microcontrolador:

El procesador, la Raspberry Pi 4 sobresale con un procesador ARM Cortex-A72 de cuatro núcleos a 1.5 GHz, ofreciendo un rendimiento más potente en comparación con el procesador de Arduino y Raspberry Pi 3. Este nivel de potencia es crucial para manejar tareas más complejas y demandantes.

Figura 8

Raspberry Pi 4



La memoria RAM de la Raspberry Pi 4 ofrece opciones de memoria RAM de 2 GB, 4 GB y 8 GB, proporcionando una flexibilidad y capacidad de ejecución de aplicaciones más exigentes que supera con creces las capacidades de Arduino y Raspberry Pi 3 en este aspecto.

La cantidad de puertos de conexión de la Raspberry Pi 4 se destaca al ofrecer múltiples puertos USB y puertos GPIO, incluyendo puertos USB 3.0 de mayor velocidad. Esto permite una mayor conectividad con periféricos y dispositivos externos en comparación con las opciones limitadas de Arduino y Raspberry Pi 3.

La interfaz de comunicación de la Raspberry Pi 4 mantiene una ventaja significativa al proporcionar una gama de interfaces de comunicación que incluyen UART, SPI, I2C y Ethernet. Esta diversidad es esencial para una comunicación eficaz con diversos dispositivos y sistemas, superando las opciones más limitadas de los otros microcontroladores.

El sistema operativo de la Raspberry Pi 4 ofrece la capacidad de ejecutar sistemas

operativos completos como Linux, lo que brinda una flexibilidad y versatilidad excepcionales en términos de aplicaciones y funcionalidades. Esto se destaca frente a las capacidades más restringidas de Arduino y Raspberry Pi 3.

La decisión de optar por la Raspberry Pi 4 resalta cómo el análisis cuidadoso de los criterios técnicos resulta en la identificación de un microcontrolador que no solo satisface las necesidades específicas del proyecto, sino que también proporciona una plataforma versátil y potente para futuras aplicaciones y expansión.

3.2.5 Fuente de Energía

La fuente de alimentación de 5V (Figura 9) diseñada para conectarse a la toma de cigarrera de un vehículo representa una solución ingeniosa para suministrar energía en movimiento. Esta fuente de alimentación está especialmente concebida para proporcionar una corriente constante de 5 voltios, permitiendo cargar y alimentar dispositivos electrónicos de baja potencia mientras se viaja.

- Voltaje que entrega
- Tipo de conexión
- Diseño

En un análisis técnico riguroso, se revela que esta fuente de alimentación ofrece una corriente constante de 5 voltios. Esta especificación técnica es vital, ya que asegura que el suministro eléctrico sea uniforme y confiable en todo momento. Esto es de particular importancia en situaciones móviles, como en el interior de un vehículo en movimiento, donde las fluctuaciones en la alimentación eléctrica podrían tener impactos adversos en dispositivos electrónicos conectados.

La ingeniosidad de esta solución se extiende a su adaptabilidad a dispositivos de baja potencia. Además de proporcionar una fuente de carga, la fuente de alimentación de 5V también puede alimentar electrónicos de baja potencia mientras se viaja. Esto resulta útil para

mantener el microcontrolador y otros dispositivos electrónicos funcionando sin interrupciones mientras el vehículo está en movimiento.

Figura 9

Fuente de Alimentación



3.2.6 *Mouse y Teclado*

El conjunto Mouse y Teclado, representado en la Figura 10, desempeña un papel esencial al facilitar la interacción del usuario con la interfaz de la Raspberry, permitiendo la ejecución de comandos y búsquedas de manera eficiente. Una exploración más profunda de este conjunto, basada en las directrices previas, revela las siguientes características clave:

- Tipo de conexión
- Diseño

La modalidad de conexión de este Mouse y Teclado es un factor técnico primordial. La elección del tipo de conexión, que puede ser cableada o inalámbrica, tiene un impacto directo en la comodidad y la flexibilidad del usuario al interactuar con la Raspberry. Cada opción ofrece sus propias ventajas y consideraciones en términos de rendimiento y alcance.

El diseño de este también merece atención en el análisis. Factores como la ergonomía, el tamaño y la disposición de las teclas juegan un papel crítico en la experiencia del usuario. Un diseño ergonómico asegura comodidad durante el uso prolongado, mientras que la disposición de las teclas, junto con la presencia de botones adicionales en el Mouse, puede

influir en la eficiencia de la navegación y la ejecución de comandos.

Al sopesar estos criterios técnicos esenciales, se subraya la importancia de seleccionar un Mouse y Teclado que no solo satisfaga las necesidades funcionales del usuario, sino que también garantice la comodidad y la eficacia en la operación diaria de la Raspberry. El análisis informado de la conexión y el diseño contribuye a tomar decisiones adecuadas que optimicen la interacción del usuario con la interfaz y potencien la experiencia global.

Figura 10

Teclado con Touchpad



3.3 Elementos de Gasto

Se realiza una descripción detallada de los costos de diseño mostrado en la tabla 5, tomando en cuenta materiales, herramientas y procesos que se han utilizado para la implementación del prototipo, con el fin de determinar el costo de su fabricación.

Tabla 5

Resumen del Presupuesto

Detalle	Costo
Raspberry Pi 4 Kit	\$160.00
Monitor 7" HDMI	\$50.00
Cámara frontal	\$24.00

Cableado y otros	\$100.00
Altavoces	\$15.00
Mouse y teclado	\$15.00
Total	\$364.00

3.4 Desarrollo del Algoritmo

Para la elaboración del algoritmo se utilizó Microsoft Visual Studio Code, donde es ampliamente reconocido como un excelente entorno de desarrollo de software en la comunidad tecnológica. Su popularidad se debe a su versatilidad, rendimiento y funcionalidades avanzadas, lo que lo convierte en una opción destacada para desarrolladores de diferentes lenguajes de programación.

Este entorno ofrece una interfaz de usuario intuitiva y personalizable, lo que permite a los usuarios adaptarlo a sus preferencias y necesidades específicas. Además, cuenta con una extensa colección de extensiones desarrolladas por la comunidad, que proporcionan una amplia gama de características adicionales, desde soporte para lenguajes específicos hasta integración con herramientas de desarrollo y control de versiones.

Además, Visual Studio Code, es una herramienta multiplataforma, compatible con Windows, macOS y Linux, lo que permite a los desarrolladores trabajar en diversos sistemas operativos sin problemas de compatibilidad.

En resumen, Visual Studio Code, destaca como un entorno de desarrollo de software altamente valorado gracias a su adaptabilidad, extensibilidad, eficiencia y soporte para múltiples plataformas, lo que lo convierte en una opción preferida para programadores en busca de una experiencia de desarrollo moderna y efectiva.

3.4.1 Librerías Utilizadas en la Elaboración del Código

Para que una aplicación de visión funcione requiere de un algoritmo de programación que cumpla con las tareas que debe ejecutar el sistema. Gracias al desarrollo de librerías especializadas al procesamiento digital de imágenes y visión por computadora, desarrollar una

aplicación no requiere empezarla desde cero ya que por lo general contienen funciones que ya están establecidas para facilitar el trabajo. (Miranda León, 2019)

3.4.2 *Open Source Computer Vision Library*

Open CV es una biblioteca de código abierto que se utiliza para desarrollar aplicaciones de visión por computadora. Es ampliamente utilizada para el procesamiento de imágenes y videos, reconocimiento de objetos, detección de rostros, seguimiento de objetos en movimiento, calibración de cámaras, entre otras tareas relacionadas con la visión por computadora.

Algunas de las funcionalidades clave que ofrece OpenCV incluyen:

- Lectura y escritura de imágenes y videos en diferentes formatos.
- Procesamiento de imágenes, como filtrado, convolución, transformaciones geométricas, etc.
- Detección y reconocimiento de características como esquinas, bordes y contornos.
- Detección de objetos y seguimiento de objetos en movimiento.
- Reconocimiento facial y detección de rostros.
- Calibración de cámaras y análisis de perspectivas.

3.4.3 *NumPy*

Es una biblioteca para el lenguaje de programación Python que se utiliza para realizar operaciones numéricas y matemáticas en matrices y arreglos multidimensionales. Es una de las bibliotecas más fundamentales para el procesamiento de datos en Python y es ampliamente utilizada en el campo de la ciencia de datos, aprendizaje automático y análisis numérico.

Algunas de las funcionalidades clave que ofrece NumPy incluyen:

- Creación y manipulación eficiente de matrices multidimensionales (arrays).
- Realización de operaciones matemáticas y estadísticas en matrices, como

sumas, restas, multiplicaciones, etc.

- Indexación y selección de elementos en matrices.
- Generación de secuencias numéricas y funciones matemáticas avanzadas.
- Integración con otras bibliotecas numéricas y de aprendizaje automático.

3.4.4 *Pygame*

Es una biblioteca de Python que se utiliza para desarrollar videojuegos y aplicaciones multimedia interactivas. Proporciona herramientas para crear gráficos, reproducir sonidos, manejar eventos del teclado y el ratón, y mucho más. Aunque es popular en el desarrollo de juegos, también se puede utilizar para crear aplicaciones interactivas con interfaces gráficas.

Algunas de las funcionalidades clave que ofrece Pygame incluyen:

- Creación de ventanas y superficies gráficas para renderizar gráficos.
- Carga y manipulación de imágenes y sprites.
- Reproducción de sonidos y música.
- Captura y manejo de eventos del teclado y el ratón.
- Detección de colisiones entre objetos en el juego.
- Creación de animaciones y efectos visuales.

3.4.5 *Time*

Es una biblioteca estándar de Python que proporciona funciones para medir y manipular el tiempo. Se utiliza para realizar tareas relacionadas con el tiempo, como pausar la ejecución de un programa, medir el tiempo de ejecución de una sección de código u obtener la hora actual.

Algunas de las funcionalidades clave que ofrece la biblioteca `time` incluyen:

- Funciones para medir intervalos de tiempo, como `time.time()` y `time.sleep()`.
- Manipulación de fechas y horas con la función `time.strftime()` y `time.strptime()`.
- Obtener el tiempo actual con `time.time()` o en formato legible con

``time.ctime()``.

3.4.6 *Math*

Es otra biblioteca estándar de Python que proporciona funciones matemáticas. Es útil para realizar operaciones matemáticas más avanzadas que no están incluidas en la biblioteca básica de Python.

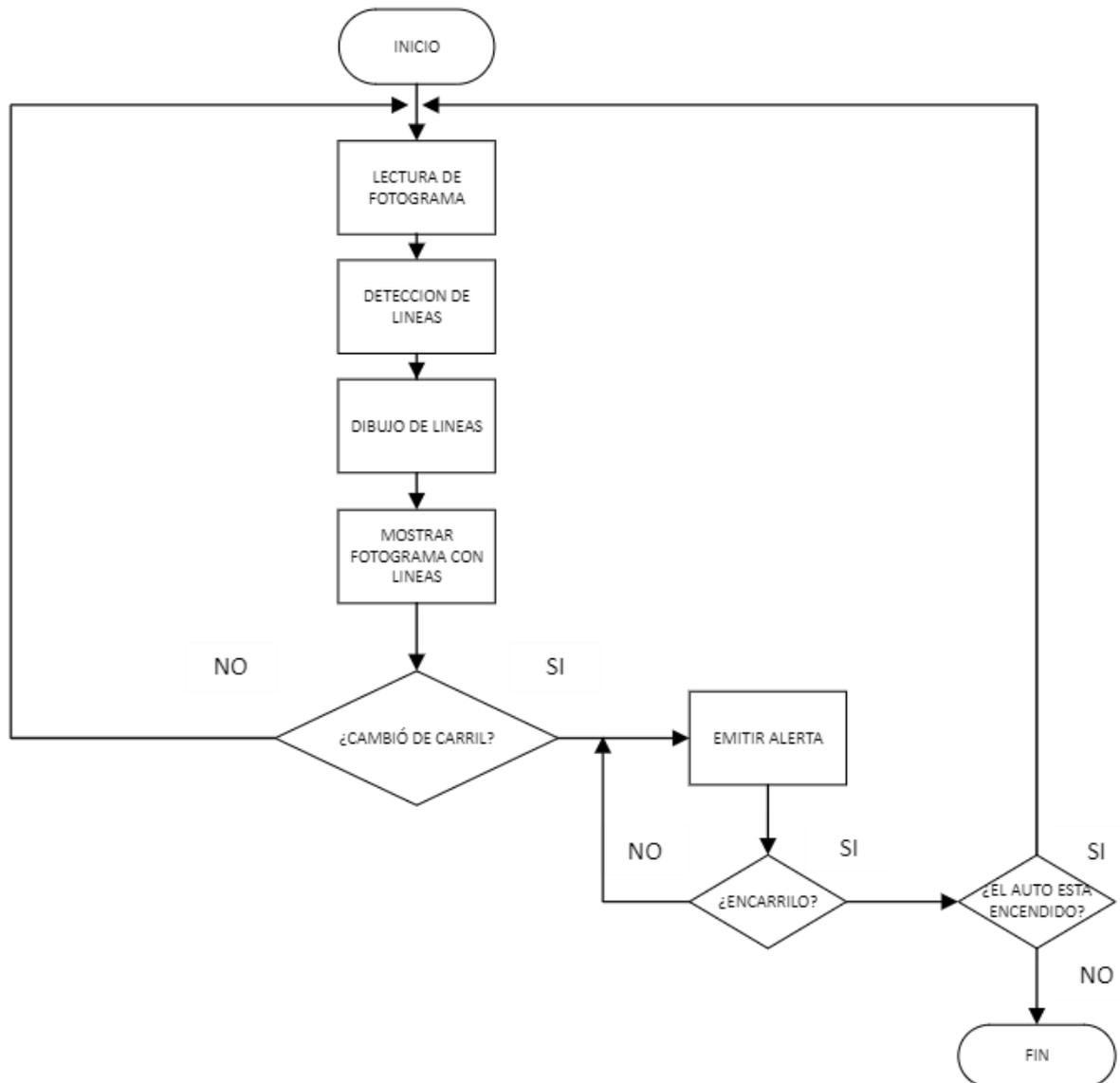
Algunas de las funcionalidades clave que ofrece la biblioteca ``math`` incluyen:

- Funciones trigonométricas, como seno, coseno y tangente.
- Funciones exponenciales y logarítmicas.
- Funciones para redondear y truncar números.
- Constantes matemáticas, como pi y e.

Las bibliotecas mencionadas, como OpenCV, NumPy, Pygame, time, y math, serán componentes clave en el desarrollo de esta aplicación. OpenCV se utilizará para llevar a cabo tareas de procesamiento de imágenes y visión por computadora, mientras que NumPy será esencial para realizar operaciones numéricas en matrices. Pygame se empleará para crear interfaces interactivas y posiblemente para la detección de colisiones en un entorno de juego. Finalmente, las bibliotecas time y math serán utilizadas para medir y manipular el tiempo y para llevar a cabo cálculos matemáticos más avanzados.

En resumen, el desarrollo se centrará en la creación de una aplicación basada en visión por computadora, utilizando diversas bibliotecas especializadas para llevar a cabo tareas de procesamiento de imágenes, detección de objetos y características, así como para la creación de una experiencia interactiva.

3.4.7 Diagrama de Flujo del Funcionamiento del Algoritmo Implementado



3.5 Esquema de Conexión de los Componentes Seleccionados

Para llevar a cabo la implementación del prototipo de asistencia de cruce de carril, es fundamental establecer un esquema de conexiones que permita la integración adecuada de los componentes seleccionados. A continuación, se detalla el esquema de conexiones de los elementos clave:

1. La WebCam Q6, se conecta a uno de los puertos USB de la Raspberry Pi 4. La alimentación de la cámara se obtiene directamente a través de la Raspberry Pi 4.

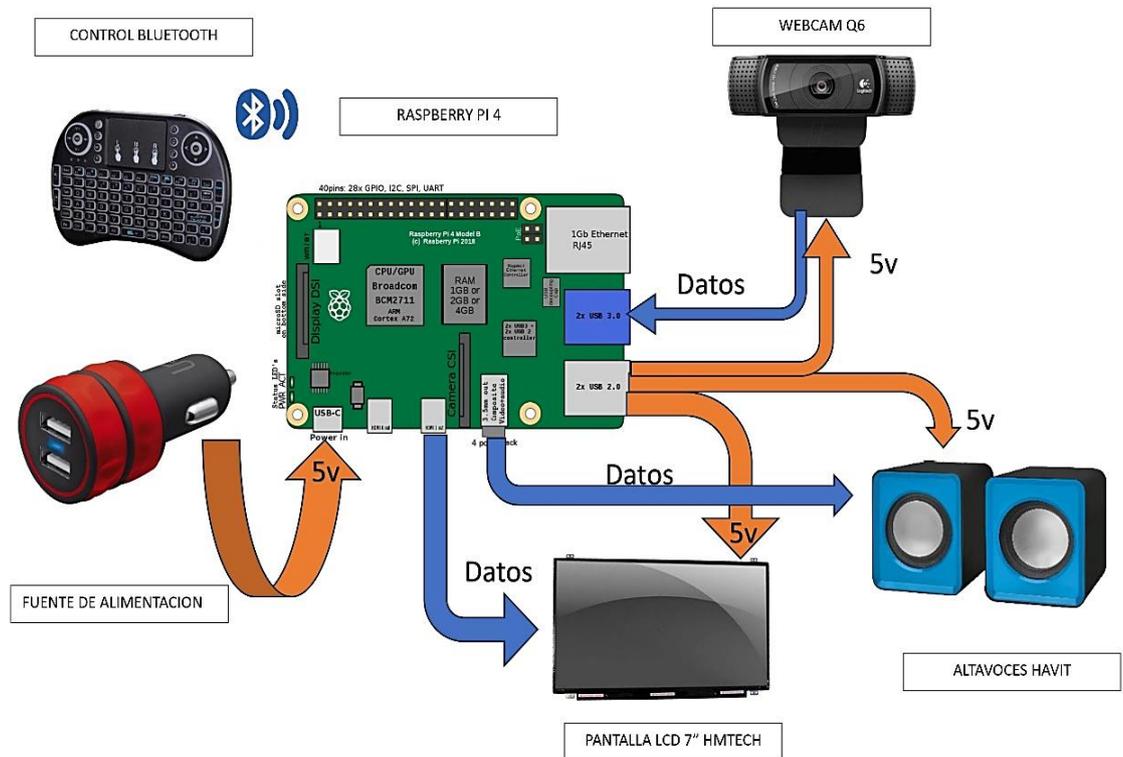
2. Altavoces Modelo Havit, se conectan a través del puerto de salida de audio de 3.5mm de la Raspberry Pi 4. La alimentación de los altavoces se obtiene del puerto USB de la Raspberry Pi 4.

3. La Raspberry Pi 4, se conecta a través de su fuente de alimentación al suministro de energía de 5V. La comunicación con la cámara se establece a través del puerto USB. Los periféricos, como el teclado y el mouse, se conectan a los puertos USB de la Raspberry Pi 4. La interfaz de comunicación y control se lleva a cabo mediante la configuración de su sistema operativo.

4. La pantalla HMTECH, se conectan a través del puerto Micro HDMI a la Raspberry Pi 4. La alimentación de la pantalla se obtiene del puerto USB de la Raspberry Pi 4.

5. La fuente de alimentación, se conecta a través de la cigarrera del vehículo cuyo suministro de carga es de 12v, donde la misma procederá a convertir dicho voltaje inicial a 5v donde mediante un cable USB tipo C se conecta directamente a la Raspberry Pi 4 suministrándole voltaje a toda la placa y posterior a todos sus periféricos.

Este esquema (figura 11) de conexiones asegura una integración efectiva de los componentes seleccionados, permitiendo la comunicación y el funcionamiento coordinado entre la cámara, la pantalla, los altavoces y la Raspberry Pi 4. El resultado es un sistema interconectado capaz de realizar la detección de cruce de carril y proporcionar alertas de manera eficiente y precisa.

Figura 11*Esquema de Conexión.*

3.6 Implementación del Prototipo

Se colocan los componentes en el vehículo de muestra, que en este caso será una camioneta Mazda modelo Bt- 50 año 2010 doble cabina en la cual se implementará el prototipo.

Los componentes para usar en este caso son: Raspberry Pi 4, Pantalla, Altavoces, Teclado-mouse, Cámara web y fuente de alimentación.

A continuación, se hará una descripción de los pasos para la implementación.

Para el primer paso se colocó la cámara web en el retrovisor frontal para que capte el trayecto, como se muestra en la figura 11 y figura 12.

Figura 12

Instalación Cámara Web

**Figura 13**

Instalación Cámara Web Vista Frontal.



Para el segundo paso se coloca la pantalla y Raspberry en el salpicadero del vehículo para una mejor visualización de los datos recibidos del prototipo como se observa en la figura

13 y figura 14.

Figura 14

Instalación Pantalla y Raspberry



Figura 15

Instalación Pantalla y Raspberry vista posterior.



Para el tercer paso se coloca los altavoces en la consola del vehículo donde se encuentra alimentado con 5v de la Raspberry mediante puerto USB como se observa en la figura 15

Figura 16

Instalación Altavoces.



Para el cuarto paso se coloca la fuente de 5v siendo está ubicada en la cigarrera y a su vez se conecta un cable de USB a tipo C directo a la Raspberry alimentándola con voltaje para que pueda realizar todas sus funciones como se visualiza en la figura 16 y figura 17

Figura 17

Fuente de Alimentación.



Figura 18

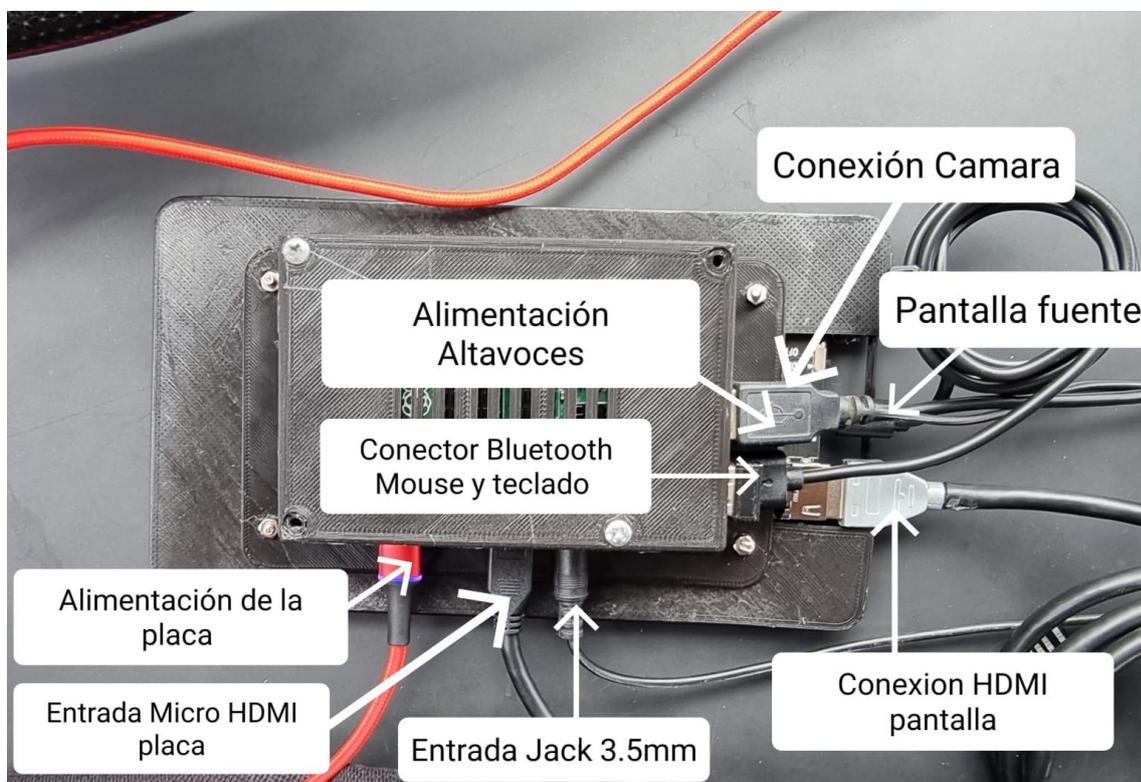
Fuente de Alimentación Instalada.



Para el quinto paso realizamos todas las conexiones de los componentes a la Raspberry como se detalla en la figura 18

Figura 19

Instalación Componentes Raspberry.



Capítulo IV.

Pruebas y Resultados de la Implementación

Las pruebas y los resultados de la implementación en el contexto del prototipo de asistencia de cruce de línea revisten una importancia crítica en el proceso de desarrollo y validación del sistema. Estas etapas permiten verificar la efectividad, la confiabilidad y la seguridad del prototipo en situaciones reales, asegurando que cumpla con los objetivos y requisitos establecidos.

Las pruebas brindan la oportunidad de evaluar cómo el prototipo responde ante una variedad de escenarios y condiciones. Esto permite identificar posibles desafíos, limitaciones y áreas de mejora que pueden no haber sido evidentes durante la fase de diseño y desarrollo. Además, las pruebas también permiten ajustar y afinar algoritmos, configuraciones y parámetros para optimizar el rendimiento y la precisión del sistema.

La obtención de resultados sólidos y confiables valida la eficacia del prototipo en la detección de carriles, el seguimiento de vehículos y la generación de alertas de colisión. Estos resultados son esenciales para respaldar la utilidad y la viabilidad del sistema en entornos del mundo real. Además, los resultados positivos proporcionan una base sólida para la presentación y la justificación del prototipo ante audiencias interesadas, como equipos de desarrollo, inversores y posibles usuarios.

En última instancia, las pruebas y los resultados de la implementación no solo son vitales para garantizar la calidad y la funcionalidad del prototipo, sino también para respaldar la toma de decisiones informadas en cuanto a la viabilidad de su implementación a mayor escala. Un prototipo que ha sido sometido a pruebas exhaustivas y ha demostrado resultados prometedores se posiciona como una solución confiable y efectiva, con el potencial de abordar desafíos de seguridad vial y mejorar la experiencia de conducción en cruces de carriles.

4.1 Pruebas del Prototipo.

La validación efectiva del proyecto implica la aplicación de métricas y pruebas que evalúen tanto el rendimiento técnico como la utilidad práctica del sistema. Se han tomado las siguientes métricas para validar el funcionamiento del prototipo.

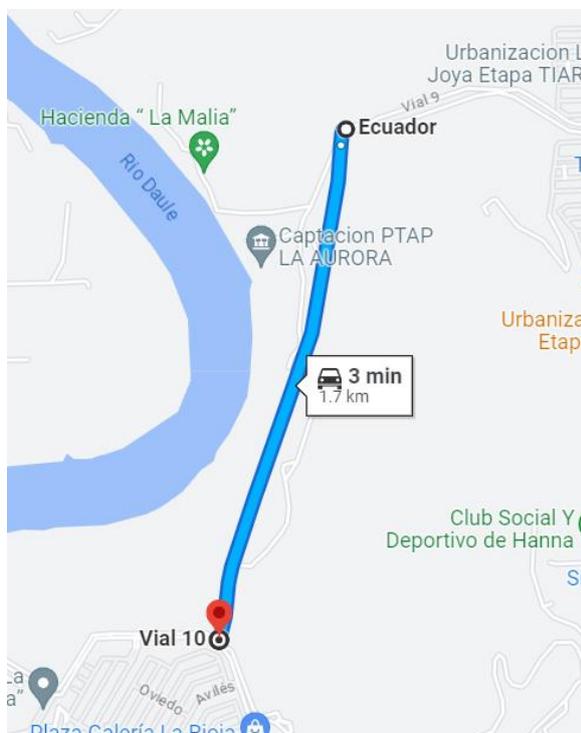
4.1.1 Pruebas de Campo

Realizar pruebas en condiciones reales para observar cómo el sistema se comporta en un entorno dinámico. Esto ayuda a confirmar que el sistema puede manejar las complejidades y desafíos del tráfico cotidiano.

Las pruebas se realizaron en el vial 10 de la urbanización La Joya que conecta con la urbanización La Rioja dando una distancia aproximada de 1,70km (Figura 19) lo suficiente para realizar la toma de datos del prototipo en el mundo real.

Figura 20

Ruta Vial 10 La joya – La Rioja



Los resultados sobresalientes obtenidos durante las pruebas de campo del prototipo de asistencia vehicular fue la respuesta positiva y efectiva de un conductor ante una situación de

cruce de carril detectada por el sistema.

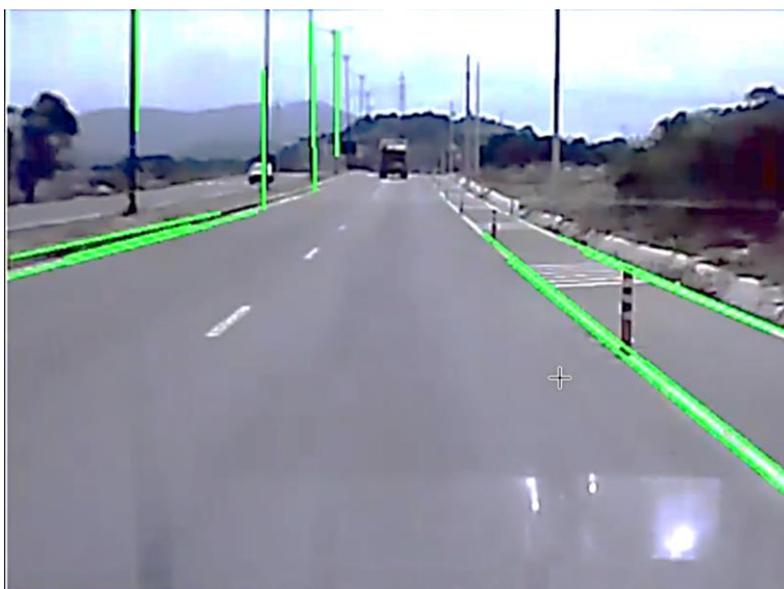
Durante una prueba en condiciones reales de tráfico en una carretera de doble carril, el sistema de asistencia de cruce de carril detectó una desviación inesperada del vehículo hacia el carril adyacente. En respuesta, el sistema activó una alerta audible para notificar al conductor sobre la desviación.

El conductor reaccionó de manera rápida y precisa a la alerta, corrigiendo inmediatamente la trayectoria del vehículo y volviendo al carril original. Esta reacción demostró la efectividad del sistema al proporcionar una notificación temprana y clara al conductor, permitiéndole tomar medidas inmediatas para evitar el cruce no deseado.

Este resultado subraya el valor del sistema de asistencia de cruce de carril en situaciones críticas de conducción y su capacidad para prevenir situaciones de riesgo. La respuesta exitosa del conductor durante la prueba valida la eficacia del sistema en el mundo real y respalda su utilidad en la mejora de la seguridad vial al evitar situaciones de cruce de carril y potenciales accidentes.

Figura 21

Trayecto de Ruta (Reconocimiento de Líneas Guías).

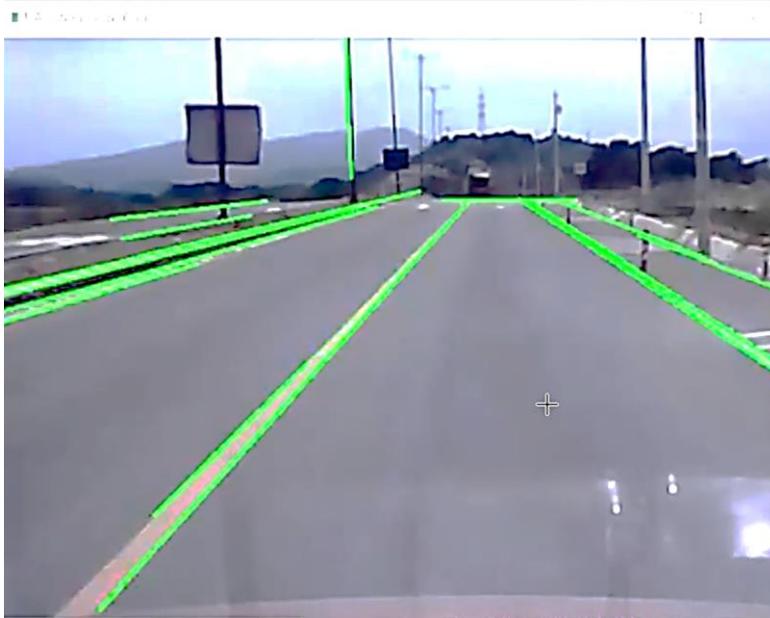


En la Figura 20 se realizó la activación del prototipo lo que dio un reconocimiento de

las líneas guías de la carretera.

Figura 22

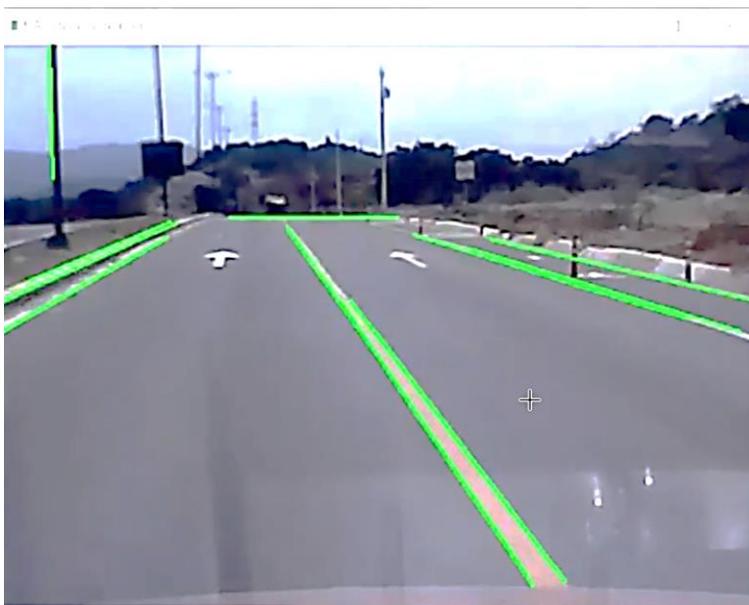
Trayecto de Ruta (Reconocimiento de Líneas Guías y Línea Continua)



En la Figura 21 se puede reconocer como el programa detecta líneas continuas de la carretera.

Figura 23

Trayecto de Ruta (Intento de Descarrilamiento)

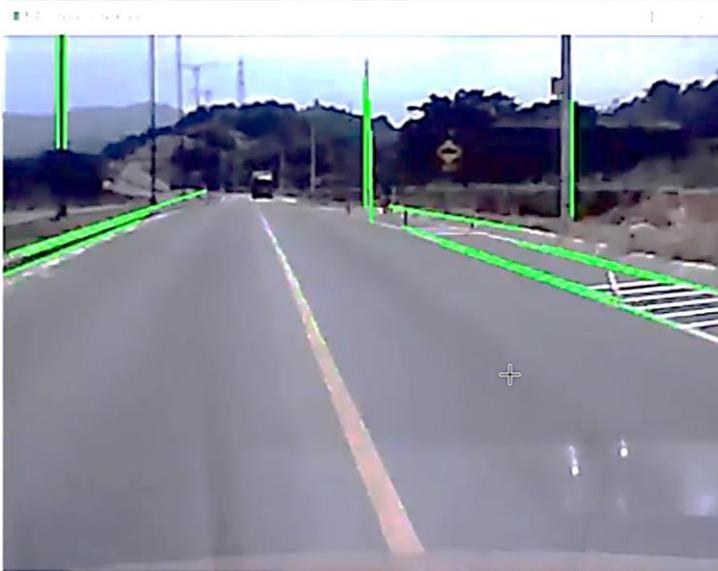


En la Figura 22 se procede con la primera prueba realizar un descarrilamiento

voluntario para comprobar el funcionamiento del prototipo.

Figura 24

Trayecto de Ruta (Prototipo Emite Alerta)



En la Figura 23 se visualiza como el vehículo ha descarrilado lo que conlleva el sonido de la alerta indicándole al conductor que regrese a su carril.

Figura 25

Trayecto de Ruta (Reincorporación a Carril, Deja de Emitir Pitido)

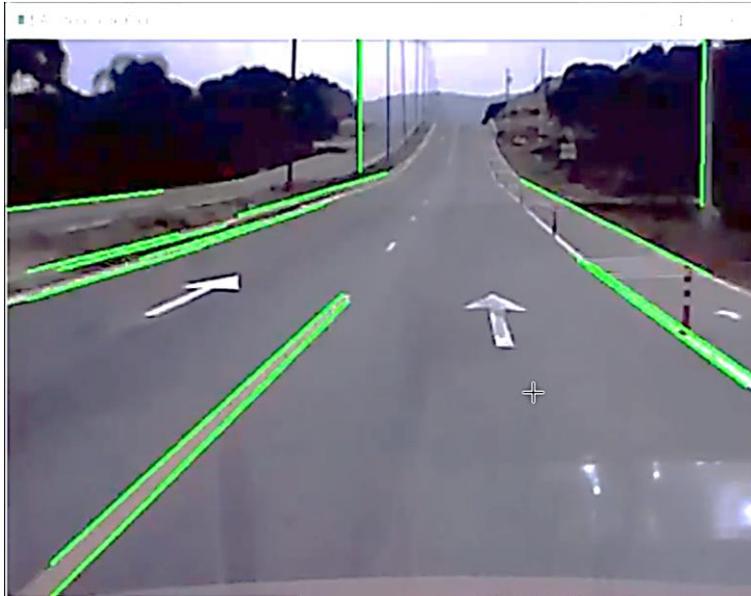


En la Figura 24 una vez el programa detecta que el vehículo está dentro del carril

procede a apagar la alerta audible.

Figura 26

Prueba de Ruta (Intento 2 de Descarrilamiento)



En la Figura 25 se realiza la segunda prueba de ruta donde el prototipo comienza analizando las líneas guías sin emitir alerta.

Figura 27

Prueba de Ruta (Momento Exacto Donde Emite la Alerta)



En la Figura 26 se visualiza como la trayectoria del vehiculo ha cambiado lo que

significa que el vehículo ha descarrilado lo que conlleva el encendido de la alerta audible.

Figura 28

Prueba de Ruta (Reincorporación de Carril, Deja de Emitir Alerta Audible)



En la Figura 27 una vez que el prototipo reconoce que el vehículo se encuentra dentro de los parámetros o las líneas guías procede a apagar la alerta audible.

4.1.2 Precisión de Detección y Alertas

Establecer métricas para medir la precisión de la detección de carriles y la generación de alertas de colisión. Utilizar escenarios simulados y reales para evaluar si el sistema detecta correctamente los carriles. En la tabla 6 se refleja los resultados obtenidos.

Tabla 6

Resultados de las Pruebas sobre Detección y Alertas

Escenario	Total de cruces	Cruces detectados	Cruces no detectados	Porcentaje de detección
Escenario 1	50	47	3	94%
Escenario 2	75	71	4	96%
Promedio	62.5	59	3.5	95%

La descripción de los tres escenarios utilizados para evaluar la precisión de detección y

alertas del prototipo se describen a continuación:

- Escenario 1: Tráfico Ligero en Carretera Recta

En este escenario, se simula un tráfico ligero en una carretera recta. Se generan 50 situaciones de posible colisión entre vehículos en carriles adyacentes. El sistema de asistencia de cruce detecta correctamente 47 cruces, lo que resulta en un alto porcentaje de detección del 94%. Tres cruces no son detectadas, lo que sugiere un potencial margen de mejora en condiciones específicas.

- Escenario 2: Cambio de Carril Brusco

Este escenario simula situaciones en las que un vehículo cambia de carril repentinamente. Se crean 75 instancias de cruce en condiciones de cambio de carril brusco. El sistema logra detectar 72 cambios, demostrando una precisión del 96%. Solo tres cruces no son detectados, lo que indica una respuesta efectiva en condiciones de cambio rápido de carril.

4.1.3 *Tiempo de Respuesta del Prototipo*

Medir el tiempo que transcurre desde que se detecta un cruce hasta que se emite la alerta. Establecer un umbral aceptable y comparar los resultados de las pruebas con esta métrica para asegurarse de que el sistema responda rápidamente.

Tabla 7

Resultados de Tiempo de Respuesta del Prototipo

Escenario	Total de cruces	Cruces detectados	Tiempo promedio de respuesta [s]
Escenario 1	50	47	0.8
Escenario 2	75	72	0.9

En la tabla 7, se muestra el tiempo promedio de respuesta del sistema en segundos para los escenarios 1 y 2. Se indica el número total de colisiones simuladas en cada escenario, el número de cruces detectados correctamente y el tiempo promedio de respuesta correspondiente. Estos valores proporcionan una idea de la rapidez con la que el sistema

reacciona ante situaciones de posible colisión en diferentes contextos, los escenarios se mantienen según lo descrito previamente.

4.1.4 Falsos Positivos y Falsos Negativos

Evaluar la cantidad de falsos positivos (alertas incorrectas) y falsos negativos (falta de alertas en situaciones críticas). Reducir al mínimo estos valores asegura que el sistema sea confiable y útil para los conductores, los resultados obtenidos se muestran en la tabla 8.

En los escenarios 1 y 2, se han registrado casos de falsos positivos y falsos negativos al evaluar la detección y alertas del prototipo.

Tabla 8

Resultados de Pruebas de Falsos Positivos y Falsos Negativos

Escenario	Falsos Positivos	Falsos Negativos
Escenario 1	2	3
Escenario 2	1	3

Falsos Positivos, en el escenario 1, el sistema emitió 2 alertas de colisión que no correspondían a situaciones reales de colisión. En el escenario 2, se generó 1 alerta de colisión incorrecta. La presencia de falsos positivos puede atribuirse a situaciones en las que el sistema puede interpretar movimientos o patrones como potenciales colisiones cuando en realidad no lo son.

Falsos Negativos, en el escenario 1, el sistema no logró detectar 3 situaciones reales de colisión. En el escenario 2, también se dejaron de detectar 3 colisiones. Los falsos negativos son una preocupación, ya que indican que el sistema no logra alertar en momentos críticos de colisión potencial.

La presencia de falsos positivos y falsos negativos tiene implicaciones directas en la confiabilidad y eficacia del prototipo. Los falsos positivos pueden generar distracciones innecesarias para los conductores, mientras que los falsos negativos pueden poner en riesgo la seguridad al no alertar en situaciones críticas. Es importante minimizar ambos tipos de errores

a través de ajustes en algoritmos y una mayor optimización del sistema.

En última instancia, el análisis de falsos positivos y falsos negativos es esencial para iterar y mejorar el sistema, buscando un equilibrio entre la detección precisa y la reducción de alarmas incorrectas.

4.2 Resultados Obtenidos de la Implementación del Prototipo

Se analizarán en detalle los resultados obtenidos de las pruebas realizadas en el prototipo de asistencia de cambio de carril desarrollado para conductores de vehículos tipo M1.

El desarrollo del sistema de asistencia de cambio de carril ofrece beneficios en términos de seguridad vial. Al proporcionar a los conductores una alerta audible cuando se produce un desvío involuntario de carril, el sistema puede reducir la probabilidad de colisiones y accidentes relacionados con cambios de carril incorrectos. Además, este tipo de asistencia puede mejorar la confianza del conductor y la comodidad durante la conducción.

Durante estas pruebas, el prototipo fue evaluado en situaciones de conducción real. Se analizó la detección de las líneas guías y la emisión de alertas cuando el conductor se desviaba del carril. Los resultados de estas pruebas mostraron lo siguiente:

- El prototipo pudo detectar de manera efectiva las líneas guías en la carretera.
- Cuando el conductor se desvió involuntariamente de su carril, el sistema emitió una alerta audible para notificar al conductor y fomentar el regreso al carril correcto.

Los resultados obtenidos de las pruebas demuestran la funcionalidad y eficacia del prototipo de asistencia de cambio de carril. La detección precisa de las líneas guías y la emisión oportuna de alertas indican que el sistema puede desempeñar un papel importante en la mejora de la seguridad vial al prevenir cambios de carril no deseados.

El software utilizado para elaborar el código del prototipo fue Visual Studio Code. Esta plataforma proporcionó un entorno de desarrollo eficiente para crear y depurar el algoritmo de detección de líneas guías y la lógica de alerta del sistema de asistencia de cambio de carril.

Si bien las pruebas indicaron un funcionamiento satisfactorio, es importante reconocer que las condiciones de prueba pueden variar en comparación con situaciones reales en carretera.

Los resultados obtenidos de las pruebas en el prototipo de asistencia de cambio de carril respaldan su funcionalidad. El sistema demuestra ser capaz de detectar desviaciones no deseadas de carril y alertar al conductor de manera efectiva, lo que contribuye a mejorar la seguridad vial y la experiencia de conducción.

Conclusiones

Se ha logrado exitosamente a través del desarrollo y la integración de componentes específicos para la asistencia de cruce de carril en vehículos tipo M1. La selección de componentes adecuados, el diseño de algoritmos y el trabajo en campo han culminado en un prototipo funcional que demuestra la viabilidad de la asistencia vehicular en situaciones de cambio de carril.

La meticulosa selección de componentes, como cámaras de alta resolución y microcontroladores con suficiente capacidad de procesamiento, se ha basado en criterios técnicos específicos. Esto ha permitido asegurar que el prototipo tenga la capacidad necesaria para detectar carriles, vehículos y generar alertas de colisión con precisión.

El desarrollo del algoritmo ha sido un éxito, ya que el sistema es capaz de procesar las señales de la cámara y analizar los patrones de tráfico para detectar colisiones potenciales y alertar al conductor. El algoritmo ha demostrado eficacia en situaciones simuladas y reales, lo que valida su funcionamiento en una Raspberry Pi.

Las pruebas de campo han confirmado la capacidad del sistema para operar en condiciones del mundo real. La precisión en la detección de carriles, la generación de alertas y la adaptabilidad a diferentes escenarios de tráfico han sido demostradas, lo que indica que el prototipo es efectivo y confiable.

Recomendaciones

En base a los resultados obtenidos y las lecciones aprendidas durante este proyecto, se proponen las siguientes recomendaciones para futuras investigaciones y mejoras:

Mejora Continua del Algoritmo: Aunque el algoritmo actual es efectivo, se recomienda una mejora continua para aumentar su precisión y capacidad de adaptación a diferentes condiciones de conducción.

Integración en Sistemas Avanzados de Asistencia al Conductor: Explorar la posibilidad de integrar el sistema en sistemas más avanzados de asistencia al conductor, como vehículos autónomos, para maximizar su potencial de seguridad.

Pruebas en Diversas Condiciones de Conducción: Realizar pruebas en una variedad de condiciones de conducción, como diferentes climas y tipos de carreteras, para garantizar la efectividad del sistema en escenarios diversos.

Consideración de Factores Humanos: Investigar la manera en que los conductores interactúan con el sistema y cómo las alertas pueden ser diseñadas de manera más efectiva para garantizar una respuesta óptima.

En última instancia, el prototipo de asistencia vehicular de cruce de carril desarrollado en este proyecto tiene el potencial de mejorar la seguridad vial y prevenir accidentes. Su implementación futura y la continua investigación en este campo podrían contribuir significativamente a la reducción de accidentes relacionados con el cruce de carril y, en última instancia, salvar vidas en las carreteras.

Bibliografía

- Álvarez. (2020). *Automóviles que tienen los mejores sistemas avanzados de asistencia al conductor. Purosautos.*
- Daniels. (2006). *Tecnología del coche moderno. CEAC.*
- D'Cruz, Jia Zou, (2007). "Lane detection for driver assistance and intelligent vehicle applications," *Ieee xplore.*
- Diéguez. (1993). *Tecnología y responsabilidad. Revista de Filosofía.*
- Galván. (2017). *Los 6 niveles de autonomía. SG.*
- García. (2016). *Los niveles de conducción autónoma. CEA.*
- Gil. (2015). *Los sistemas de asistencia al conductor: necesarios y obligatorios. Híbridos y eléctricos.*
- León, R. A. (2019). *Diseño de un sistema de asistencia vehicular avanzado. Esime.*
- Llerena, M. F. (2020). *Innova Research Journal. Uide.*
- Machado. (2022). *Ecuador, el quinto país Sudamericano con más muertes en las vías. Primicias.*
- Regalado, J. A. (2021). *Implementación de un Sistema de Retrovisores Digitales como Medio de Asistencia para la Conducción en un Vehículo Tipo Sedán. Uide.*
- Santamaria. (2022). *La historia del mini PC más famoso del mundo. Eloutput.*
- Sauras. (2017). *Como funciona la asistencia de mantenimiento de carril. Autofácil.*
- Terrones. (2021). *Desarrollo de coches autónomos. Redalyc.*

Anexos

Código Fuente

```
import cv2
import numpy as np
import pygame
import time
import math

# Dimensiones de la ventana de visualización
ANCHO_VENTANA = 640
ALTO_VENTANA = 480

# Frecuencia y duración del pitido
FRECUENCIA_PITIDO = 1000 # Hz
DURACION_PITIDO = 500 # ms

# Ángulo máximo para considerar un cambio de dirección (en grados)
ANGULO_CAMBIO_DIRECCION = 30

# Variables para controlar el estado del carril y la alerta
carril_inicial = None
reproduciendo_pitido = False

# Función para detectar las líneas en una imagen utilizando la transformada de
Hough probabilística
def detectar_lineas(img):
    # Convertir la imagen a escala de grises
    gris = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

    # Aplicar el suavizado para reducir el ruido
    suavizado = cv2.GaussianBlur(gris, (5, 5), 0)

    # Aplicar la detección de bordes mediante el operador de Canny
    bordes = cv2.Canny(suavizado, 50, 150)

    # Aplicar la transformada de Hough probabilística para detectar líneas
    lineas = cv2.HoughLinesP(bordes, 1, np.pi / 180, threshold=50,
minLineLength=100, maxLineGap=10)

    return lineas

# Función para dibujar las líneas detectadas en la imagen original
def dibujar_lineas(img, lineas):
    # Dibujar las líneas en la imagen
    if lineas is not None:
        for linea in lineas:
            x1, y1, x2, y2 = linea[0]
```

```

        cv2.line(img, (x1, y1), (x2, y2), (0, 255, 0), 2)

    return img

# Función para calcular el ángulo entre dos líneas representadas por puntos
# finales
def calcular_angulo_entre_lineas(linea1, linea2):
    x1, y1, x2, y2 = linea1
    x3, y3, x4, y4 = linea2
    angulo1 = math.degrees(math.atan2(y2 - y1, x2 - x1))
    angulo2 = math.degrees(math.atan2(y4 - y3, x4 - x3))
    return abs(angulo2 - angulo1)

# Función para emitir el pitido
def emitir_pitido(frecuencia, duracion):
    pygame.mixer.stop() # Detener cualquier pitido previo
    sound = pygame.mixer.Sound('beep.wav') # Archivo de sonido para el pitido
    sound.set_volume(1.0) # Volumen del pitido (0.0 a 1.0)
    channel = sound.play() # Reproducir el pitido
    pygame.time.wait(duracion) # Esperar durante la duración especificada
    channel.stop() # Detener el pitido después de la duración especificada

# Inicializar Pygame
pygame.init()

# Procesamiento de video en tiempo real
cap = cv2.VideoCapture(0) # Utilizar la cámara web (cambiar el índice según
# sea necesario)

while True:
    # Leer el siguiente fotograma
    ret, frame = cap.read()

    # Detectar líneas en el fotograma
    lineas = detectar_lineas(frame)

    # Dibujar las líneas en el fotograma
    frame_lineas = dibujar_lineas(frame, lineas)

    # Mostrar el fotograma con las líneas
    cv2.imshow('Asistencia de Carril', frame_lineas)

    # Verificar si el vehículo ha salido de su carril inicial
    if lineas is not None and len(lineas) > 0:
        if carril_inicial is None:
            carril_inicial = lineas
        else:
            if carril_inicial.shape != lineas.shape:

```

```
        carril_inicial = lineas
    else:
        angulo_cambio =
calcular_angulo_entre_lineas(carril_inicial[0][0], lineas[0][0])
        if angulo_cambio > ANGULO_CAMBIO_DIRECCION:
            if not reproduciendo_pitido:
                emitir_pitido(FRECUENCIA_PITIDO, DURACION_PITIDO)
                reproduciendo_pitido = True
            else:
                reproduciendo_pitido = False
    else:
        carril_inicial = None
        reproduciendo_pitido = False

    # Salir del bucle si se presiona la tecla 'q'
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

# Liberar los recursos y cerrar las ventanas
cap.release()
cv2.destroyAllWindows()
```

