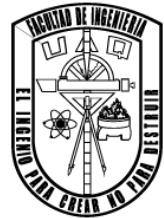




Universidad Autónoma de Querétaro

Facultad de Ingeniería

Posgrado



**DESARROLLO DE SISTEMA ADAS DE ALERTA PARA EL FRENADO DE
EMERGENCIA EWB EN VEHÍCULO AUTOMOTOR BASADO EN DEEP
LEARNING**

T E S I S

Que como parte de los requisitos para obtener el título de

Maestro en Ciencias Mecatrónica

Presenta:

Ing. Victor Jose Ortiz Granados

Dirigido por:

Dr. Salvador Martínez Cruz

San Juan del Río, Querétaro, enero de 2025

La presente obra está bajo la licencia:
<https://creativecommons.org/licenses/by-nc-nd/4.0/deed.es>



CC BY-NC-ND 4.0 DEED

Atribución-NoComercial-SinDerivadas 4.0 Internacional

Usted es libre de:

Compartir — copiar y redistribuir el material en cualquier medio o formato

La licenciatario no puede revocar estas libertades en tanto usted siga los términos de la licencia

Bajo los siguientes términos:

 **Atribución** — Usted debe dar [crédito de manera adecuada](#), brindar un enlace a la licencia, e [indicar si se han realizado cambios](#). Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que usted o su uso tienen el apoyo de la licenciatario.

 **NoComercial** — Usted no puede hacer uso del material con [propósitos comerciales](#).

 **SinDerivadas** — Si [remezcla, transforma o crea a partir](#) del material, no podrá distribuir el material modificado.

No hay restricciones adicionales — No puede aplicar términos legales ni [medidas tecnológicas](#) que restrinjan legalmente a otras a hacer cualquier uso permitido por la licencia.

Avisos:

No tiene que cumplir con la licencia para elementos del material en el dominio público o cuando su uso esté permitido por una [excepción o limitación](#) aplicable.

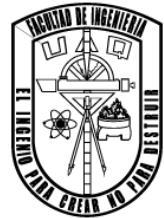
No se dan garantías. La licencia podría no darle todos los permisos que necesita para el uso que tenga previsto. Por ejemplo, otros derechos como [publicidad, privacidad, o derechos morales](#) pueden limitar la forma en que utilice el material.



Universidad Autónoma de Querétaro

Facultad de Ingeniería

Posgrado



**DESARROLLO DE SISTEMA ADAS DE ALERTA PARA EL FRENADO DE
EMERGENCIA EWB EN VEHÍCULO AUTOMOTOR BASADO EN DEEP
LEARNING**

T E S I S

Como parte de los requisitos para obtener el grado de

Maestro en Ciencias Mecatrónica

Presenta:

Ing. Victor Jose Ortiz Granados

Dirigido por:

Dr. Salvador Martínez Cruz

SINODALES:

Dr. Salvador Martínez Cruz

Presidente

Firma

Dr. Ángel Pérez Cruz

Secretario

Firma

Dr. Luis Morales Velázquez

Vocal

Firma

Dr. Juan Pablo Amézquita Sánchez

Sinodal

Firma

Dr. J. Jesús de Santiago Pérez

Sinodal

Firma

San Juan del Río, Querétaro, México

RESUMEN

Los accidentes vehiculares son ocasionados por errores humanos, puesto que son susceptibles a distracciones que restan milisegundos valiosos para realizar maniobras de evasión. Para dar solución a ello, se desarrollan los sistemas avanzados de asistencia al conductor (ADAS, por sus siglas en inglés) que delegan tareas como navegación, dirección, alerta y frenado a sistemas que integran algoritmos de aprendizaje de máquinas y profundo. En este proyecto de tesis se presenta el desarrollo de un sistema ADAS de alerta al frenado de emergencia basado en cámaras digitales comerciales, utilizando técnicas de procesamiento de imágenes y de señales, para obtener la base de datos que entrena el algoritmo de aprendizaje profundo (Deep Learning), la cual es capturada en un vehículo eléctrico con un ambiente controlado en un espacio de estacionamiento vehicular acondicionado para las pruebas necesarias bajo distintas condiciones. Se utilizan objetivos dinámicos como *dummies* de adulto, objetos estáticos como un bote de seguridad y vehículos de prueba. Con los datos obtenidos se entrena el algoritmo de Deep Learning y así clasifica la posible colisión y con qué objetivo está siendo detectado, y así generando una alarma visual que genera una colisión realizando 160 pruebas en condiciones controladas, además de una prueba en entorno no controlado, obteniendo un 91.25% de precisión en la alerta oportuna de una posible colisión.

Palabras clave: Asistencia de frenado de emergencia, ADAS, EWB, Deep Learning, procesamiento de imágenes, cámara, seguridad, sistemas autónomos.

SUMMARY

Traffic accidents are primarily caused by human error, as drivers are susceptible to distractions that reduce the valuable milliseconds needed to perform evasive maneuvers. To address this problem, Advanced Driver Assistance Systems (ADAS) have been developed, delegating tasks such as navigation, steering, warning, and braking to systems that integrate machine learning and deep learning algorithms. This Thesis Project presents the development of an ADAS emergency braking alert system based on commercial digital cameras, using image and signal processing techniques to build the database that trains the deep learning algorithm. The dataset is captured using an electric vehicle operating in a controlled environment in a parking area specially conditioned for the required tests under various conditions. Dynamic targets such as adult-sized dummies, static obstacles such as a safety cone, and test vehicles are used. Based on the collected data, the deep learning algorithm is trained to classify potential collisions and identify the type of detected object, generating a visual alarm when a collision risk is present. A total of 160 tests were conducted under controlled conditions, along with an additional test in an uncontrolled environment, achieving 91.25% accuracy in providing timely alerts of a possible collision.

Keywords: Emergency braking assistance, ADAS, EWB, Deep Learning, image processing, camera, safety, autonomous systems.

*“Si quieres llegar rápido, ve solo.
Si quieres llegar lejos, ve acompañado”*

- Desconocido

*A mis padres; Sergio y Guadalupe, por
brindarme su confianza y amor en este trayecto.*

*A mis hermanos; Sergio, Mario y Adriana, por
haberme brindado su apoyo en todo.*

AGRADECIMIENTOS

A mi familia, por ser siempre apoyarme en mis proyectos profesionales y personales. A mis padres; por siempre encontrar la manera de llevar comida a la mesa y que mi única preocupación sea el estudio y por educarme con los valores que ellos me inculcaron y ser el hombre que he llegado a ser, no lo hubiera logrado sin ustedes.

Al Dr. Salvador Martínez Cruz, por apoyar técnica y personalmente en el desarrollo de este proyecto de Tesis y en mi desarrollo personal y profesional.

A todos los profesores que tuve el gusto en encontrarme en esta etapa, me llevo algo de todos ellos para aplicarlo en lo que sea que se me presente en el futuro.

A mis compañeros; Pablo, Daniel, Osiris, Rafael y Cinta, por hacer más ameno y divertido el trayecto en la universidad. Mucho éxito a todos ustedes donde quiera que la vida los envíe. Espero verlos triunfar en lo que sea que se propongan.

ÍNDICE

CAPÍTULO I: INTRODUCCIÓN.....	14
1.1 Antecedentes	15
1.1.1 Sistemas Avanzados de Asistencia al Conductor	15
1.1.2 Sistemas ADAS desarrollados en vehículos modernos.....	17
1.1.3 Velocidad de Crucero Adaptativo (ACC)	17
1.1.4 Mantenimientos de Carril	18
1.1.5 Detección de punto ciego	19
1.1.6 Asistente para Frenado de Emergencia:	21
1.2 Justificación	25
1.3 Descripción del problema.....	26
1.4 Hipótesis	27
1.5 Objetivos.....	27
1.5.1 Objetivo General.....	27
1.5.2 Objetivos específicos.....	27
1.5.3 Planteamiento General.....	29
CAPITULO II: FUNDAMENTACIÓN TEÓRICA.....	32
2.1 Hardware.....	32
2.1.1 Vehículo Eléctrico	32
2.1.2 Cámara Web Logitech© HD Pro C920s	33
2.1.3 Dispositivos de Alerta	35
2.1.4 Pista de pruebas	35
2.1.4 Maniquís de pruebas (<i>Dummies</i>)	36
2.2 Software	37
2.2.1 Matlab©	38

2.2.2	Lenguaje Python©	38
2.3	Técnicas de procesamiento de datos	39
2.3.1	Sistemas de procesamiento de imágenes	39
2.3.2	Deep Learning (DL)	41
2.3.3	Aprendizaje supervisado.....	41
2.3.3	Redes Neuronales Artificiales	42
2.3.4	Redes Neuronales de Convolucionales.....	44
CAPÍTULO III: DESARROLLO DE SISTEMA ADAS PARA ALERTA DE FRENADO EWA		50
3.1	Captura de base de datos.....	50
3.1.1	Configuración de Vehículo Hospedador	50
3.1.2	Adecuación de pista de pruebas para captura.....	52
3.1.3	Captura de clase “peatón”.....	53
3.1.5	Captura de clase “Vehículo”.....	55
3.1.6	Captura de “Ruta Libre”	56
3.1.7	Algoritmo de captura automatizada de imágenes.....	57
3.2	Topología de CNN para clasificación.....	60
3.3	Histograma para filtrado de señal de salida	62
3.4	Algoritmo de EWA	64
3.5	Puesta experimental	66
3.5.1	Casos de pruebas: Objetivos.....	68
3.5.2	Casos de pruebas: Ruta Libre	70
CAPÍTULO IV: PRUEBAS Y RESULTADOS		71
4.1	Resultados del entrenamiento y validación de la CNN	71
4.2	Resultados del desempeño del sistema en pista de pruebas.....	73

4.2.1	Resultados de desempeño del sistema en la clase Peatón	75
4.2.2	Resultados de desempeño del sistema en la clase Infraestructura.....	78
4.2.3	Resultados de desempeño del sistema en la clase Vehículo.....	80
4.3	Comparación del sistema con trabajos similares	82
4.4	Pruebas realizadas en vehículo eléctrico.....	83
CAPÍTULO V: CONCLUSIONES		86
BIBLIOGRAFÍA		88

INDICE DE FIGURAS

Figura 1. Ejemplo de ACC (Neuman, 2024).....	17
Figura 2. Ejemplo gráfico de sistema LK (Neuman, 2024).	18
Figura 3. Ejemplo representativo de BSD (Neuman, 2024).....	20
Figura 4. Diagrama del planteamiento general de la metodología a implementar.....	29
Figura 5. Vehículo eléctrico de la facultad de ingeniería de la Universidad Autónoma de Querétaro (FI, 2024).....	33
Figura 6. Web Logitech© HD Pro C920s	34
Figura 7: a) Control de crucero adaptativo y b) sistema Driving Assistant implementados en los vehículos BMW© serie 3 Berlina y Touring (BMW, 2024).	35
Figura 8: Pista de pruebas para sistemas automotrices (Shutterstock, 2024).....	36
Figura 9. <i>Dummies</i> en prueba de choque (Jaśkiewicz et al., 2021).....	37
Figura 10: interfaz de sitio oficial de Python©	39
Figura 11. Diagrama a bloques de un sistema de visión (Gonzalez & Woods, 2018).....	40
Figura 12. a) Perceptrón simple y b) arquitectura de una DNN.....	43
Figura 13. Componentes de una CNN.....	44
Figura 14. Representación visual de cálculos primarios de un Kernel (Taye, 2023).....	46
Figura 15: Efectos de diferentes Kernels en una imagen (Taye, 2023).	47
Figura 16: Ejemplos de operaciones de pulido.	48
Figura 17: Acondicionamiento de vehículo hospedador con sistema de adquisición de imágenes.	51
Figura 18: a) Acondicionamiento de la pista de pruebas con las regiones de detección en el estacionamiento de la Facultad de Ingeniería y b) ejemplificación de escenario de peatón caminando para la captura de imágenes	52
Figura 19. Captura de imágenes de clase “peatón” a) una persona desplazándose, grupo de b) 2 personas, c) 3 personas, d) 5 personas desplazándose y e) dummie estático.	54

Figura 20: Clase de infraestructura capturada variando ángulo de captura	55
Figura 21: Vehículos capturados para la clase “Vehículo”.....	56
Figura 22. Zona de adquisición de imágenes para clase ruta libre	56
Figura 23: Diagrama de flujo para captura de imágenes de manera automática.	57
Figura 24: Secuencia de imágenes capturadas desde el punto de vista del vehículo hospedador.	
59	
Figura 25: Imágenes capturadas en la etapa de adquisición de datos.	60
Figura 26: Validación de acondicionamiento de imágenes por medio de una red Neuronal Convolucional.....	61
Figura 27: Arquitectura de CNN para detección de 4 clases.	62
Figura 28: Diagrama de flujo de función “Histograma” para el filtrado de señal.	63
Figura 29: Diagrama de flujo de sistema EWA.	65
Figura 30: Puesta experimental propuesta para el sistema EWA.....	66
Figura 31: a) Pista de pruebas acondicionada para validación del sistema EWA y b) ejemplo real de escenario de prueba con clase “Peatón”	67
Figura 32: Consideraciones adicionales para que una prueba sea a) exitosa o b) fallida por clasificación incorrecta de objetivo o c) fallida por no alertar en el caso de prueba con “objetivos”.....	69
Figura 33: a) Prueba exitosa de conducción segura o ruta libre y b) prueba fallida al detectar un objetivo en una ruta libre.	70
Figura 34: a) Precisión de la CNN implementada con las imágenes adquiridas y b) pérdida durante el entrenamiento	71
Figura 35. Matriz de confusión con 20% de imágenes por clase para prueba de CNN.	72
Figura 36. Interfaz usada para las pruebas del sistema a) inicio de la prueba a 40 m de distancia del objetivo y b) Parte final de la prueba en zona de detección con el vehículo hospedador moviéndose a una velocidad de 15 m/s.	74
Figura 37. TTC obtenido por el sistema EWA en la clase “Peatón”.....	76

Figura 38. TTC obtenido por el sistema EWA en la clase “Peatón”.....	78
Figura 39: Resultados de tiempo estimado a colisión en pruebas.....	80
Figura 40. a) Vehículo Eléctrico Acondicionado para pruebas en escenarios reales.....	83
Figura 41. Circuito donde se conduce el vehículo eléctrico dentro de las instalaciones de la Universidad Autónoma de Querétaro, Campus San Juan del Río.....	84

Índice de Tablas

Tabla 1: Resultados de tiempo de detección en clase “peatón” con una distancia inicia de 40m a velocidades definidas.....	77
Tabla 2: Resultados de tiempo de detección en clase infraestructura con una distancia inicia de 40m a velocidades definidas.....	79
Tabla 3: Resultados de tiempo de detección en clase infraestructura con una distancia inicia de 40m a velocidades definidas.....	81
Tabla 4. Comparativa de sistema EWB con trabajos similares	82

CAPÍTULO I: INTRODUCCIÓN

Los Sistemas Avanzados de Asistencia al Conductor (ADAS, por sus siglas en inglés) son, hoy en día, una línea de investigación en auge, debido a la necesidad de dotar a los vehículos modernos con sistemas que ayuden a minimizar la posibilidad de sufrir un accidente vial, que ponen en peligro tanto a los bienes materiales (los propios vehículos, infraestructura vial, entre otros), y a las vidas humanas involucradas (peatones, conductores, pasajeros, etc.), que es lo más importante. Estos sistemas que se implementan adquieren información del entorno con sensores como ultrasonido, visión artificial, radares o LiDAR ("Light Detection and Ranging", por sus siglas en inglés), que junto con algoritmos de Aprendizaje Profundo (mejor conocido por sus siglas en inglés como "Deep Learning" (DL)), pueden desarrollarse tecnologías como la detección de puntos ciegos, mantenimientos de carril, control de crucero adaptativo, evasión de colisiones en las intersecciones, y la asistencia al frenado de emergencia. Debido a esta necesidad, el presente trabajo de tesis presenta el desarrollo de un sistema de alerta al frenado de emergencia con la integración de cámaras comerciales, procesamiento de imágenes y señales, así como una tarjeta de para integrar un sistema embebido. Se implementa en un vehículo eléctrico con un ambiente controlado en un espacio de estacionamiento vehicular acondicionado para las pruebas necesarias bajo distintas condiciones, con una serie de escenarios que se pueden presentar en la vida diaria, para probar y validar el funcionamiento de asistencia de frenado de emergencia.

1.1 Antecedentes

En esta sección, se describen los trabajos más relevantes de los últimos años relacionados al presente trabajo de tesis.

1.1.1 Sistemas Avanzados de Asistencia al Conductor

Los ADAS ayudan a atacar la problemática de los accidentes viales; si bien no a erradicarla, disminuyen en gran medida las posibilidades de accidentes vehiculares ocasionados por factores humanos. Estos sistemas tienen la función de auxiliar al conductor mientras navega en el vehículo, dotando de información del medio para la toma de decisiones (Ziebinski et al., 2017). Los ADAS analizan el entorno por medio de sensores y cámaras, implementan algoritmos avanzados con software para analizar la información recabada y generan alertas,

activar un asistente de conducción o de frenado, o tomar acciones autónomas para prevenir accidentes que pongan en riesgo a peatones o tripulantes de vehículos. Los ADAS pueden contener diferente cantidad de sensores dependiendo del modelo del vehículo, nivel de equipamiento o versión (Neumann, 2024), algunos sensores utilizados son:

Radares: Dispositivos que usan ondas electromagnéticas para detectar la presencia de objetos en el espacio, enviando una serie de ondas de microondas y detectándola cuando es reflejada por un objeto. Las ondas son reflejadas de manera no uniforme dependiendo del material, la forma, y su predisposición para reflejar ondas electromagnéticas.

Detección y medición de distancia por luz: LiDAR por sus siglas en inglés, tiene un principio de funcionamiento similar al radar, con la diferencia que utilizan un haz de luz con longitud de onda específica, en lugar de ondas de micro ondas. Este principio de funcionamiento permite una mejor precisión; reconociendo detalles desde centímetros hasta 100 m. Aunque su resolución se ve afectada dependiendo de la cantidad de LiDAR utilizados, el incorporar una mayor cantidad de LiDAR.

Ultrasonido: Son ondas de sonido de alta frecuencia imperceptibles para los humanos (de 20 kHz a 1 GHz). Utiliza el mismo principio lanzando ondas ultrasónicas y recibidas después de una reflexión con un objeto, calculando así la distancia a la que se encuentra. Los sensores ultrasónicos tienen un rango limitado de medición limitado, desde centímetros hasta varios metros. Esta limitante se compensa colocando varios sensores alrededor del vehículo. Estos sensores se usan en sistemas de asistencia al estacionarse, detección de puntos ciegos (BSD, por sus siglas en inglés), y asistencia en maniobras en bajas velocidades.

Cámaras o arreglos de cámaras: Es la solución más popular, barata y simple que se utilizan en los ADAS, y se pueden utilizar varias cámaras para obtener una vista de 360° alrededor del vehículo. La posibilidad que brindan las cámaras de adquirirlas e instalarlas por los usuarios, y de implementar algoritmos de procesamiento de imágenes, ha contribuido al uso de cámaras para el desarrollo de los ADAS. Las cámaras pueden detectar patrones para identificar vehículos como peatones o amenazas, y realizar acciones que respondan a este riesgo. Las cámaras son utilizadas en sistemas de asistencia al estacionamiento, monitoreo del comportamiento del conductor, y en funciones de velocidad de crucero adaptativo. o ACC por sus siglas en inglés.

1.1.2 Sistemas ADAS desarrollados en vehículos modernos.

Los vehículos modernos son equipados con estos sistemas que permiten mejorar la seguridad, confort, y eficiencia al conducir. Su desarrollo ha mejorado conforme la tecnología avanza y se realizan avances software como algoritmos de clasificación, como en hardware en el desarrollo de sistemas embebidos y de sensores (Neumann, 2024). A continuación, se mencionan las funciones relevantes, tecnologías implementadas cómo se utilizan dichos sensores para la implementación de dichas funciones ADAS.

1.1.3 Velocidad de Crucero Adaptativo (ACC)

El ACC es un sistema de control longitudinal que modifica la velocidad del vehículo hospedador, aumentándola o disminuyéndola en función de la proximidad que se encuentren en frente de él o en su parte posterior. En la **Figura 1** se muestra un ejemplo gráfico de esta función.

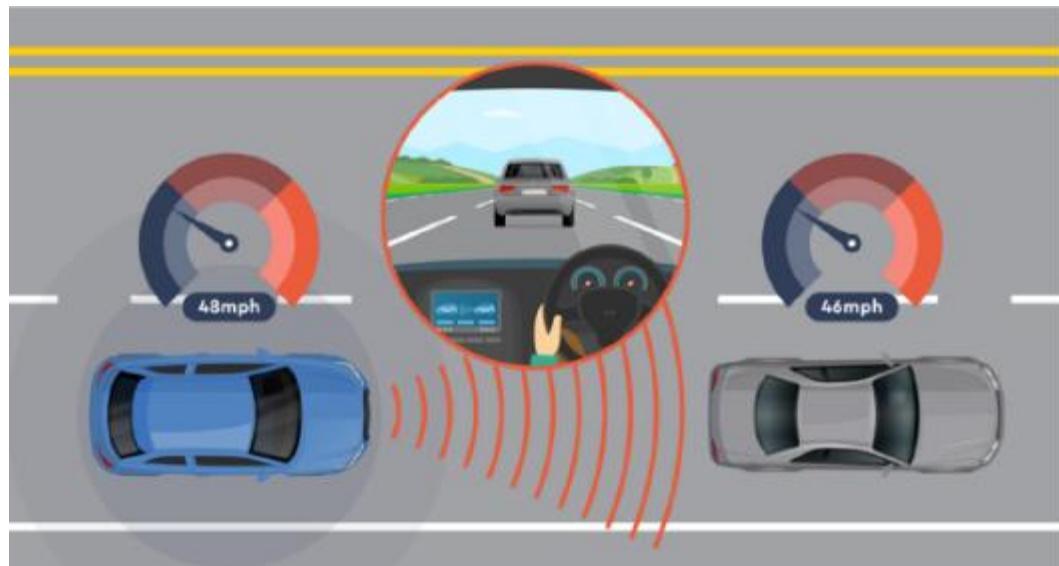


Figura 1. Ejemplo de ACC (Neuman, 2024).

Los métodos utilizados son los basados en modelos de control predictivo (MPC) como el desarrollado por Wei et al. (2025), que utiliza pesos dinámicos que se modifican en función del estado del tráfico y la velocidad del hospedador mejorando el confort y la predicción

dinámica para mayor estabilidad en entornos variables a costa de un mayor costo computacional. Manolis et al. (2020) desarrollan una estrategia de ACC en tiempo real que ajusta la velocidad mediante control jerárquico mejorando el tiempo de respuesta y estabilidad del vehículo en maniobras rápidas en entornos reales aplicable en autopistas, con eficiencia energética y confort. K. Lee & Lee (2025) utilizan un PD de tiempo constante, estableciendo condiciones necesarias para selección de sus ganancias y logrando estabilidad individual y de cadena, validado en simulación. Z. Yang et al. (2021) Utilizan un control jerárquico con MPC y un control activo de rechazo de perturbaciones, mejorando así el seguimiento con menos oscilaciones. Yu et al. (2025) utilizan un MPC con un Predictor Smith para compensar retardos para un modelo longitudinal en simulación, reduciendo oscilaciones y mejorando estabilidad. En los desarrollos de estos sistemas se encuentran que la mayoría de ellos dependen de radares y sensores LiDAR, además de uso de MPC para la implementación en simulaciones.

1.1.4 Mantenimientos de Carril

Esta función (**Figura 2**) tiene como objetivo incrementar la seguridad para ayudar a mantener el vehículo en el carril, corrigiendo la dirección.



Figura 2. Ejemplo gráfico de sistema LK (Neuman, 2024).

Investigaciones como el desarrollado por Cheng et al. (2025), donde utilizan aprendizaje por refuerzo profundo con enfoque fin a fin para decidir el ángulo de dirección a partir de imágenes de cámara, implementándolo en simulación y entornos reales a escala, adquiriendo una base de datos propia, obteniendo una detección LK eliminando la dependencia de modelos cinemáticos complejos. Xie et al. (2023), utiliza el filtrado adaptativo y detección online que actualiza parámetros dinámicamente, adquiriendo imágenes con visión monocular y fusión de datos para ajustar coeficientes de líneas de carril en tiempo real, identificando los carriles en zonas urbanas y autopistas. Y. Jeong, (2022) utiliza una red neuronal recurrente con celdas de aprendizaje de tiempo corto y largo, recolectando imágenes en autopistas. Redujo desviaciones laterales y mejora la respuesta en presencia de vehículos adyacentes e introduciendo un enfoque interactivo que imita las decisiones humanas. También Na et al. (2025) muestran su metodología que realizan detección de carriles con librerías como YOLOv5 y adquisición de imágenes con cámaras internas, externas frontales y traseras, junto con aumento de datos. Como se muestra en esta tecnología, las cámaras brindan una solución económica y fiable en entornos de realidad virtual para implementación a escala o en entornos reales como carreteras, con la desventaja de que todavía dependen de sensores LiDAR para su implementación, encareciendo la implementación de estos sistemas.

1.1.5 Detección de punto ciego

Otro sistema de asistencia avanzado es el sistema de detección de punto ciego (BSD, por sus siglas en inglés). Esos sistemas son designados para minimizar el riesgo de colisión vehículos estándar, pequeños o peatones en puntos difíciles de detectar por el conductor, conocidos como puntos ciegos (**Figura 3**).

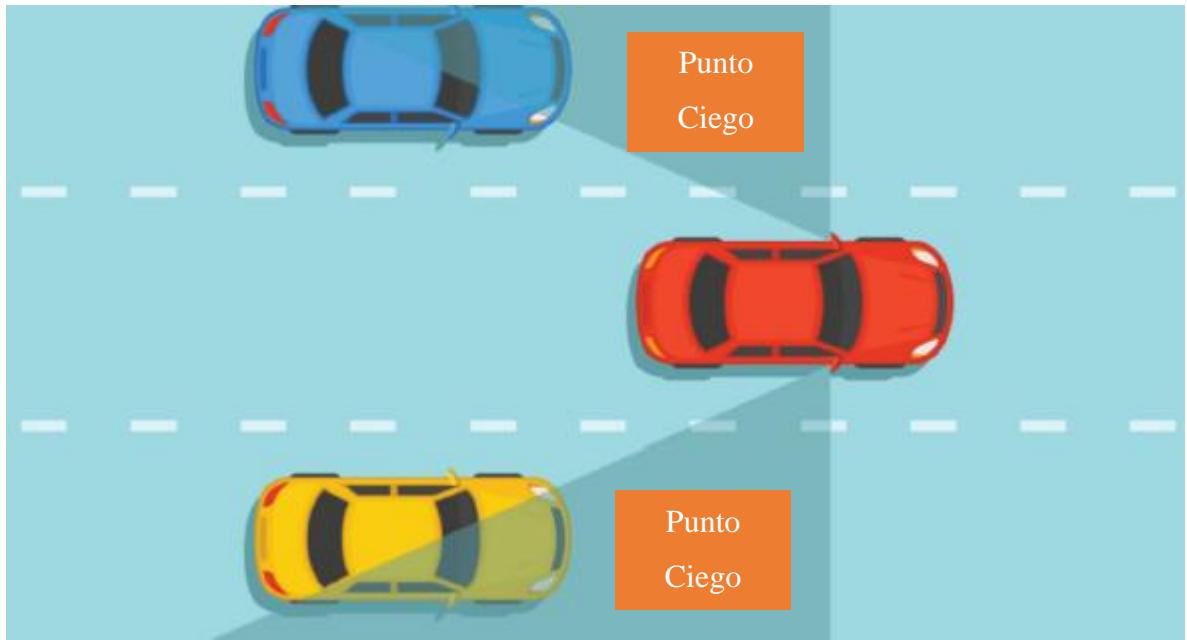


Figura 3. Ejemplo representativo de BSD (Neuman, 2024).

Y. Lee & Park (2025) implementan una combinación de funciones de BSD y LK, incorporando una cámara trasera con YOLOv9 con un filtro de Kalman para identificación de carriles y de objetos, adquieren una base de datos de 12,537 imágenes en carreteras urbanas y autopistas con una cámara gran angular estándar. Muzammel et al. (2022) fusionan arquitecturas de redes neuronales convolucionales (CNN, por sus siglas en inglés) ResNet50 + ResNet101 y redes CNN de su autoría integradas a una red neuronal recurrente rápida. Utilizan una base de datos de 3,000 imágenes propias ubicadas en espejos laterales de buses y una base de datos llamada LISA de vehículos. Kim et al. (2023) desarrollan un sistema BSD basado en radar de onda continua modulada en frecuencia con antenas de microstrip para cubrir la desventaja de las cámaras cuando se utilizan en climas de niebla/lluvia en entornos de realidad virtual y escenarios reales básicos. H.-S. Jeong & Kim (2025) utilizan un anillo de 8 sensores ultrasónicos junto con algoritmos de filtrado y compensación ambiental en entornos de lluvia y niebla, alertando en lluvia intensa a 1 m y en niebla a 2 m. Zhao et al. (2019) implementan una CNN ligera bloque tipo Sep-Res-SE de clasificación binaria con una base de datos propia de 10,000 imágenes en un vehículo lateral, dando solución a la dependencia de sensores costosos como radares o LiDAR. Se puede apreciar que las cámaras son una opción viable, para el

desarrollo de detección de objetivos, además de dotar de escenarios específicos con la adquisición de bases de datos para diferentes entornos.

1.1.6 Asistente para Frenado de Emergencia:

EBA (por sus siglas en inglés) es un sistema que reacciona automáticamente en situaciones de frenado repentino si una amenaza por colisión es detectada, reduciendo los tiempos de reacción del conductor o del mismo sistema como las cifras registradas en países como el Reino Unido se estima que la implementación de estos sistemas reduciría en un 23.8% sus accidentes de tránsito (Masello et al., 2022), y un 13.2% en China (Tan et al., 2020). L. Yang et al. (2022) plantean los procesos que involucran como una etapa normal, advertencia temprana (alerta visual y auditiva), y frenado automático. Identifican los factores que afectan el desempeño de los EBA como la tecnología y forma del vehículo (sensores, ángulo de visión, errores del sistema, retardos de frenado, capacidad de deceleración y algoritmos implementados), el estado del conductor (estilo de conducción, confort, destreza) y factores del entorno (clima, iluminación, tipo de terreno, pendientes). En la literatura reciente se abarcan distintos enfoques que abarcan desde arquitecturas predictivas hasta modelos híbridos de control. En la literatura reciente se cubren distintos enfoques que abarcan desde arquitecturas predictivas hasta modelos híbridos de control. Shaout & Castaneda-Trejo (2025), utilizan un control difuso adaptativo modelado en escenarios y validado por simulación, como alternativa más flexible que los PID tradicionales, logrando frenados más estables y reducción de distancia de detención y menor bloqueo de ruedas teóricamente, también Gunjate & Khot (2023) utilizan el control difuso y modulación por ancho de pulso (PWM) para la integración de EBA a Sistemas de Antibloqueo de Frenos (ABS, por sus siglas en inglés), recabando información y haciendo simulaciones comparativas encontrando una reducción en distancia de frenado con mayor estabilidad reduciendo una distancia de frenado a 26 m con una velocidad de 80 km/h. Hu et al. (2025) realizan un trabajo en un entorno con múltiples escenarios de tráfico simulados de evasión de colisión en el cual desarrollan un modelo de evasión de colisiones proactivo basado en reglas implementado en un modelo de distancia de seguridad para vehículos frontales, con un control de velocidad de crucero adaptativo y aprendizaje por aprendizaje de refuerzo profundo, logrando tasa de éxito en evitación de colisiones frente a modelos tradicionales. Deo et al. (2021) realizan una comparación en las arquitecturas de fusión de sensores centralizada (OCSF) y descentralizada (ODSF) con cámaras y LiDAR, utilizaron el

algoritmo YOLOv3 para detectar objetos en imágenes, y DBSCAN para identificar objetos 3D en cajas y un filtro de Kalman no lineal, usando una base de datos KITTI y validando su funcionamiento en el entorno virtual ROS, generando una alerta ante la presencia de un objeto enfrente del vehículo a 5 m. Teniendo como resultado una precisión del 64% de precisión promedio media (mAp) con OCSF y un tiempo de latencia de 32.5 ms, y una precisión de 70% mAp con ODSF y una latencia de 48.5 ms. Esta investigación genera un aporte a la combinación de sensores como cámaras y LiDAR pero no toman a consideración la velocidad del vehículo, además de dejar su investigación solo en simulación. Losada et al. (2023) proponen un sistema combinado llamado OPREVU-AES que integra un frenado autónomo de emergencia y una dirección automática de emergencia (AES) para maniobras evasivas, y se integran estos para tomar decisiones como frenar o esquivar según sea la situación. Se simula un Hyundai Ioniq 2020 en Carsim calibrando su respuesta de frenado dinámico y dirección, definiendo toma de decisiones a velocidades de 40-70 km/h y distancias de 12-24 m según la velocidad, integran un árbol de decisión basado en datos de Realidad Virtual (VR) de peatones que permitía al sistema decidir entre frenar o esquivar según la reacción del peatón, utilizando sensores como cámaras y radares para detectar carriles, puntos ciegos y obstáculos laterales en su entorno virtual. Usaron escenarios que se presentaron en 40 accidentes peatonales reales en Madrid y se recrearon estos en una herramienta de reconstrucción de accidentes reales llamada PCCrash considerando 3 casos: sin ADAS, con frenado automático y su sistema OPREVU-AES. En el caso sin ADAS tuvo una efectividad de 0% (por ser el control), con el AEB comercial tuvo un 53% de efectividad, y con su sistema OPREVU-AES lograron una disminución de accidentes hasta el 77.8%. Por su parte, Dai et al. (2024) enfatizan lo importante que es la detección de objetos para vehículos autónomos, proponen detección de objetos con sensores LiDAR y cámaras digitales. Realizan la detección con YOLOv5 para imágenes y *PointPillars* para nubes de puntos, intersección de unión, teoría de Dempster-Shafer mejorada, y finalmente seguimiento con un DeepSORT mejorado con Filtro de Kalman sin estela. Utilizan una base de datos con imágenes propias de alrededor 700 imágenes de un recorrido en un campus universitario, además de utilizar la base de datos KITTI, obteniendo un 93% de efectividad en detección de vehículos y 95.4% de peatones de día, y de noche 94.1% de autos y 92.5% peatones de noche. Si bien se obtuvieron métricas prometedoras, no mencionan si se piensa aplicar en algún sistema ADAS. Gulino et al. (2023) analizan un sistema ADAS adaptativo que combina

frenado y dirección para minimizar el riesgo de lesiones en escenarios críticos de “salida de carril” con colisión inminente en un entorno virtual. La simulación consta de dos vehículos desplazándose a 50 km/h, se realiza el monitoreo de sensores LiDAR y cámaras de un vehículo “Ego” con un algoritmo denominado “Software en el lazo” el cual monitorea tres casos: Sin intervención, frenado automático de emergencia (AEB, por sus siglas en inglés) y Lógica adaptativa para disminuir el riesgo de lesión, reudiendo un riesgo de lesión hasta 40% comparado con el AEB. Aoki et al. (2024) atacan la problemática de los accidentes ocasionados en intersecciones debido a los giros a la derecha (cerca de un 40% de los accidentes en cruces), e identifican que el problema ocurre cuando otro vehículo aparece repentinamente. Proponen un sistema de asistencia al conductor proactivo, que predice el riesgo de colisión en una curva triclothoidal (Un giro continuo del vehículo sin cambios bruscos de curvatura) implementado en un vehículo hospedador que tiene un sensor LiDAR montado en la esquina delantera derecha, El sistema proactivo modifica la velocidad del vehículo de manera gradual, calculando velocidades críticas como la velocidad de seguridad y la velocidad de escape en función de si el vehículo se encuentra en la zona de riesgo que ellos definen, realizan las pruebas en simulación 441 veces en el software IPG CarMaker® con escenarios distintos, variando la velocidad que un vehículo irrumpie el trayecto del hospedador (30-50 km/h) y su posición inicial (0 - 40m), logrando evitar un 100% de las colisiones en el escenario propuesto. Lai & Yang (2023) detectan el problema que tienen los sistemas automáticos de frenado de emergencia (AEB) actuales pueden evitar colisiones frontales, pero en carreteras curvas ven comprometida la pérdida de estabilidad lateral, debido a esto desarrollan un sistema de control integrado longitudinal y lateral para vehículos inteligentes, el cual combina un AEB con un asistente de mantenimiento de LK. El AEB utiliza el parámetro de tiempo estimado a colisión (TTC) para determinar un frenado óptimo y decidir si emitir una alarma audible si es menor a 2.6 s o mayor a 1.6 s, o si es menor que 1.6 s, activar gradualmente el freno. Mientras que el LK utiliza la desviación lateral y el ángulo de guiñada (yaw, por su nomenclatura en inglés). Realizan pruebas del sistema AEB en solitario, AEB y LK por separado, e integrados con simulaciones en Matlab/Simulink–PreScan–CarSim®, a velocidades de 50 y 60 Km/h, con radios de curvatura de 60, 90 y 120 m, teniendo un total de 18 pruebas, teniendo como resultado la evasión de colisiones frontales en todos los escenarios analizados, pero con el AEB en solitario teniendo un desplazamiento lateral máximo de 1.72 m, el AEB y LK independientes obtuvieron

un desplazamiento lateral de 0.29 m y el AEB-LK integrados logró una desviación lateral de 0.21 m. Wang et al. (2025) desarrollaron un sistema de detección tridimensional orientado a vehículos autónomos mediante la fusión de sensores LiDAR y cámaras, para mejorar la precisión en la identificación y localización de objetos al combinar imágenes 3D obtenidas con LiDAR e imágenes 2D capturada por la cámara. Para lograrlo, implementaron una arquitectura de aprendizaje profundo que integra dos redes especializadas: PointNet++, para extraer características espaciales de las nubes de puntos, y ResNet-18, para obtener características visuales de las imágenes. Ambas salidas se combinaron a través de un módulo de atención SENet, que pondera los canales más relevantes antes de ingresar a una red de detección basada en puntos centrales, capaz de estimar el centro, orientación y dimensiones de cada objeto. El modelo fue validado en el conjunto de datos nuScenes y posteriormente probado en un vehículo real equipado con un LiDAR RoboSense RS-Helios y una cámara SONY IMX264, alcanzando una precisión promedio (mAP) del 64.5 % y un puntaje de detección de nuScenes (NDS) de 63.7, con un tiempo de inferencia de 147 ms por predicción. En su trabajo de Liu et al. (2025) se aborda la necesidad de contar con métodos de evaluación más precisos y adaptativos para los sistemas de AEB en vehículos inteligentes conectados, ya que enfoques tradicionales se basan en ponderaciones estáticas que no toman en cuenta la variación del entorno (como la velocidad del vehículo o la adherencia del pavimento), ni vinculan los modelos teóricos con pruebas experimentales reales. Por ello, los autores proponen un método integral de evaluación basado en el Proceso Analítico Jerárquico de Peso Dinámico (DWAHP) acoplado a una plataforma de Vehículo-en-el-lazo (VIL) que permite ajustar los pesos de los indicadores de seguridad, fiabilidad y confort según las condiciones dinámicas de conducción. Realizaron una co-simulación entre MATLAB/Simulink y CarSim, modelando escenarios de tráfico tipo E-NCAP, realizan una validación experimental en una estación de pruebas VIL como en pruebas reales con un Volvo S90L, equipado con el sistema denominado “City Safety” de fusión cámara-radar y sensores ópticos Kistler para medir velocidad, aceleración y distancia al obstáculo. En las pruebas, se utilizó un maniquí móvil que simulaba el cruce de un peatón en una intersección, mientras el vehículo circulaba a velocidades de 20, 30 y 40 km/h sobre pavimento seco. Los resultados mostraron que el sistema AEB evitó completamente la colisión a 20 y 30 km/h con tiempos de intervención de 1.79 s y 1.88 s, pero a 40 km/h impactó al peatón con una velocidad residual de 20.7 km/h, mostrando la influencia de la velocidad en la

capacidad de respuesta del sistema. En conjunto, el método DWAHP-VIL logró una consistencia del 99.5% entre simulación y prueba real (desviación global del 0.51%), demostrando su eficacia para evaluar el desempeño del AEB bajo condiciones dinámicas de conducción. Una vez revisado el estado del arte de las tecnologías en sistemas ADAS, además de los sensores, funciones como EBA que es una parte objeto de estudio de este Trabajo de Tesis. Se identifica la necesidad de desarrollar un sistema EWA que identifique y alerte sobre posibles situaciones de riesgo como la presencia de objetivos que puedan presentar una posible colisión frontal para, en un futuro, poder realizar un sistema un sistema activo como el EBA.

1.2 Justificación

De acuerdo con la Organización Mundial de la Salud (OMS), se estima que hubieron 1.19 millones de muertes a causa de accidentes de tránsito en 2021; alrededor de 15 accidentes por cada 100,000 habitantes. Según datos del 2019 de la OMS, los accidentes de tránsito siguen siendo la principal causa de muerte en personas jóvenes de entre 5 a 29 años, y la doceava causa de muerte considerando todo el espectro de edades (World Health Organization, 2023). Por su parte, en México, en el periodo de 2015-2020 (Salud, 2021), presenta un número de 13,630 siniestros, donde los peatones muestran ser los más afectados con mayor porcentaje de fallecimientos, 2,536 atropellamientos fatales (18.6 %), seguidos por ciclistas con 111 (0.8%) y otras con 27 (0.2%). Además, se considera que muchos de estos accidentes son causados por los propios errores humanos, ya sea por falta de concentración en la conducción, la alteración de sus sentidos debido al cansancio o la ingesta de sustancias, o por la ocurrencia de eventos imprevistos. En consecuencia, los conductores no logran tomar acciones correctivas a tiempo cuando reciben un estímulo visual, teniendo en cuenta que el tiempo promedio de reacción en humanos es de aproximadamente 250 a 300 ms, tomando en cuenta que tienen su total atención en la conducción (Jain et al., 2015). Con estos datos y el estado del arte, se muestra que el desarrollo de los sistemas ADAS, en especial los sistemas de que generen alertas ante posibles impactos o eviten colisiones, es una necesidad para crear vehículos más seguros y salvaguardar la vida de los usuarios (conductores) y de los peatones, delegando las tareas de control o toma de decisiones a dichos sistemas inteligentes. Estos sistemas integran tecnología de vanguardia como lo son los algoritmos de clasificación, redes neuronales, sistemas embebidos, sensores de visión, proximidad, radares, etc. Por esto, el presente proyecto de Tesis se centra en el desarrollo de un método para un sistema ADAS de Alerta de Frenado de Emergencia (“EWB: Emergency

Warning Breaking”, por sus siglas en inglés), incorporado al vehículo eléctrico de la Facultad de Ingeniería, de la Universidad Autónoma de Querétaro, campus, San Juan del Río, para detectar obstáculos estáticos (conos y vehículo falso) y dinámicos (*dummies* de peatones), el cual robustecerá los sistemas del vehículo dotándolo de cierto nivel de autonomía y además de realizarlo en una plataforma de código abierto y accesible económicamente en comparación con los sistemas desarrollados actualmente, cabe mencionar que la etapa de pruebas del sistema mayormente se realizaran en un vehículo de pruebas bajo condiciones controlada antes de realizar una implementación en el vehículo eléctrico. Mediante la validación del sistema se abrirá una nueva línea de investigación en la institución que permita promover el desarrollo de tecnologías innovadoras en esta área, y contribuir a la investigación para disminuir la tasa de accidentes y siniestros que afectan a la sociedad. También, el proyecto de Tesis aporta a los programas nacionales estratégicos (PRONACES) en el apartado de Salud y Seguridad Humana, al generar un sistema de asistencia de alerta para el frenado de emergencia.

1.3 Descripción del problema

En México, la mayoría de los accidentes son ocasionados por errores humanos, basado en datos del informe de seguridad vial de la secretaría de salud (Salud, 2021). Los sistemas de frenado de emergencia autónomos han demostrado que, en efecto, han reducido la cantidad de incidentes con peatones (Tan et al., 2020b). A pesar de esto y de la literatura revisada, se identifican que las principales problemáticas que se presentan para el desarrollo de este proyecto de Tesis es el tiempo de cómputo al utilizar algoritmos de DL en investigaciones similares es un factor importante para considerar, como también lo es el tiempo de respuesta desde que el sistema envía la alerta visual hasta la acción humana. También, que el desarrollo de los sistemas ADAS se concentran mayoritariamente fuera de México, debido a que muchos de estos requieren una gran cantidad de inversión económica, factor que, en la región de Latinoamérica, y en México, merma la posibilidad de que la investigación pueda desarrollarse, en su mayoría son las industrias privadas las que invierten en la investigación de estos, además que esa investigación permanece restringida como capital intelectual. Entonces, para resolver esta problemática, se han identificado los siguientes desafíos: 1) Contar con un tiempo de respuesta oportuno del sistema EWB, siendo este el TTC, que es el tiempo desde que se genera la alerta, hasta que se realiza la acción de frenado determinado en un valor igual o mayor a 1.2 s (ENCAP, 2023) para pruebas definidas por el Programa Europeo de Evaluación de Autos

Nuevos (ENCAP, por sus siglas en inglés). 2) Construir un sistema con elementos de bajo costo económico, en comparación con los sistemas desarrollados por la industria privada. 3) Crear un sistema de arquitectura abierta para futuros desarrollos, y que la comunidad científica pueda ampliar el desarrollo y el conocimiento de esta área de investigación, también contribuyendo la solución de la confidencialidad de sistemas desarrollados por el sector privado. 4) Crear una puesta experimental segura que no ponga en riesgo a los colaboradores, a terceros ni a objetos materiales.

1.4 Hipótesis

Es posible desarrollar un sistema de alerta al conductor ante una colisión frontal para frenado de emergencia en vehículo automotor basado en un clasificador CNN con imágenes digitales como entrada capaz de diferenciar entre 4 clases distintas; ruta libre, peatón, vehículo e infraestructura. Así como, realizar pruebas en campo para medir su desempeño y verificar el tiempo de anticipación oportuno para evitar la colisión que sea igual o mayor a 1.33 s.

1.5 Objetivos

En esta sección, se revisa el propósito de la presente investigación, y se delimitan los alcances de este proyecto.

1.5.1 Objetivo General

Desarrollar y validar mediante pruebas en campo un clasificador de imágenes basado en redes neuronales convolucionales (CNN) para identificar cuatro tipos de obstáculos en la vía: peatón, vehículo, infraestructura y camino libre, mediante una base de datos propia, capturada en un entorno controlado con una pista y un vehículo de prueba, para proveer alertas al conductor ante posibles colisiones y formar parte de un sistema ADAS del tipo EWA.

1.5.2 Objetivos específicos

1. Desarrollar la puesta experimental que consiste en; una pista de pruebas en el estacionamiento de la Facultad de Ingeniería campus San Juan del Río, objetivos a detectar como; dummies de adultos, objetos estáticos o infraestructura y vehículos falsos para obtener la base de datos y realizar las pruebas del sistema.

2. Adquisición de la base de datos usando un sensor de visión, como la cámara digital, en el campo de pruebas para la etapa de entrenamiento del algoritmo de DL y también una parte de la base de datos para la validación de este.
3. Desarrollar un algoritmo de procesamiento de imágenes y DL, en software de código abierto, para segmentación de áreas de interés, la obtención de características, clasificación y detección de los objetivos.
4. Integrar los algoritmos de DL y de procesamiento de imágenes, junto con el sistema de visión y la alerta visual, montándolo en el vehículo eléctrico, y poder realizar pruebas de validación en campo.
5. Realizar la validación del sistema EWB en la pista de pruebas, conduciendo el vehículo hacia la zona de detección de objetivos a velocidades definidas, para validar la detección tomando en cuenta los parámetros de tiempo de detección, tiempo estimado de colisión, tipo de objetivo, la velocidad de crucero, distancia recorrida y distancia de detección.

1.5.3 Planteamiento General

Para el desarrollo de este trabajo, se toma como base el diagrama de flujo mostrado en la **Figura 4**.

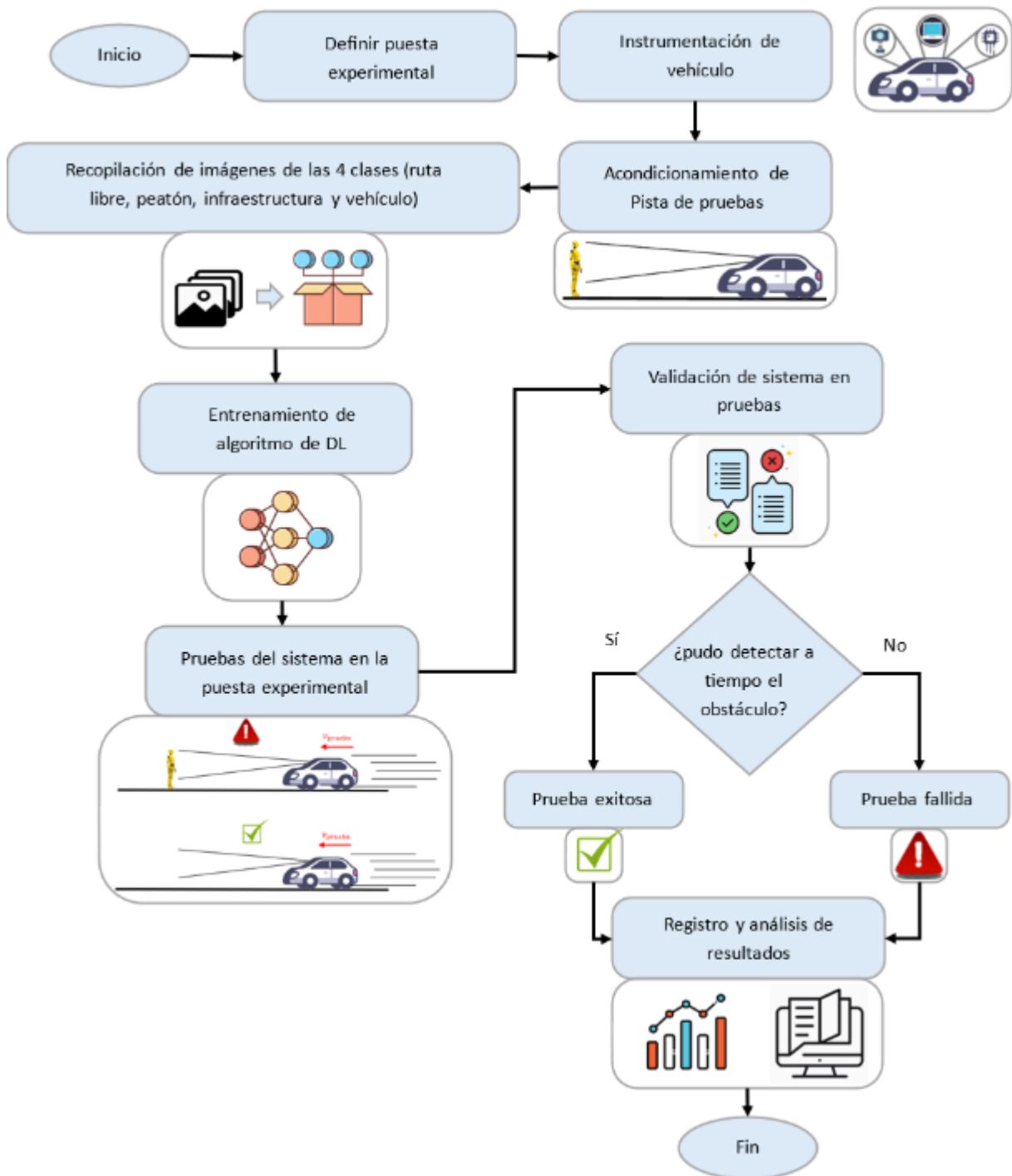


Figura 4. Diagrama del planteamiento general de la metodología a implementar.

En esta, se detallan los pasos requeridos para la identificación de posibles colisiones con los objetivos para generar una alerta de emergencia para el frenado. Este sistema se implementará en el vehículo eléctrico desarrollado por la Universidad Autónoma de Querétaro. La validación del de la metodología propuesta se realizará en un ambiente con condiciones controladas, con luz de día clara, puesto a que en la industria es un criterio para la captura de datos y las pruebas.

Se comenzará por acondicionar el vehículo con un sensor de visión y un ordenador, que se encargarán de adquirir la base de datos para entrenar el algoritmo de reconocimiento de objetos. Además, se revisa el estado del arte para identificar las áreas de oportunidad que son consideradas en este proyecto de Tesis.

Después, se adaptará un estacionamiento como una pista de pruebas, la cual se encontrará dentro de las instalaciones de la Universidad Autónoma de Querétaro, en el estacionamiento que se encuentra en la Facultad de Ingeniería, campus San Juan del Río. En esta, se situará el vehículo eléctrico con el sistema de visión; para realizar tareas de recopilación de datos para el post procesamiento y entrenamiento del algoritmo de DL.

Después, para obtener la base de datos, se realizan pruebas con 4 casos: enviando un *dummy* de adulto y de niño como prueba dinámica, también colocando cilindros de seguridad y un vehículo falso de manera estática. Se realizan al menos 10 pruebas por cada caso de estudio, con velocidades de 5, 10, 15 y 20 km/h, dando como resultado un total de 160 pruebas. Cabe mencionar que las pruebas se realizarán primero en un vehículo de pruebas para validar el sistema y después se integrará en el vehículo eléctrico EFACI, en horario que no ponga en riesgo a los peatones dentro de la Universidad Autónoma de Querétaro.

Una vez adquirida la base de datos, se procede a acondicionar el material recabado, con el algoritmo de procesamiento de imágenes desarrollado para extraer información que pueda ser de interés al momento de entrenar el algoritmo de DL.

Posteriormente, se procede al entrenamiento del algoritmo de DL desarrollado para la detección de los objetivos, con los datos preprocesados por el algoritmo de procesamiento de imágenes de la etapa anterior. Con una parte de los datos destinados para el entrenamiento, y otra para la validación.

La validación del sistema EWB se realizará repitiendo los casos de estudio que la obtención de la base de datos, integrando el sistema EWB, el sistema de visión y los algoritmos de DL desarrollados en el vehículo eléctrico, se tomará en cuenta el tiempo de procesamiento en el que le toma detectar el objetivo y el tiempo estimado de colisión; siendo que el sistema tiene que haber generado la alerta en un tiempo estimado de colisión superior a 1.5 s, además de considerar la distancia en el que frena el vehículo.

Finalmente, se procede a reportar los resultados obtenidos durante el desarrollo este proyecto de Tesis. Cabe mencionar que la metodología que se desarrollará buscará abrir una línea nueva de investigación, donde se podría realizar investigaciones de más ADAS como BSD, ACC, EBA, entre otros, para así, aportar a la problemática planteada.

CAPITULO II: FUNDAMENTACIÓN TEÓRICA

En este capítulo se realizó una recopilación de aspectos técnicos y teóricos que deben ser tomados en cuenta con el objetivo de generar el sistema de procesamiento de imágenes, y el algoritmo de DL, además de la implementación en el vehículo eléctrico EFACI y del acondicionamiento del estacionamiento para realizar las pruebas y la validación del sistema ADAS.

2.1 Hardware

Para poder interactuar con el entorno, se requieren elementos físicos los cuales se encargan de efectuar físicamente las instrucciones requeridas por el sistema ADAS, por ello, se mencionan los dispositivos de Hardware que se emplearán en este proyecto de Tesis.

2.1.1 Vehículo Eléctrico

El vehículo eléctrico (**Figura 5**) fue una iniciativa de estudiantes y alumnos de las carreras de ingeniería Electromecánica (IE) en conjunto con la Ingeniería Mecánica y Automotriz (IMA) de la Facultad de Ingeniería (UAQ, 2024) de la Universidad Autónoma de Querétaro, en la cual se adaptó un vehículo de combustión a un banco de baterías, haciendo así un vehículo eléctrico. Con este proyecto, se busca dar solución a la problemática de transporte, ambiental y tecnológica, implementando aplicaciones del área de electrónica, eléctrica, control, automatización y diseño mecánico.



Figura 5. Vehículo eléctrico de la facultad de ingeniería de la Universidad Autónoma de Querétaro (FI, 2024).

Si bien ya tiene integrado varios sistemas que ayudan a su funcionamiento, aún quedan varias áreas de oportunidad para desarrollarse, especialmente en el campo de vehículos autónomos. Este proyecto de tesis pretende ser el primer sistema ADAS implementado en el EFACI, aplicando técnicas de Aprendizaje de Maquina (“ML: Machine Learning”, por sus siglas en inglés) y reconocimiento de patrones, se lograría un gran avance en campo del desarrollo de sistemas autónomos.

2.1.2 Cámara Web Logitech© HD Pro C920s

La cámara Web Logitech© HD Pro C920s (**Figura 6**) captura imágenes con una claridad, nitidez y detalle excepcionales, resaltando colores vibrantes ya que cuenta con una resolución máxima de 1080p/30 fps a 720p/30 fps y un enfoque automático. Su enfoque automático y corrección de iluminación en alta definición se adaptan al entorno para mantener una calidad constante. Cuenta con dos micrófonos ubicados a los lados de la lente de cristal, los

cuales registran la voz de manera natural. Su campo de visión de 78 grados permite incluir a dos personas en el encuadre.



Figura 6. Web Logitech© HD Pro C920s

Para este proyecto, se utilizará esta tarjeta para enviar la señal de alerta visual del sistema EWB, para alertar al conductor de una posible colisión.

2.1.3 Dispositivos de Alerta

Como se mencionó en secciones anteriores, los vehículos modernos cuentan con sistemas ADAS con distintas funciones, estas pueden ser tan complejas como el evaluar el entorno para identificar riesgos teniendo de ejemplo vehículos que estén muy cerca del conductor, y así mantener o cambiar la dirección, conservar una distancia de seguridad, o evadir obstáculos repentinos. También pueden ser sistemas un poco menos robustos como una alerta visual (como un Led o pantalla), una alerta sonora (un pitido constante) o ambas. Un ejemplo de estos son los implementados en los vehículos BMW© serie 3 Berlina y Touring (**Figura 7**), el cual cuenta con un sistema ACC, y el sistema Driving Assistant que realiza la función de avisar la posible salida de un trayecto o de una posible colisión (BMW, 2024).



Figura 7: a) Control de crucero adaptativo y b) sistema Driving Assistant implementados en los vehículos BMW© serie 3 Berlina y Touring (BMW, 2024).

En este proyecto de tesis, se busca generar una alerta para el frenado de emergencia ante la presencia de un obstáculo, y también busca implementarse en un sistema EWB en el vehículo eléctrico.

2.1.4 Pista de pruebas

Las pistas de pruebas (**Figura 8**) son ambientes controlados en las cuales se instrumentan los vehículos de pruebas junto con los dummies, y con estos se ponen a prueba

tanto la estructura del vehículo, los materiales con los que está hecho, o algún sistema como puede ser de frenado, transmisión, suspensión, tecnologías de materiales, bolsas de aire y, hoy en día, reconocimiento y evasión de obstáculos, frenado automático, control y cambios de dirección, entre otros (Büyük et al., 2018).



Figura 8: Pista de pruebas para sistemas automotrices (Shutterstock, 2024).

Para este proyecto de Tesis se planea acondicionar una pista de pruebas para realizar adquisición de imágenes y pruebas del sistema EWB.

2.1.4 Maniquís de pruebas (*Dummies*)

Estos maniquís (Figura 9), también conocidos en la industria como “Crash test dummies” (o simplemente “dummies”), son un tipo de muñecos especiales que asemejan a la morfología humana, tanto como en forma adulta o infante con características de peso y articulaciones similares. Actualmente estos cuentan con diferentes sensores para medir distintas variables que

puedan influir en la seguridad del ser humano, como fuerza, velocidad y aceleración de impacto, fuerzas de torsión, velocidad, presiones de impacto, para poder recabar la mayor cantidad de información posible. Estos son utilizados ampliamente en la industria del transporte para desarrollar vehículos y aeronaves, para realizar pruebas de choque y evaluar los posibles daños que podría sufrir un ser humano al momento de una colisión en el caso de que es piloto, copiloto o pasajeros, para evaluar el desempeño de los sistemas de seguridad, o el nivel de seguridad del chasis de un vehículo (Jaśkiewicz et al., 2021).



Figura 9. *Dummies* en prueba de choque (Jaśkiewicz et al., 2021).

Para este proyecto de tesis, se plantea el uso de un dummie de forma adulta para ser parte de los objetivos a identificar por el sistema EWB.

2.2 Software

Para poder procesar la información, e implementar los algoritmos de identificación de patrones, ML, DL, o procesamiento de imágenes, además de generar las instrucciones requeridas por el sistema ADAS, se mencionan los dispositivos de Hardware que se emplean en este proyecto de Tesis.

2.2.1 Matlab[©]

Es un entorno de programación y un lenguaje de programación de alto nivel basado en C/C++, lo que ofrece un lenguaje de programación fácil de aprender y expresivo, lo que permite a los usuarios realizar diseño e implementación de sistemas de control, entrenar modelos de machine learning, procesamiento de señales, implementación de algoritmos de Deep learning, procesamiento de imágenes y visión artificial, creación de aplicaciones, entre otros gracias a sus extensa cantidad de funciones herramientas. Además, con sus herramientas para crear gráficos de alta calidad se puede organizar y visualizar de mejor manera información, visualizar datos, obtener información, e identificar patrones y tendencias. También se puede generar y ejecutar código en C, C++, CUDA, Verilog, VHDL y texto estructurado, e implementarlo en hardware, además de poder interactuar con dispositivos de adquisición de datos y controladores de instrumentos (MATLAB, 2024).

2.2.2 Lenguaje Python[©]

Python[©] (**Figura 10**) es un lenguaje de programación de alto nivel fácil de entender para programadores principiantes o experimentados en otros lenguajes por su sintaxis limpia y legible, con una comunidad activa y actualizada para el desarrollo proyectos junto con documentación de este lenguaje. Además de ser código abierto al ser desarrollado bajo una licencia aprbada por la Open Source Iniciative (OSI, por sus siglas en inglés), lo cual permite su uso y distribución para usos comerciales, como el desarrollo web y de internet, acceso y análisis de bases de datos, desarrollo de interfaces gráficas, hasta ML y DL.

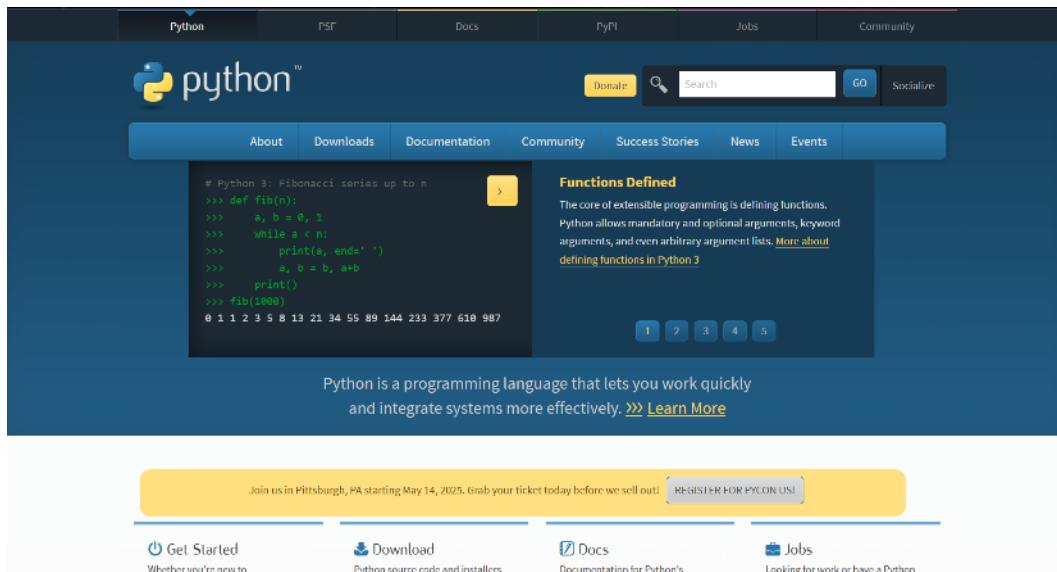


Figura 10: interfaz de sitio oficial de Python©

Por las características vistas anteriormente, se opta por utilizar este lenguaje para fines de código abierto, validación de resultados en entrenamiento de algoritmos de DL e implementaciones físicas del sistema EWB.

2.3 Técnicas de procesamiento de datos

Para poder delegar la tarea de identificación de patrones, se utilizan técnicas de procesamientos de datos obtenidos en imágenes, señales, o datos. A continuación, se mencionan las utilizadas en este Proyecto de Tesis.

2.3.1 Sistemas de procesamiento de imágenes

Son un conjunto de elementos como cámaras digitales (**Figura 11**), iluminación y unidades de procesamiento, que analizan datos otorgados por fotografías y videos, para adquirir información y extraer características como la identificación de personas, objetos, patrones, colores, formas, texturas, entre otros aspectos, con técnicas de procesamiento de imágenes.

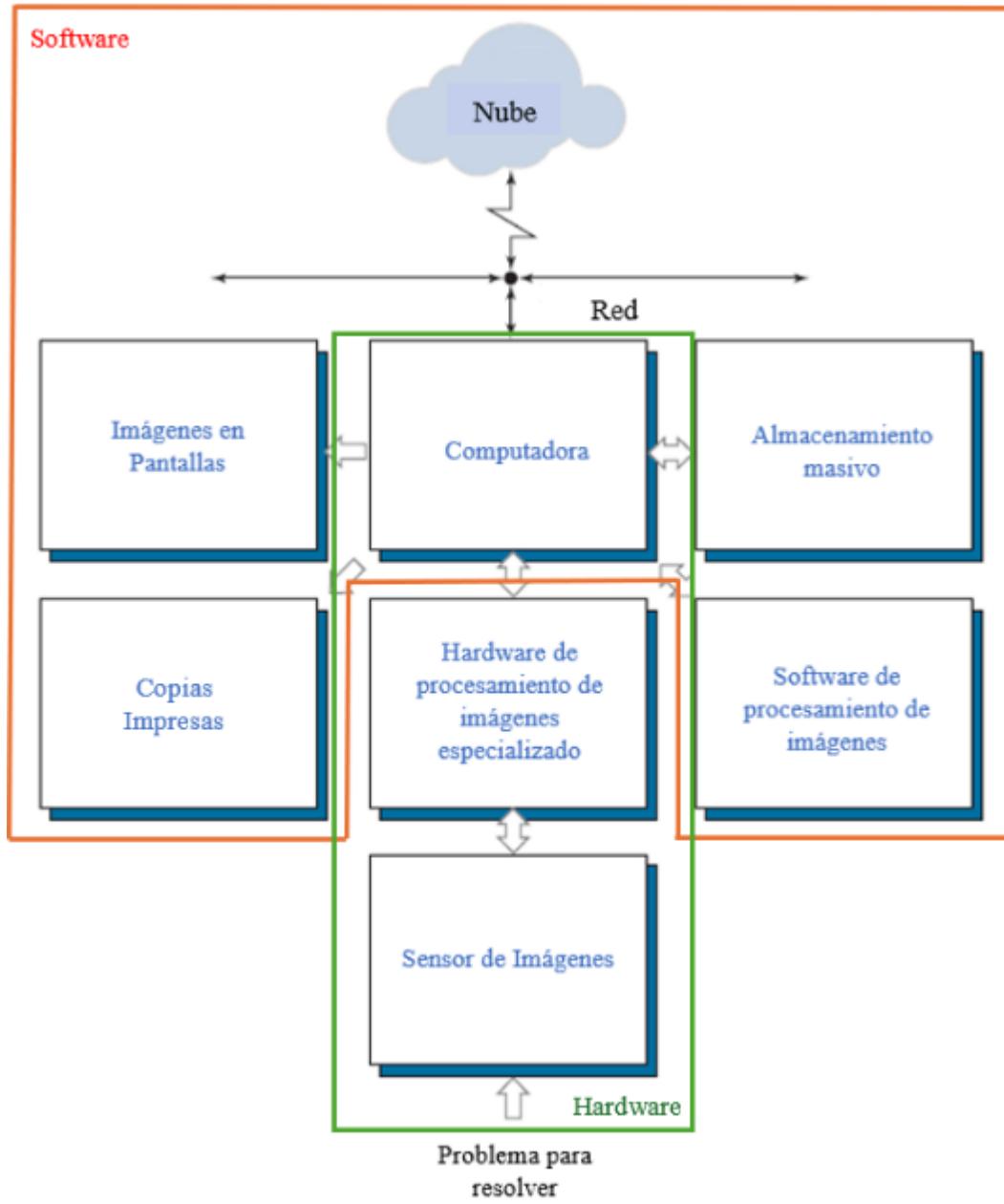


Figura 11. Diagrama a bloques de un sistema de visión (Gonzalez & Woods, 2018).

En la **Figura 11** se muestra el diagrama a bloques que conforman un sistema de procesamiento de imágenes el cual consta de una serie de elementos. El primero es el sensor de imágenes, que es sensible a la energía radiada por el objeto de interés, y el segundo es el digitalizador, que convierte la señal generada por el sensor en una imagen digital, que, en conjunto con el hardware especializado, pueden realizarlo a una alta velocidad. La computadora es un sistema de procesamiento de manera general, puesto que esta recibe la información en

forma de imágenes y video, y por medio de softwares de procesamiento de imágenes dedicados a tareas específicas y analizar la información obtenida para la toma de decisiones. El almacenamiento masivo es requerido para almacenar imágenes puesto la gran cantidad de información que puede contenerse en una sola imagen. Para visualizar la información se requiere desplegarlas en pantallas, regularmente utilizadas a color. Los dispositivos de copia impresa son los que pueden ser impresoras láser, cámaras de película, dispositivos sensibles al calor, unidades de inyección de tinta y unidades digitales, cada una utilizada para distintos propósitos. Finalmente, la red y la nube son indispensables para el procesamiento de imágenes y acceder a la información no importa la distancia del objeto de análisis (Gonzalez & Woods, 2018).

2.3.2 Deep Learning (DL)

El aprendizaje automático es clave en muchas aplicaciones modernas, desde búsquedas web hasta reconocimiento de imágenes y voz. Sin embargo, los métodos tradicionales requerían diseñar manualmente algoritmos para la extracción de características para transformar los datos en un formato adecuado para su análisis. El DL resuelve este problema al permitir que los sistemas aprendan representaciones directamente de los datos crudos, mediante múltiples capas de transformación no lineales. Estas capas aprenden características cada vez más abstractas, lo que permite reconocer patrones complejos sin intervención humana. Gracias a su capacidad para descubrir estructuras en datos de alta dimensión, el DL ha logrado avances significativos en diversas áreas, como reconocimiento de imágenes y voz, predicción de actividad molecular, análisis de datos científicos y comprensión del lenguaje natural. Su éxito se debe a su capacidad de escalar con más datos y potencia computacional, y se espera que continúe mejorando con nuevos algoritmos y arquitecturas (Lecun et al., 2015).

2.3.3 Aprendizaje supervisado

El aprendizaje supervisado es la forma en que funcionan la mayoría de los algoritmos de ML y DL para la tarea de clasificación. Para modelos entrenados por optimización, se toma una cantidad de datos etiquetados y clasificados para adquirir características relevantes y así obtener un modelo de clasificación para las clases, una vez que se adquieren los rasgos representativos estas, se utiliza el modelo con un conjunto de datos de prueba y se obtiene la

precisión del modelo calculando el error como la diferencia entre el valor real y el valor arrojado por la predicción, comúnmente se usa el error cuadrático medio (MSE, por sus siglas en inglés) (1) o la entropía cruzada (2). En el DL, se realiza este proceso entrenando un modelo con un gran conjunto de datos etiquetados, ajustando sus parámetros (pesos) para minimizar el error entre sus predicciones y las etiquetas correctas mediante el descenso de gradiente estocástico (SGD, por sus siglas en inglés) aplicado a la función error z mostrado en (3) (Lecun et al., 2015).

$$z_1 = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (\text{Ec. 1})$$

$$z_2 = -\frac{1}{N} \sum_{i=1}^N y_i \log(\hat{y}_i) \quad (\text{Ec. 2})$$

$$(\omega^i)^{t+1} = (\omega^i)^t - \eta \frac{\partial z}{\partial x(\omega^i)^t} \quad (\text{Ec. 3})$$

Donde z_1 y z_2 son funciones de pérdida, z es la función pérdida a minimizar, y_i es el valor real de la muestra i , \hat{y}_i es el valor predicho por el modelo, N es el número de muestras, η es el coeficiente de aprendizaje, $(\omega^i)^t$ es el valor actual del parámetro (peso) en la época t , $(\omega^i)^{t+1}$ es el valor actualizado por el SGD.

2.3.3 Redes Neuronales Artificiales

A diferencia de los clasificadores lineales tradicionales utilizados en ML que solo pueden separar datos con regiones simples, dificultando tareas como el reconocimiento de imágenes y voz, el DL utiliza redes neuronales profundas (DNN, por sus siglas en inglés) que aprenden automáticamente representaciones jerárquicas de los datos. Estas están formadas por nodos o unidades de procesamiento (también denominados perceptrones), dichas unidades están organizadas en capas de entrada, ocultas y de salida (Shrestha & Mahmood, 2019), en la **Figura 12a** se muestra una representación gráfica y matemática de un perceptrón, y en la **Figura 12b** se muestra una imagen demostrativa de una arquitectura para DNN.

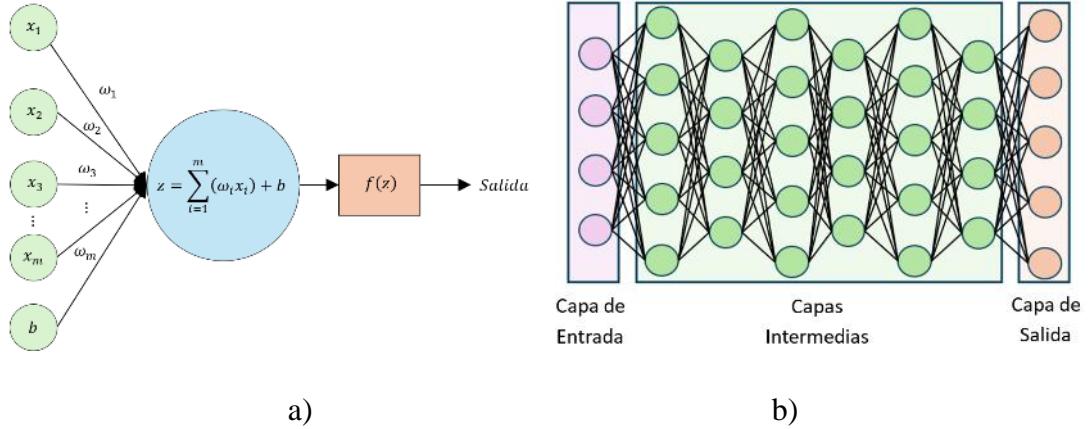


Figura 12. a) Perceptrón simple y b) arquitectura de una DNN.

Cada nodo (o perceptrón) en cada capa recibe un estímulo (o entrada), que es multiplicada por su respectivo peso que pondera la relevancia de dicha entrada, después es sumada en el nodo en cuestión para someterse a una transformación basada en una función de activación, como la sigmoide (4), tangente hiperbólica (5), unidad lineal rectificada (ReLU, por sus siglas en inglés) (6) y *softmax* (7).

$$f(z) = \frac{1}{1 + e^{-z}} \quad (\text{Ec. 4})$$

$$f(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \quad (\text{Ec. 5})$$

$$f(z_i) = \max(0, z) \quad (\text{Ec. 6})$$

$$f(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}} \quad (\text{Ec. 7})$$

Donde z es la suma ponderada de los estímulos que entraron a la neurona (Figura 8a).

La (Ec. 4), denominada sigmoidal, tiene la característica de que abarca valores de $[0,1]$ y es diferenciable en todo su dominio, haciéndola útil para el uso de probabilidades, aunque se desvanece el gradiente en valores grandes de z . La tangente hiperbólica (Ec. 5) obtiene valores de $[-1,1]$ y acelera la convergencia, pero tiene la misma desventaja de desvanecimiento del gradiente. La ReLU (Ec. 6) obtiene valores $[0, +\infty]$ es computacionalmente eficiente y reduce la el problema del desvanecimiento, su desventaja es que pueden surgir neuronas muertas y el

modelo puede dejar de aprender al tener valor en el peso que pueda hacer la entrada siempre cero. En cuando a la Softmax (Ec. 7), obtiene valores de $[0,1]$, y se utiliza en la salida de las redes neuronales para una clasificación multiclas. Transforma un vector de valores reales (las salidas de la última capa de la red) en un vector de probabilidades normalizadas.

2.3.4 Redes Neuronales de Convolucionales

Las Redes Neuronales Convolucionales (CNN, por sus siglas en inglés) están basadas en la corteza visual humana y empleada en aplicaciones de visión por computadora, reconocimiento de video y otras aplicaciones. Pueden identificar, reconocer y clasificar objetos, así como segmentar elementos dentro de imágenes. Para entender cómo funcionan las arquitecturas CNN es necesario analizar sus componentes y sus aplicaciones. En la Figura 9 se muestran los componentes de una CNN (Taye, 2023). En la **Figura 13** se muestra una imagen representativa de las etapas de una CNN.

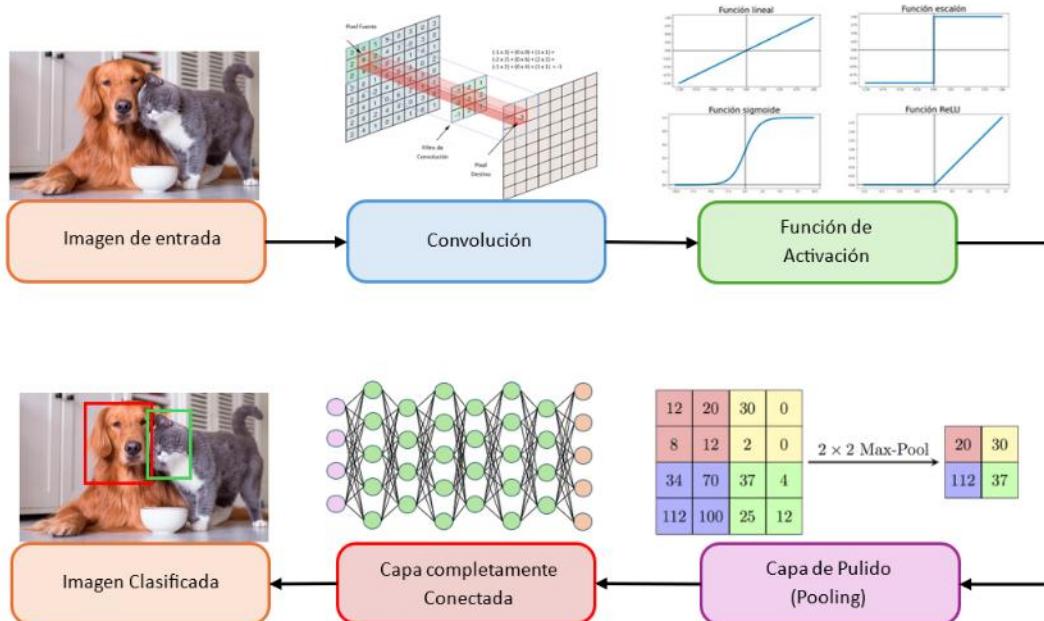


Figura 13. Componentes de una CNN

Como se puede observar, una CNN está compuesta típicamente de 4 tipos de capas: Convolucional, Pooling, Función de activación y Completamente conectada.

2.3.4.1 Imagen de entrada

Es representada matemáticamente como un arreglo rectangular en el cual tiene dimensión largo por ancho, cada casilla en este arreglo es denominado pixel y tiene un valor de 0-255, y pueden ser 3 arreglos en una sola imagen si es de 3 canales (rojo, verde y azul, en la mayoría de los casos). El conjunto de los valores que corresponden a cada casilla en el arreglo rectangular da como resultado una imagen que puede ser interpretada por la CNN y clasificar objetos con la información almacenada en el arreglo rectangular, o imagen.

2.3.4.2 Capa de convolución

Funamental en la arquitectura de una CNN. Está compuesta de un conjunto de filtros (o kernels). Un kernel es un arreglo rectangular de mucho menor dimensión, con valores enteros. A cada número se le asigna un peso (o valor) dentro del kernel. Los pesos iniciales en una CNN son un conjunto de enteros elegidos aleatoriamente, y con el proceso de entrenamiento, el kernel puede identificar características específicas desde bajo nivel (como bordes y contornos, esquinas y cruces, texturas finas, gradientes de color), mediano nivel (“mofits”, regiones de interés o texturas compuestas), y alto nivel (relaciones espaciales, objetos completos y sus partes). En la **Figura 14** se muestra la operación de Kernel y su interpretación más intuitiva.

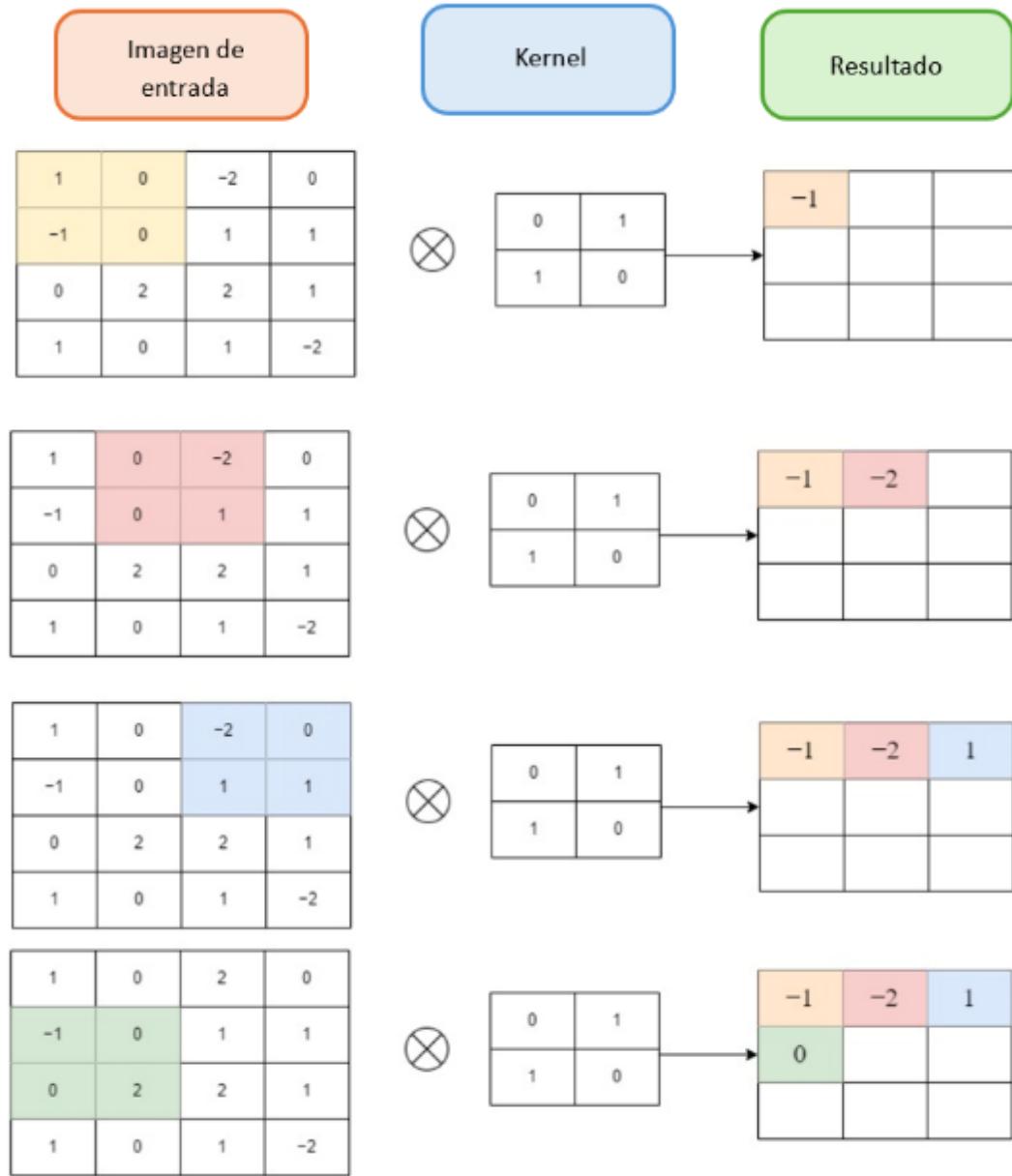


Figura 14. Representación visual de cálculos primarios de un Kernel (Taye, 2023).

La convolución toma una ventana de pixeles de la imagen de tamaño igual al del Kernel que se fija (en el caso de la **Figura 14**, es una ventana de 2x2). Utiliza esta ventana para multiplicar el valor correspondiente a su ubicación en el kernel y sumar los valores para obtener un valor final, repitiendo estas operaciones a lo largo de la imagen se obtendrá una imagen con las características obtenidas por el kernel. En este caso el kernel tiene un salto (Stride), o avance, de una casilla para realizar otra operación, esto se puede cambiar para obtener un resultado

diferente en la salida, pero podría disminuir la dimensión de la característica que se está analizando, además de perder información que podría ser valiosa. Para ello, se puede agregar un relleno (Padding) para conservar la dimensión de las características. En la **Figura 15** se muestran ejemplos de filtros similares a los obtenidos por una CNN con Kernels de dimensión 3x3.

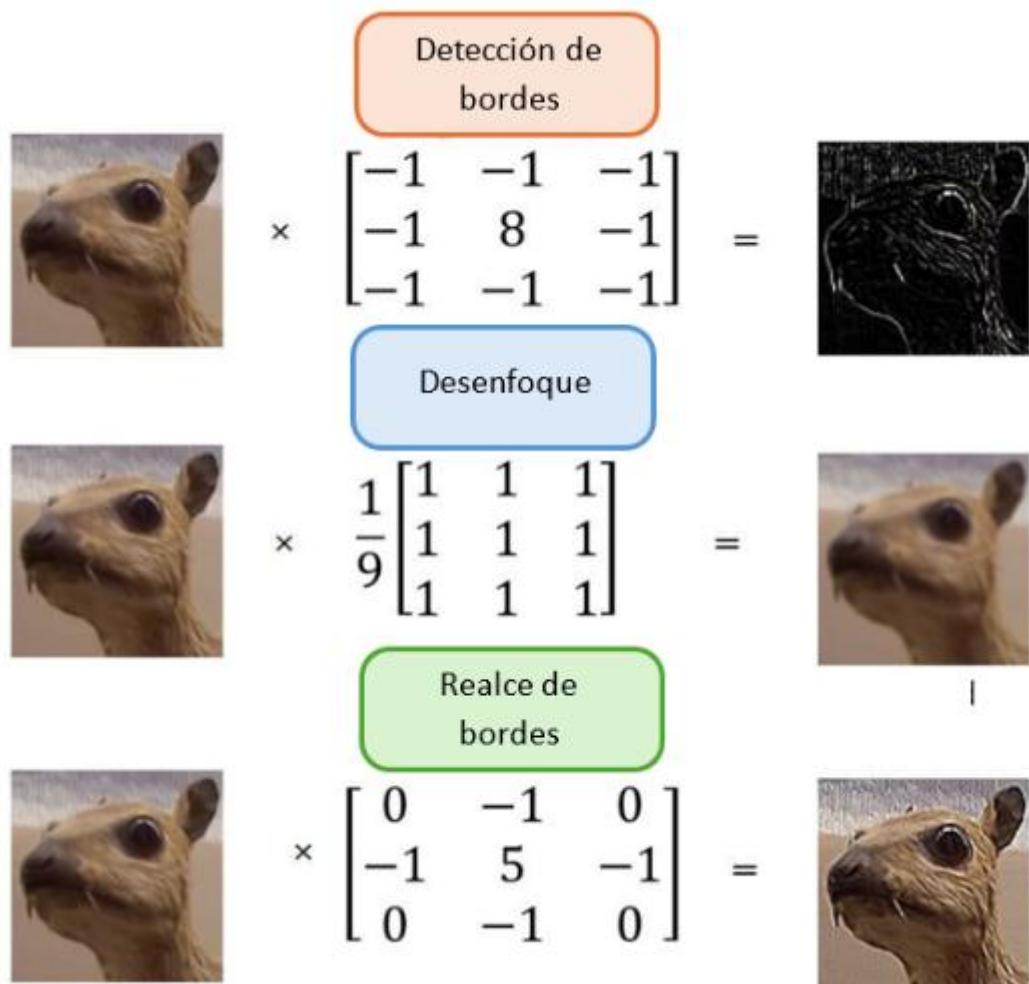


Figura 15: Efectos de diferentes Kernels en una imagen (Taye, 2023).

Cada filtro puede representar una característica específica. Si un filtro se desplaza sobre una imagen y no encuentra coincidencias, no se activa. La CNN utiliza este mecanismo para identificar los filtros más efectivos para describir los objetos.

2.3.4.3 Funciones de activación (No linealidad)

Seguida de la convolución, una vez realizada la operación de convolución y se calcula el mapa de características, la función de activación actúa pixel a pixel para acotar o evitar la sobresaturación de los valores de salida. Estas funciones pueden ser la sigmoidal (Ec. 4), tangente hiperbólica (Ec. 5) o ReLU (Ec. 6).

2.3.4.4 Pulido (Pooling)

A veces se puede La capa de Pooling se utiliza para reducir la dimensionalidad de los mapas de características obtenidos por los Kernels, estos de igual manera toman un tamaño de ventana y toman la muestra más significante que decida tomar el usuario, estas pueden ser máximo, mínimo o promedio. En la **Figura 16** se muestra un ejemplo de dicha operación.

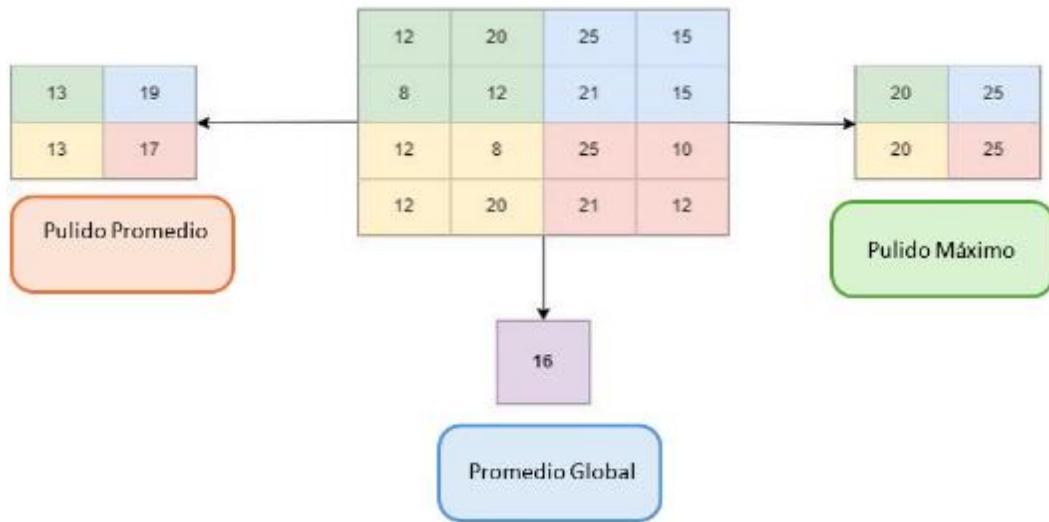


Figura 16: Ejemplos de operaciones de pulido.

Así, se logran discriminar valores que no resulten ser relevantes en los mapas de características, disminuyendo así la carga computacional antes de que pasen a la clasificación en la capa completamente conectada.

2.3.4.5 Capa completamente conectada (FCL)

Similares a las ya mencionadas en secciones anteriores, se encargan de analizar las características más importantes que han conformado los mapas de características resultantes. Se emplea una Red Neuronal Artificial para la clasificación de las clases en cuestión, teniendo como capa de entrada un vector resultante de “aplanar” los mapas de características, seguido de capas intermedias para realizar tareas de clasificación y como capa de salida la clase correspondiente a la que pertenece ese mapa de características.

CAPÍTULO III: DESARROLLO DE SISTEMA ADAS PARA ALERTA DE FRENADO EWA

Tomando como referencia la **Figura 3** para desglosar la metodología empleada para este proyecto de tesis. Se comenzará por acondicionar el sistema de visión al vehículo hospedador para la adquisición de imágenes de los objetivos (peatón, infraestructura, vehículo, camino libre). Posteriormente se acondicionará la pista de pruebas para y capturar las imágenes con el sistema de visión desde la perspectiva del vehículo hospedador para entrenar la CNN que se encargará de identificar el tipo de objetivo, modificando el número de kernels, capas de convolución, numero de neuronas, épocas para el entrenamiento. Luego, se procederá a realizar la prueba de sistema EWA que conjunta el sistema de adquisición de muestras y la CNN entrenada para su clasificación y alerta, estas pruebas consistirán en una puesta experimental basada en reglamentos viales, velocidades permitidas en espacios urbanos, tiempos estimados de frenado y del estado del arte revisado. Finalmente, se registran los resultados de las pruebas para verificar la vialidad de este sistema.

3.1 Captura de base de datos

En este apartado se desglosa la metodología empleada para adquirir una base de datos que requieran imágenes en escenarios en entornos de vialidades desde la perspectiva de un vehículo. La base de datos consta de 4 clases: Infraestructura, Peatón, Vehículo y Camino Libre, cada clase contiene 2000 imágenes redimensionadas y en escala de grises para disminuir la carga computacional del sistema final. La obtención de esta base de datos y una descripción más detallada se muestra en las siguientes secciones.

3.1.1 Configuración de Vehículo Hospedador

Se utiliza la cámara Logitech[©] C920 instalada en el tablero del vehículo hospedador para obtener la perspectiva del vehículo en su navegación, con la finalidad de obtener imágenes

lo más cercano a los que se pueden presentar en la vida real. También, se agrega un ordenador que se encargará de ejecutar el código de captura de imágenes durante la conducción hacia los objetivos, este vehículo se muestra en la **Figura 17**.

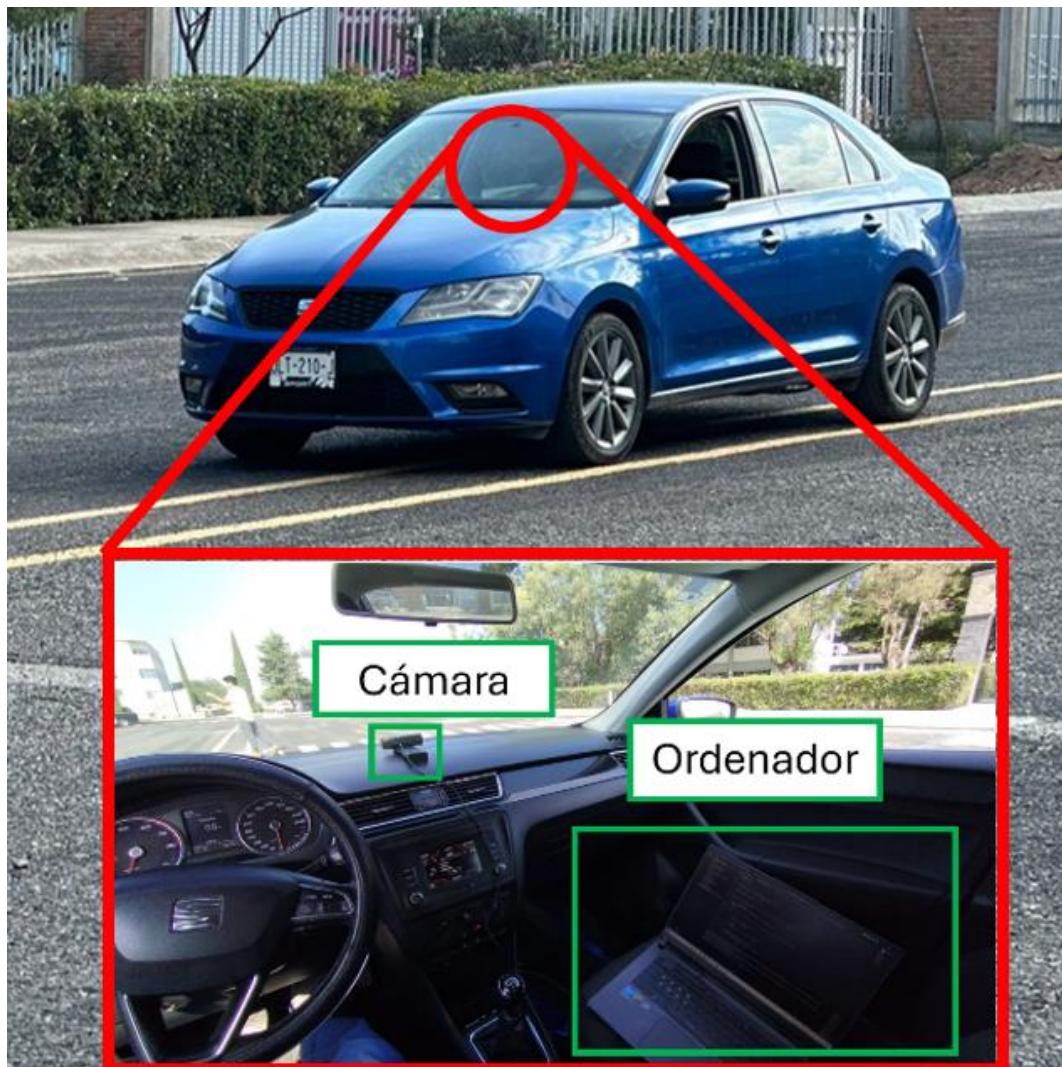


Figura 17: Acondicionamiento de vehículo hospedador con sistema de adquisición de imágenes.

Con el vehículo acondicionado, se obtienen imágenes que se asemejan a escenarios reales vistos desde la perspectiva del conductor, a velocidades establecidas con base en el reglamento de tránsito.

3.1.2 Adecuación de pista de pruebas para captura

Para tener un ambiente controlado, y no poner en riesgo a personas o infraestructura, se acondiciona el estacionamiento ubicado en la Universidad Autónoma de Querétaro, plantel San Juan del Río. Esta se divide en zonas como se muestra en la **Figura 18**.

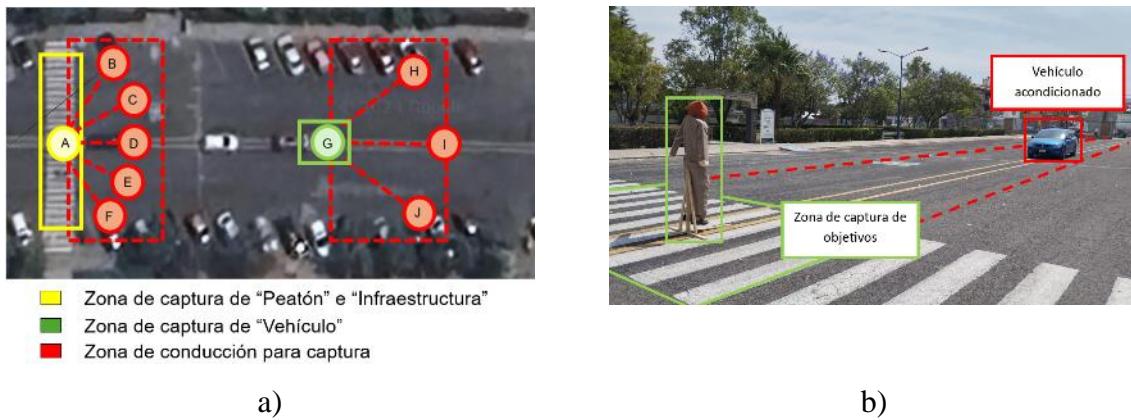


Figura 18: a) Acondicionamiento de la pista de pruebas con las regiones de captura de imágenes en el estacionamiento de la Facultad de Ingeniería y b) exemplificación de escenario de peatón caminando para la captura de imágenes

El escenario de las imágenes correspondientes a la clase de Peatón e Infraestructura se toma en cuenta a una persona desplazándose por la zona de captura de peatón y un dummie adulto estático, variando el ángulo de captura mostrados con los puntos B a F y conduciendo el vehículo hospedador en hasta desde estos puntos acercándose al punto A con velocidades variables de 0 a 10 km/h, tomando en cuenta que la velocidad relativa es la misma del vehículo hospedador dado que ningún objetivo se mueve en dirección al vehículo hospedador. En cuanto la detección de Vehículo, se colocan estos objetivos en el punto G y se conducen el hospedador desde los puntos H, I y J, hasta el punto G. Cada imagen en formato “crudo” tiene un tamaño de 640 × 480 pixeles.

3.1.3 Captura de clase “peatón”

Para capturar la clase de peatón, se considera el área correspondiente en amarillo de la Figura 18, donde se desplazan una persona individual y un grupo de 5, 3 y 2 personas, en esta área, además de un dummie de adulto estático puesto en el punto A para su captura. Se conduce el hospedador de los puntos B a F al punto A capturando imágenes de los peatones en movimiento y del dummie estático, en un horario diurno con luz natural clara y despejada. En la **Figura 19** se muestra un ejemplo de cómo se realiza este proceso

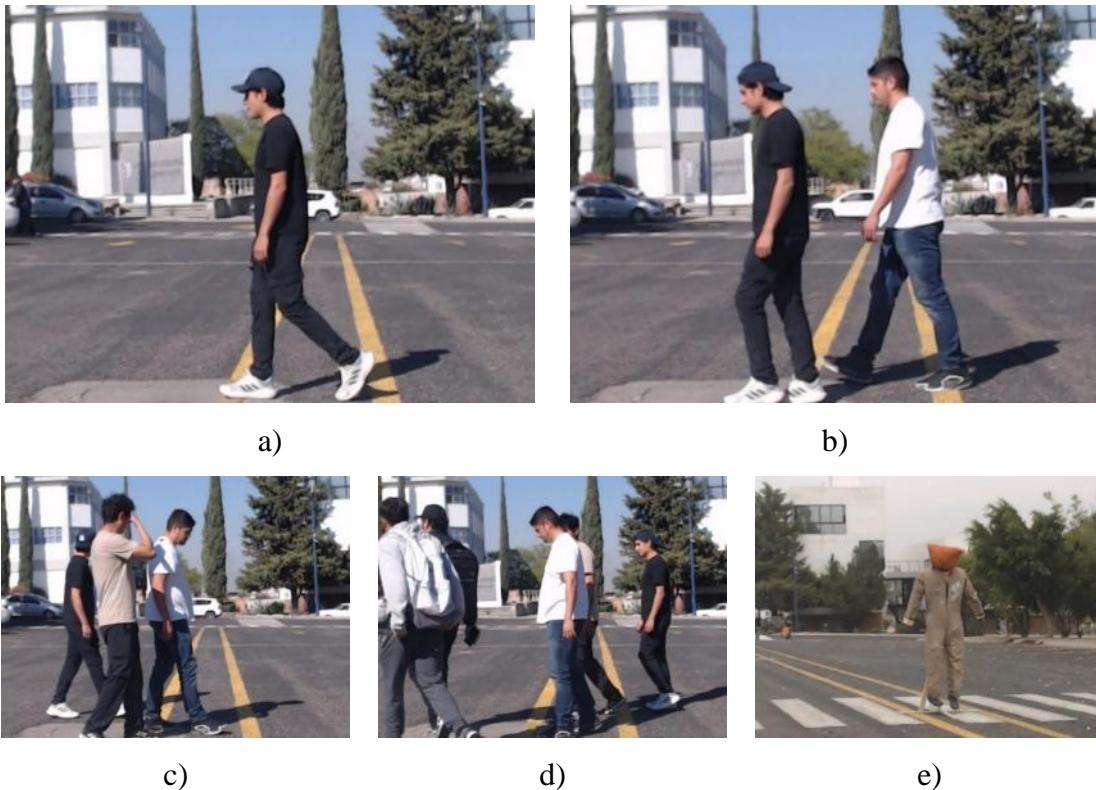


Figura 19. Captura de imágenes de clase “peatón” a) una persona desplazándose, grupo de b) 2 personas, c) 3 personas, d) 5 personas desplazándose y e) dummie estático.

Así, se contemplan distintos escenarios que se pudieran presentar en la identificación de la presencia de un peatón mientras se conduce en una vialidad urbana.

3.1.4 Captura de clase “Infraestructura”

La clase infraestructura consta de un bote de tránsito con reflejante que se coloca en el punto A de la pista de pruebas mostrada en la **Figura 18**. Para la captura de imágenes se conduce el hospedador desde los puntos B a F hacia el punto A. En la **Figura 20** se muestra un ejemplo de la captura de infraestructura con un ángulo de captura.

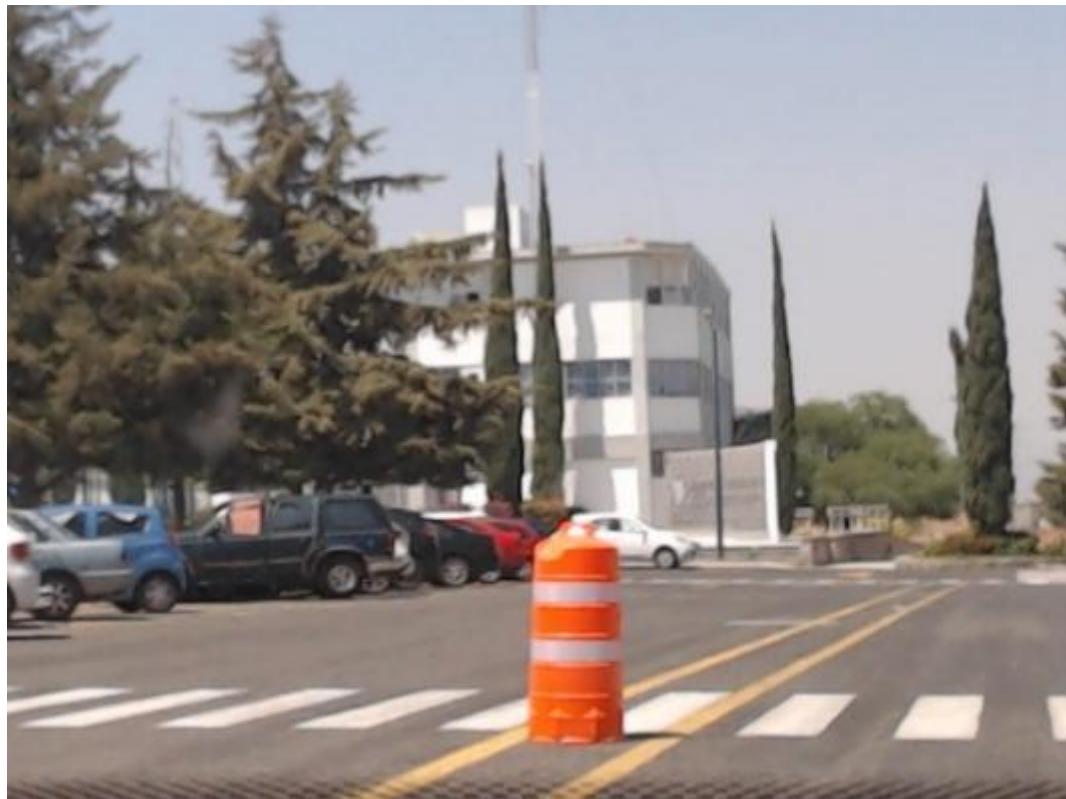


Figura 20: Clase de infraestructura capturada variando ángulo de captura

De esta manera, se toman en cuenta distintos escenarios para la captura del objetivo que se podrían presentar en un escenario real.

3.1.5 Captura de clase “Vehículo”

La clase de Vehículo consta de dos vehículos colocados en la zona G de la Figura 18 para ser capturados por el hospedador, desplazándose desde el punto H a j hasta el punto G a una velocidad constante. En la **Figura 21** se muestra un ejemplo de imágenes capturadas por el Hospedador.



a)

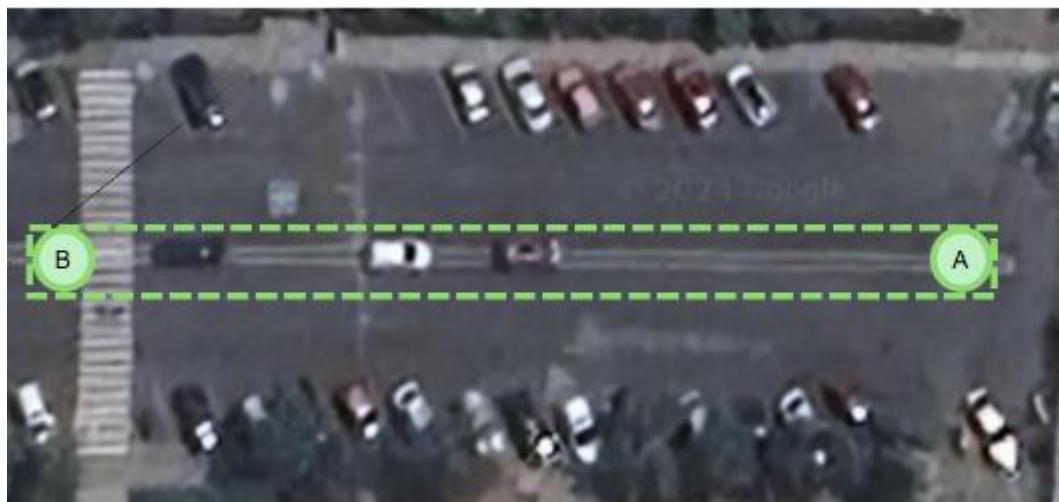
b)

Figura 21: Vehículos capturados para la clase “Vehículo”.

Así, se contempla un último objetivo para las pruebas para el sistema EWA.

3.1.6 Captura de “Ruta Libre”

Para identificar cuando el vehículo pueda conducir de manera segura sin que el sistema EWA se active, se realiza una adquisición de imágenes que no contendrán ninguno de los objetivos presentes, para esto, se utiliza una pequeña modificación a la forma de conducir el vehículo en la pista de pruebas. En la **Figura 22** se muestra el escenario para la captura de la clase de “Camino Libre”.



■ Zona de adquisición de “Ruta Libre”

Figura 22. Zona de adquisición de imágenes para clase ruta libre

Para esta clase, se contempla en movimiento al vehículo desplazándose a velocidad constante del punto A al punto B, adquiriendo imágenes con un tiempo de muestreo de 100 ms entre muestra. En posteriores secciones se mostrará el proceso del algoritmo de captura para las imágenes de la base de datos.

3.1.7 Algoritmo de captura automatizada de imágenes

Para la captura de las imágenes de los objetivos que formarán parte de la base de datos de manera automática se desarrolla un algoritmo el cual tiene como argumentos: el número de imágenes a obtener y el intervalo de tiempo en el que ocurre cada captura, este proceso se muestra en la **Figura 23**.

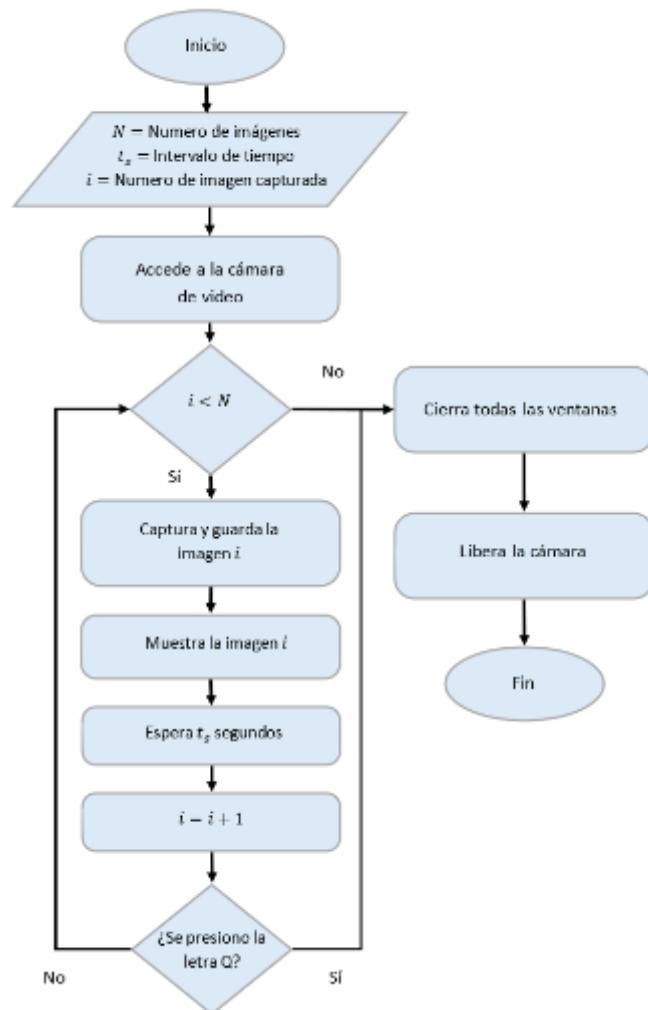


Figura 23: Diagrama de flujo para captura de imágenes de manera automática.

La captura de las imágenes comienza definiendo el número de imágenes que se requieren y el tiempo entre captura. Se accede a la cámara de video colocada en el tablero del vehículo para comenzar a tomar captura de imágenes, comienza con el indicador $i=0$ que representa la imagen actual que se está tomando, y si el indicador i es el menor que el número de imágenes toma una captura y la almacena como “ $imagen_i$ ”, e incrementa el índice hasta conseguir la N imágenes. Durante este proceso, existe una bandera de control q que si se presiona interrumpe el proceso y libera la cámara y termina la tarea. Este código se configura para adquirir las 2,000 imágenes desde el punto de vista del vehículo, con un intervalo entre captura de 100 ms. En la **Figura 24** se muestra una secuencia de 1.2 segundos donde se obtuvieron 12 imágenes con dicho tiempo de captura.

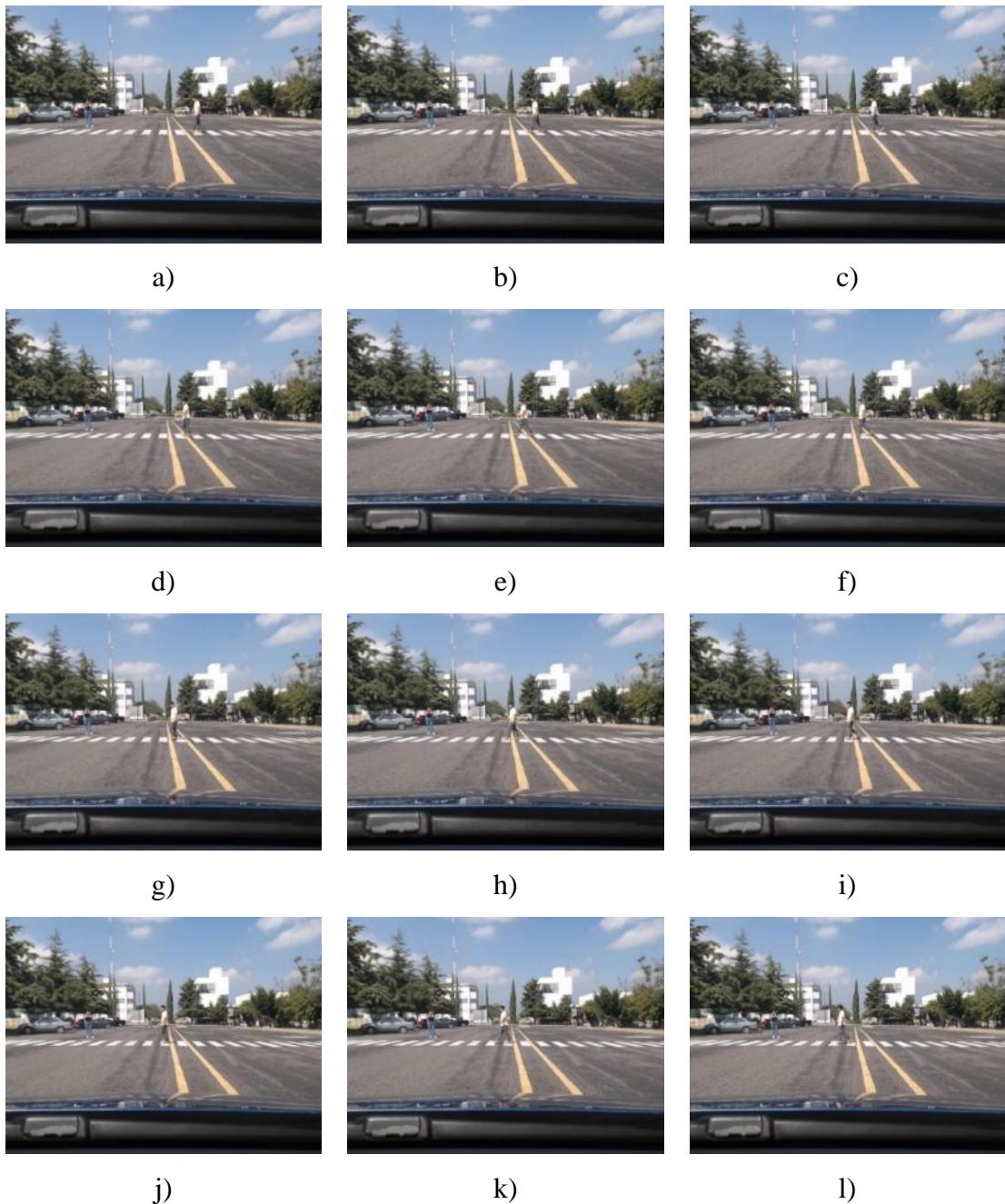


Figura 24: Secuencia de imágenes capturadas desde el punto de vista del vehículo hospedador.

De esta forma, se consigue obtener un gran número de imágenes que se requieren para el entrenamiento de algoritmos de clasificación, adecuándolos al contexto específico de su

aplicación y de su entorno. Y en la **Figura 25** se muestran las imágenes capturadas de todas las clases.

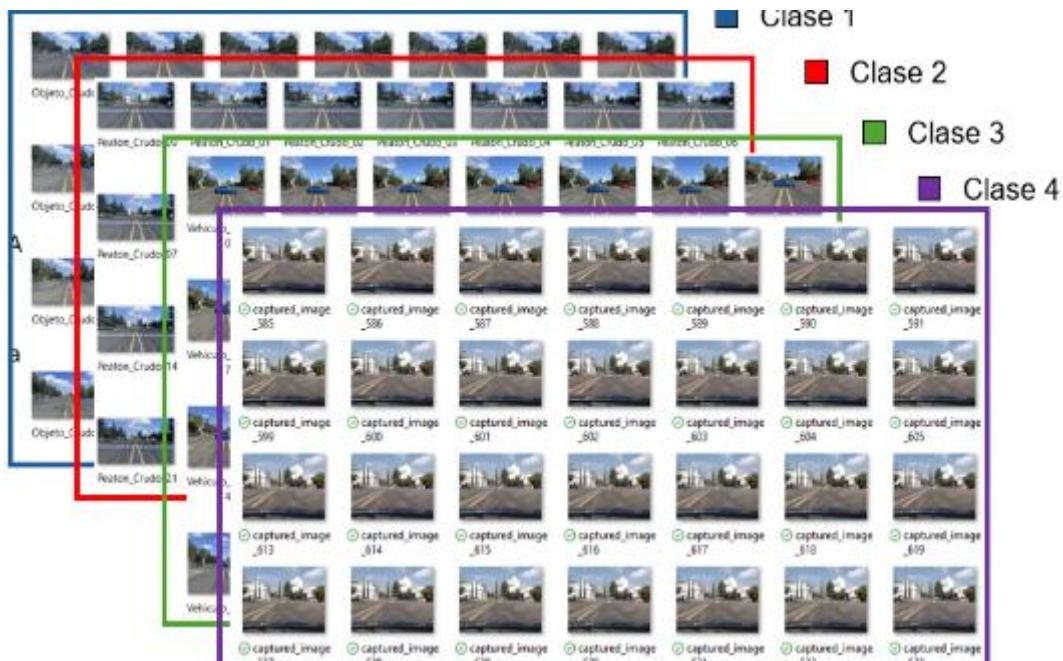


Figura 25: Imágenes capturadas en la etapa de adquisición de datos.

Estas imágenes se utilizan como clases para el entrenamiento de un clasificador de DL, modificando el tamaño y aplicando una escala de grises a estas para su clasificación.

3.2 Topología de CNN para clasificación

Para validar si el proceso de umbralización es funcional para la etapa de clasificación, es entrenando una CNN, la cual tiene como entradas las 4 clases analizadas en este trabajo de Tesis. En la **Figura 26** se muestra el proceso a seguir para su validación.

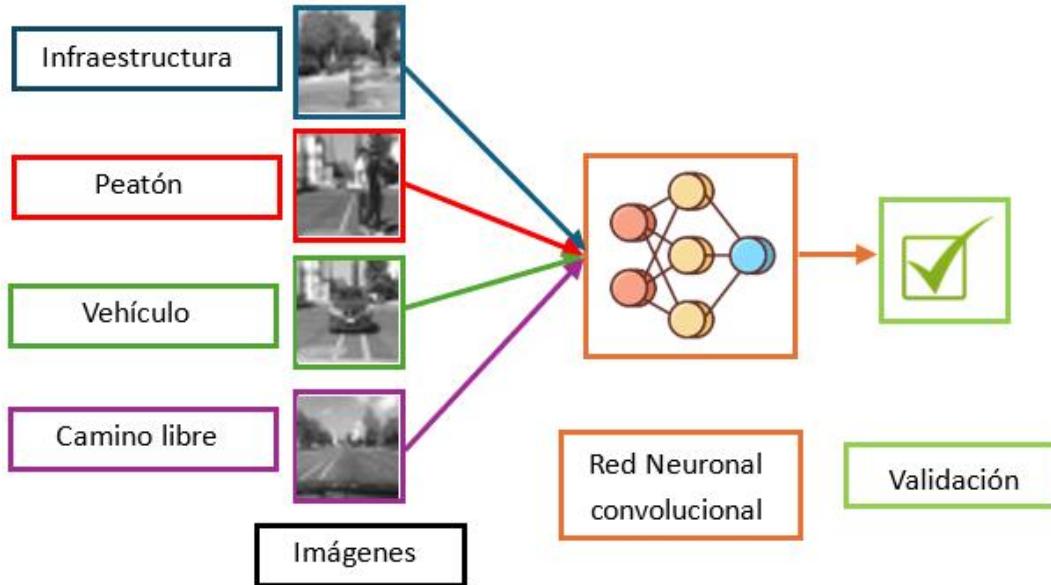


Figura 26: Validación de acondicionamiento de imágenes por medio de una red Neuronal Convolucional.

Para esta CNN, se utiliza una red con una etapa de convolución que tiene como entrada una imagen de 28×28 pixeles de un solo canal con valores de 0 a 255 (Escala de grises), seguido de una capa de convolución con 15 filtros de tamaño 3×3 , con función de activación ‘ReLU’ y un padding de ‘same’, después entra a una capa de pulido máximo con un tamaño de ventana de 3×3 y un salto (o “Stride”) de 2. Después transforma (aplana) el modelo para convertir el resultado de las capas de convolución 2D a un vector 1D de tamaño 1×196 . Posteriormente, entra a una etapa de neuronas completamente conectadas con 128 neuronas y una función de activación ReLU. Luego pasa por una tasa de apagado (“Drop Out”) del 50% que inhabilita aleatoriamente la mitad de las neuronas para reducir el sobreajuste y evitar que el modelo dependa de ciertas conexiones específicas. Finalmente, en la capa de salida se encuentran 4 neuronas de salida con función de activación ‘softmax’. En la **Figura 27** se muestra la topología de la CNN propuesta.

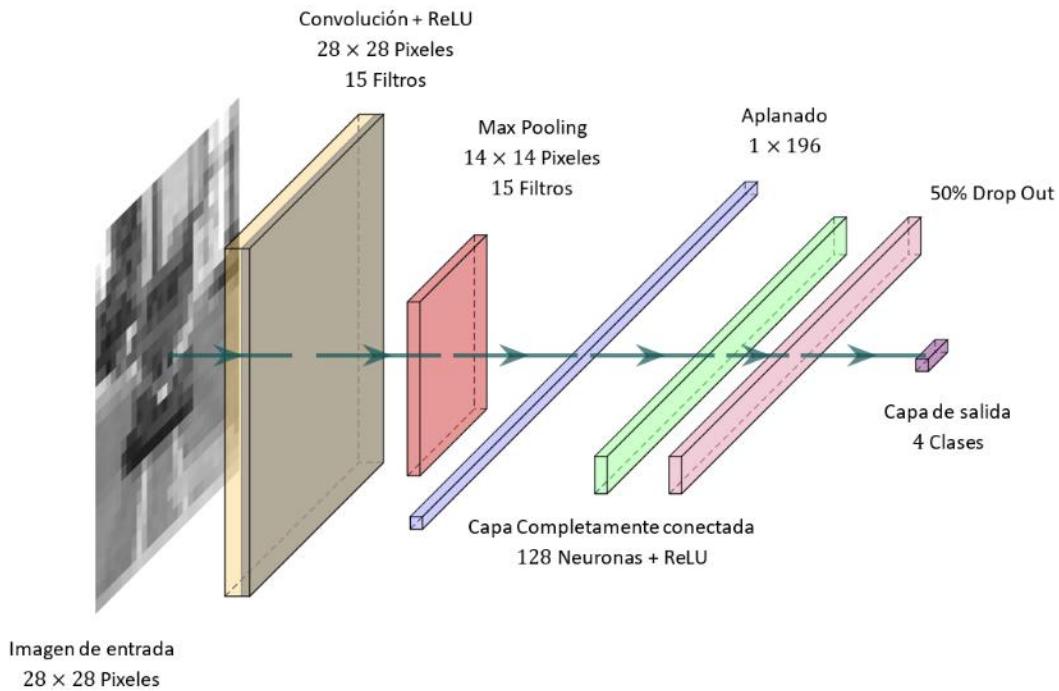


Figura 27: Arquitectura de CNN para detección de 4 clases.

Como se observa en la **Figura 27**, se aprecia que es una arquitectura con poca carga computacional en cuanto a complejidad de capas para esta aplicación.

3.3 Histograma para filtrado de señal de salida

Por cuestiones de interferencia de diversos factores como luz o interferencias, puede que la salida del clasificador sea una clase que no es la esperada cuando se incorpora al sistema de monitoreo EWA. Es por ello que se realiza un programa que monitoree la salida que arroja el clasificador, y acumule cada una durante un número de veces específico y arrojando como salida el mayor número de veces de la clase que registró en el acumulador. Este proceso se muestra en la **Figura 28**.

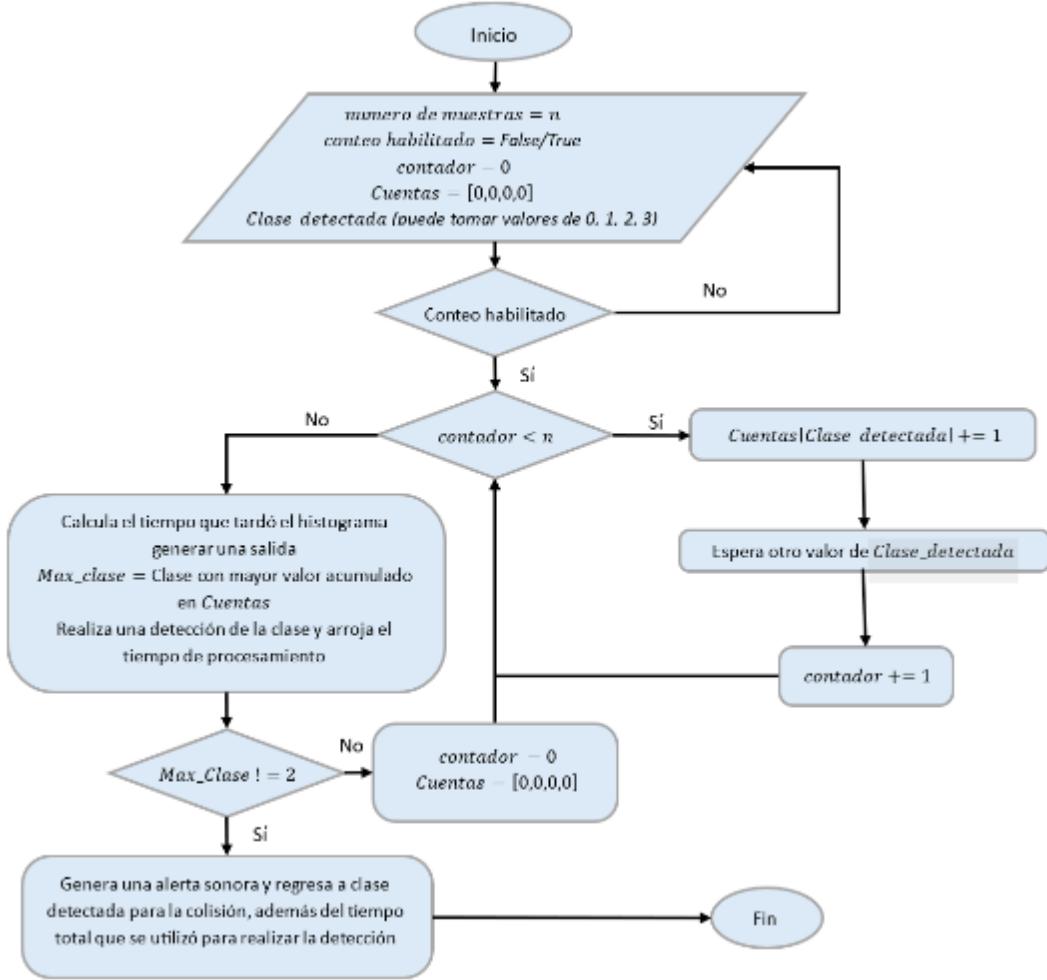


Figura 28: Diagrama de flujo de función “Histograma” para el filtrado de señal.

El programa tiene como entradas el número “ n ” de muestras a contemplar antes de arrojar una salida, una bandera “*conteo habilitado*” que activa el proceso de histograma, un contador que controla el número de veces que toma una muestra e histograma, el acumulado de la clase registrado como “*Cuentas*” y la “*clase detectada*” por la CNN. Si se habilita el histograma, se monitorea la clase que se detecta y se acumula durante “ n ” veces, al completar el periodo de monitoreo, se verifica qué clase tiene el mayor número de apariciones en el clasificador, y arroja como salida esta clase. Si es diferente a la clase “Ruta Libre” se genera una alerta de posible colisión, arroja el tiempo empleado en generar toda la clasificación desde la activación del histograma, hasta encontrar una clase que pueda ser una colisión con los objetivos propuestos en esta investigación.

3.4 Algoritmo de EWA

Una vez desarrollados todos los elementos del código utilizado para la clasificación, se implementa en conjunto con el algoritmo de captura y clasificación de imágenes por histograma, utilizando imágenes en blanco y negro, escaladas a un tamaño de 28×28 píxeles para identificar los posibles objetivos que puedan cruzarse en el camino del vehículo hospedador (**Figura 29**).

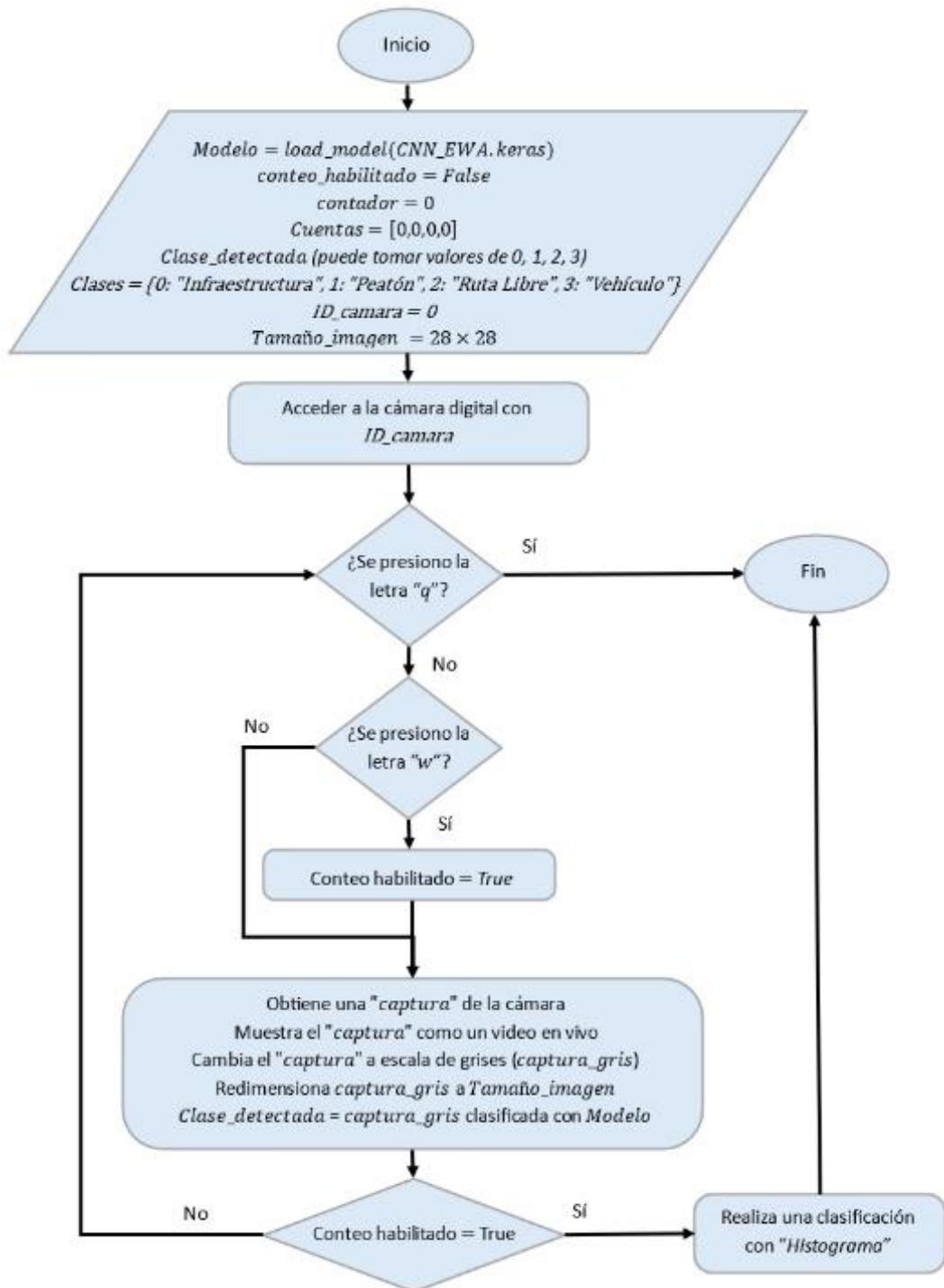


Figura 29: Diagrama de flujo de sistema EWA.

El algoritmo de EWA consiste en el modelo de DL entrenado para identificar 4 clases en cuestión previamente entrenado, este utiliza una cámara digital “*ID_camara*” que se encarga

de capturar activamente muestras del entorno desde el punto de vista del vehículo, una bandera de control Q que interrumpe todo el programa, un comando “w” que habilita el monitoreo por histograma para realizar las pruebas en la puesta experimental. Una vez habilitado el conteo haya sido activado cuando se alcanza la velocidad crucero en el punto de inicio, comienza a capturar imágenes del trayecto que sigue el vehículo en la pista de pruebas y las clasifica en tiempo real, las clasifica y si la bandera de conteo habilitado está activada, realiza la clasificación de histograma hasta detectar un objetivo diferente a ruta libre y arroja una alerta sonora.

3.5 Puesta experimental

Para validar el desempeño del sistema propuesto, se propone una puesta experimental que consiste en realizar una conducción del sistema en un entorno controlado a velocidades específicas en escenarios definidos, en la **Figura 30** se muestra las variables a considerar en esta puesta experimental.

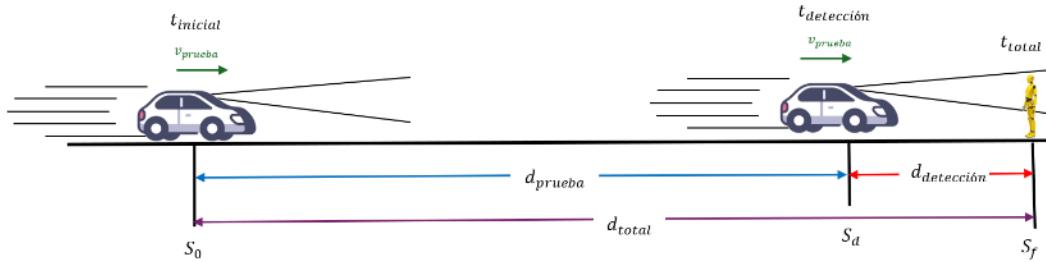


Figura 30: Puesta experimental propuesta para el sistema EWA

Las pruebas consisten en delimitar la distancia total d_{total} en la pista de pruebas para el trayecto del hospedador a una velocidad v_{prueba} definidas con base en el reglamento de tránsito para entornos urbanos las cuales son 5, 10, 15 y 20 Km/h, el hospedador ejecutará todo este tiempo el algoritmo de EWA para monitorear la presencia de un posible objetivo, se calculará la distancia d_{prueba} que recorrió el hospedador en función del tiempo el que detectó el objetivo en el camino, también, se considera la distancia que restaba para colisionar con el objetivo $d_{detección}$.

$$d_{detección} = d_{total} - t_{detección} \cdot v_{prueba} \quad (\text{Ec. 11})$$

$$t_{total} = \frac{v_{prueba}}{d_{total}} \quad (\text{Ec. 12})$$

$$t_{colisión} = t_{total} - t_{detección} \quad (\text{Ec. 13})$$

$$t_{colisión} \geq 1.33 \text{ s} \quad (\text{Ec. 14})$$

Cabe mencionar que, aunque el reglamento de tránsito permite un límite de velocidad de 40 km/h se decidió acotar la velocidad del vehículo hasta 20 km/h debido a que: 1) Se prefiere evitar el riesgo que puede ocasionar velocidades superiores a esta. 2) La pista permite conducir de manera segura solo a 20 km/h, y 3) La distancia que se requiere para alcanzar esa velocidad supera longitud de la pista de pruebas. En la Ec. 11 se calcula la distancia de detección en función de la velocidad de prueba y el tiempo de detección arrojado por el sistema EWA. Ec. 12 calcula el tiempo que tardaría el vehículo en recorrer la distancia de total a la velocidad de prueba. Y en Ec. 13 calcula el tiempo estimado de colisión, y si es menor que 1.33s (Ec. 14), se considera como una colisión inminente, teniendo así un parámetro para validar la efectividad del sistema. Se toma como valor de TTC igual a 1.33s como propuesta sin superar el valor mínimo de 1.2s dado por la ENCAP. EWA. En la **Figura 31** se muestra la pista de pruebas adecuada para las pruebas.

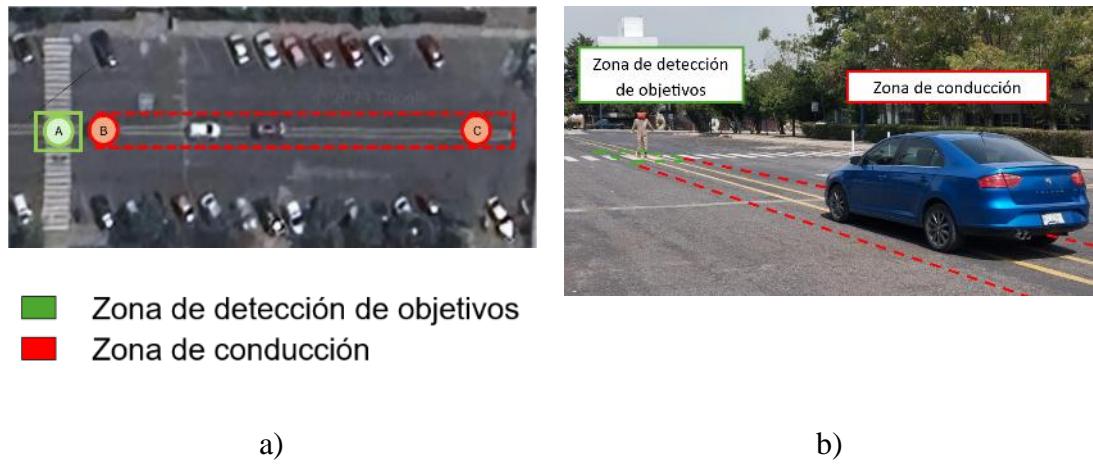


Figura 31: a) Pista de pruebas acondicionada para validación del sistema EWA y b) ejemplo real de escenario de prueba con clase “Peatón”.

En Figura 31a se muestra la pista de pruebas donde se llevará a cabo los experimentos de prueba del sistema EWA como se describió en la **Figura 30**, y en Figura 31b se muestra un ejemplo de una prueba preparada para identificar la clase “peatón”, cabe mencionar que en la zona de detección de objetivos se cambian dependiendo de la clase en cuestión analizada. El tiempo de procesamiento se considera desde el momento en que la cámara obtiene una imagen, cuando realiza las operaciones que se requiere para ajustar la imagen de 640×480 pixeles al tamaño de 28×28 pixeles y escala de grises necesaria para ser clasificada por la CNN. El tiempo total de procesamiento se ve influenciado en cuantas muestras se consideran para el filtrado por histograma, teniendo así la siguiente ecuación para el tiempo total.

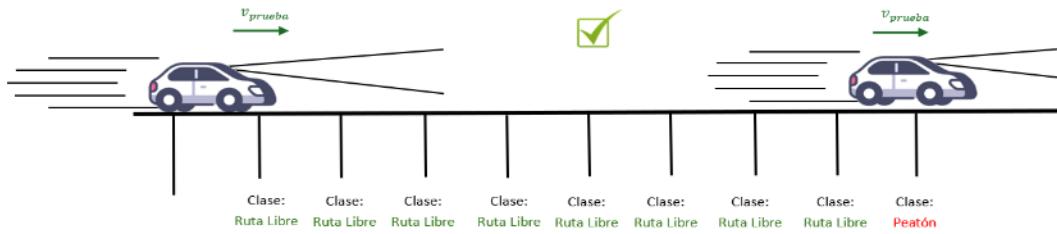
$$t_{prediccion} = \sum_{i=1}^n t_i \quad (\text{Ec. 15})$$

$$t_{prueba} = \sum t_{prediccion} \quad (\text{Ec. 16})$$

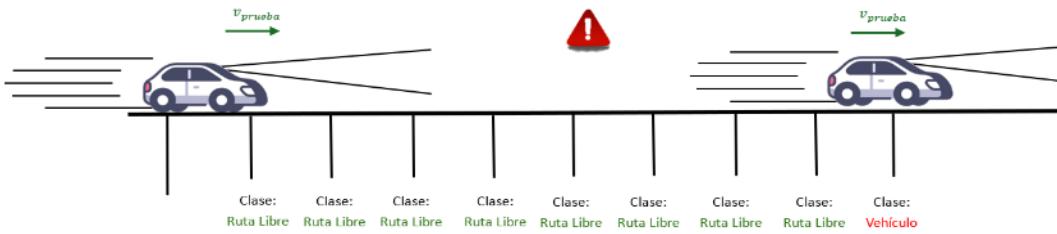
Donde $t_{prediccion}$ es el tiempo acumulado entre predicciones t_i que el histograma tarda en realizar n predicciones (Ec. 15) para realizar una detección. Y el tiempo de detección en las pruebas t_{prueba} (Ec. 16) es el tiempo que tardó en realizar una detección de la clase en cuestión.

3.5.1 Casos de pruebas: Objetivos

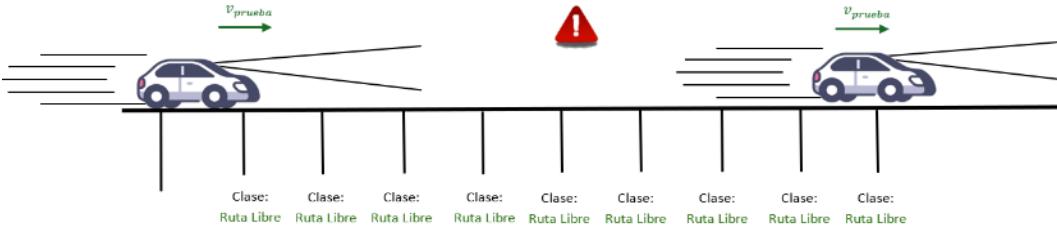
Para las pruebas para detección de objetivos, se considera el escenario cuando se lleva a cabo la puesta experimental descrita, conduciendo el vehículo hospedador a las velocidades de prueba constantes durante la distancia de conducción, ejecutando el algoritmo de captura de imágenes, clasificación y filtrado por Histograma que integran el sistema EWA, generando una salida en cada periodo de tiempo hasta detectar un objetivo y emitir una alerta ante una posible colisión con el objetivo de prueba. En la **Figura 32** se muestran los escenarios contemplados para que las pruebas sean exitosas o fallidas.



a)



b)



c)

Figura 32: Consideraciones adicionales para que una prueba sea a) exitosa o b) fallida por clasificación incorrecta de objetivo o c) fallida por no alertar en el caso de prueba con “objetivos”.

En la **Figura 32a** se considera como prueba exitosa en la cual detecta la clase analizada en cuestión (Peatón, en este caso) con un tiempo estimado de colisión (Ec. 14) superior a 1.33s y, en consecuencia, fallida es este tiempo es menor, en la **Figura 32b** se considera como una prueba fallida si el tiempo estimado de colisión cumple, pero la clase identificada es diferente a la del caso de prueba, y en la **Figura 32c** se contempla como una prueba fallida si el sistema no alerta de una posible colisión.

3.5.2 Casos de pruebas: Ruta Libre

Para garantizar el funcionamiento del sistema en una navegación segura sin presencia de objetivos en la pista de pruebas muestra en la **Figura 33**, se considera una prueba en el escenario mostrado en la **Figura 29**.

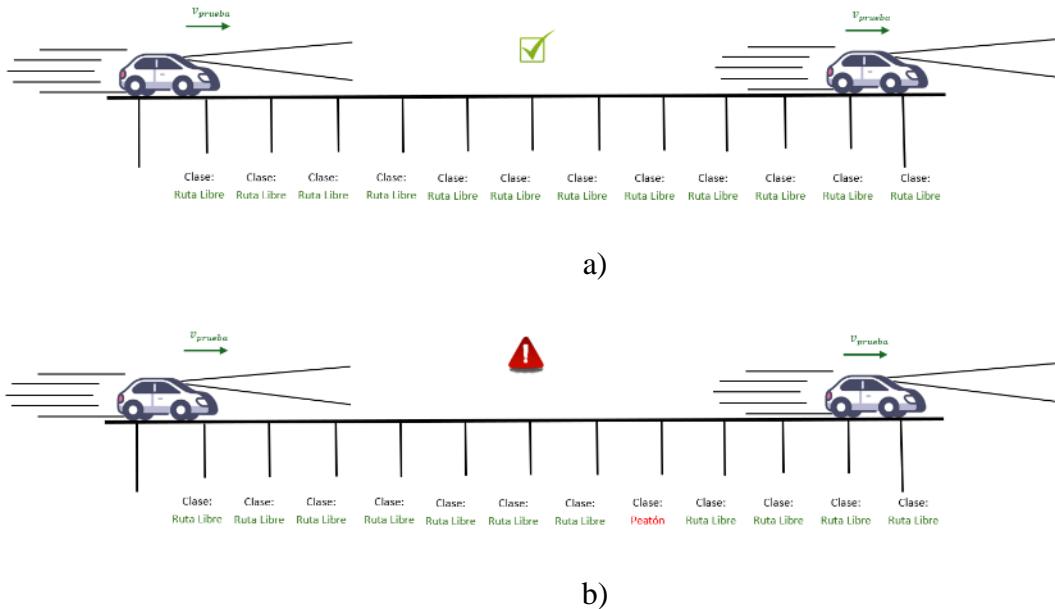


Figura 33: a) Prueba exitosa de conducción segura o ruta libre y b) prueba fallida al detectar un objetivo en una ruta libre.

En esta prueba, se considera como prueba exitosa si durante todo el trayecto se el sistema EWA no detecta ningún objetivo (**Figura 33a**), y si llega detectar algún objetivo por alguna interferencia se considera como una prueba fallida (**Figura 33b**). Para medir la fiabilidad del sistema se realiza un total de 40 pruebas de cada clase bajo condiciones controladas. Además de realizar un registro de la latencia y tiempos que tarda en realizar una alerta ante una colisión.

CAPÍTULO IV: PRUEBAS Y RESULTADOS

En este capítulo, se muestran los resultados de la implementación del algoritmo de adquisición, procesamiento, entrenamiento y clasificación de imágenes mediante la arquitectura propuesta de CNN y filtrado por histograma que conjuntan el sistema EWA en las pruebas descritas.

4.1 Resultados del entrenamiento y validación de la CNN

Al realizar el entrenamiento de la arquitectura de la CNN mostrada, cambiando el tamaño de las imágenes de entrada a 28×28 pixeles a un solo canal de color (Gris), se entrena un modelo de CNN en Matlab[®] para validar que el procesamiento de las imágenes es viable para la clasificación del sistema EWA en tiempo real. En la **Figura 34** se muestra el progreso del entrenamiento realizado y de la gráfica de la perdida con 4 épocas, optimizador con reducción del gradiente estocástico.

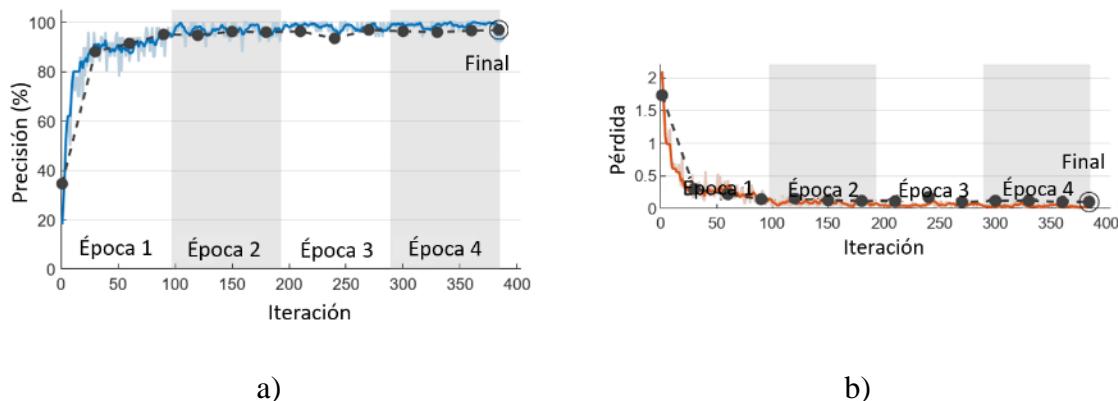


Figura 34: a) Precisión de la CNN implementada con las imágenes adquiridas y b) pérdida durante el entrenamiento

Como se puede apreciar, el utilizar imágenes en escala de grises y escaladas se logra obtener una precisión del 98.91%, con una pérdida de 5.35% final con 4 épocas y 400

iteraciones, además de la disminución de la perdida mientras aumenta la precisión, lo que indica que no existen problemas de sobreajuste. Al utilizar un 20% de las imágenes (400 por clase) para prueba, se sintetizan los resultados en la siguiente matriz de confusión de las imágenes al utilizarlas con la CNN entrenada (**Figura 35**).

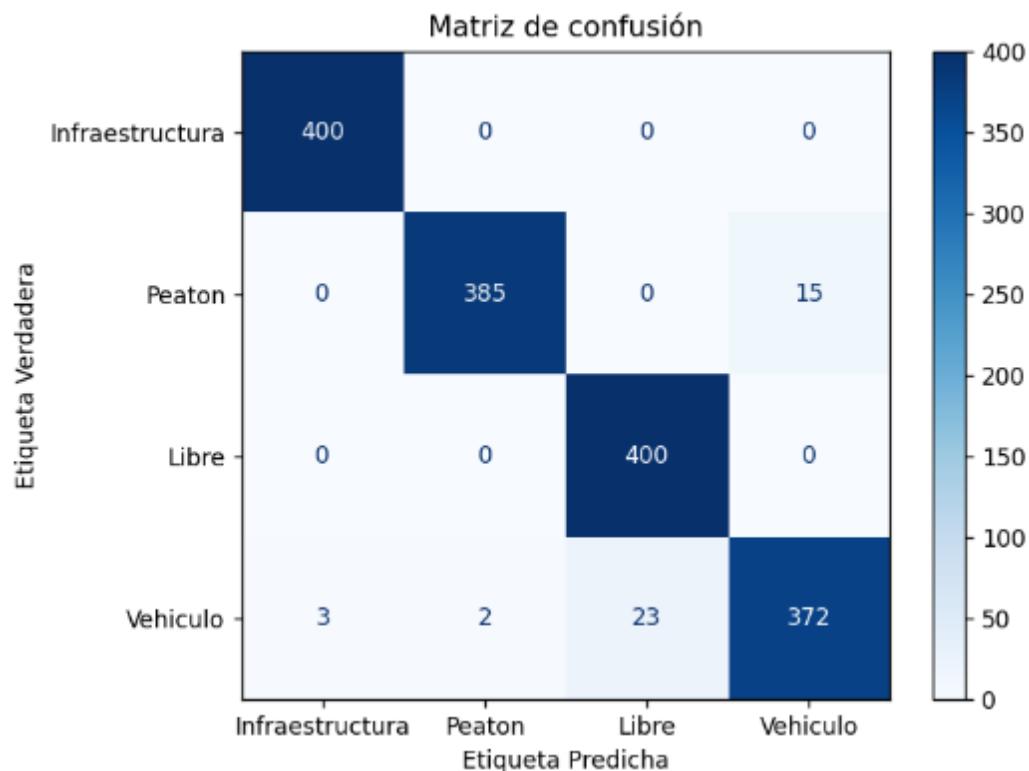
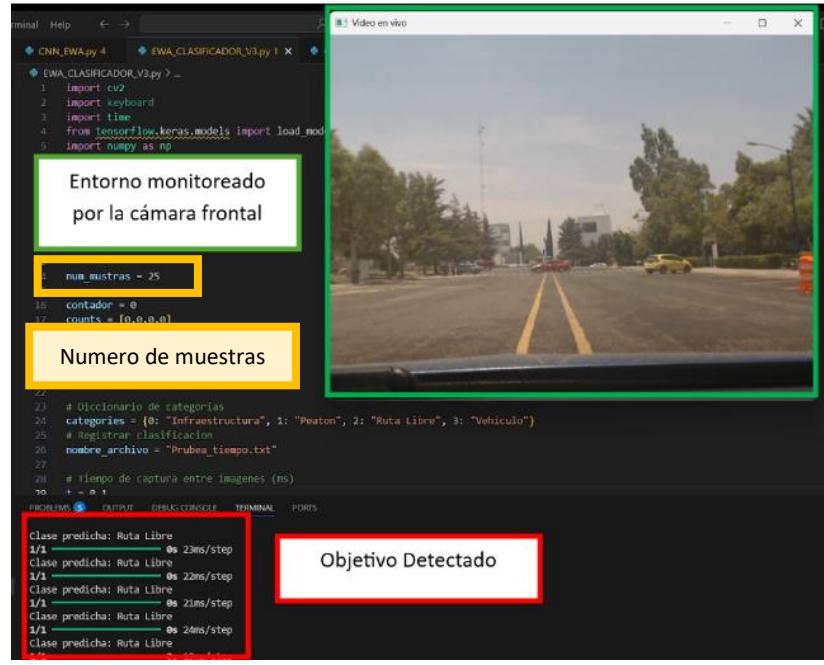


Figura 35. Matriz de confusión con 20% de imágenes por clase para prueba de CNN.

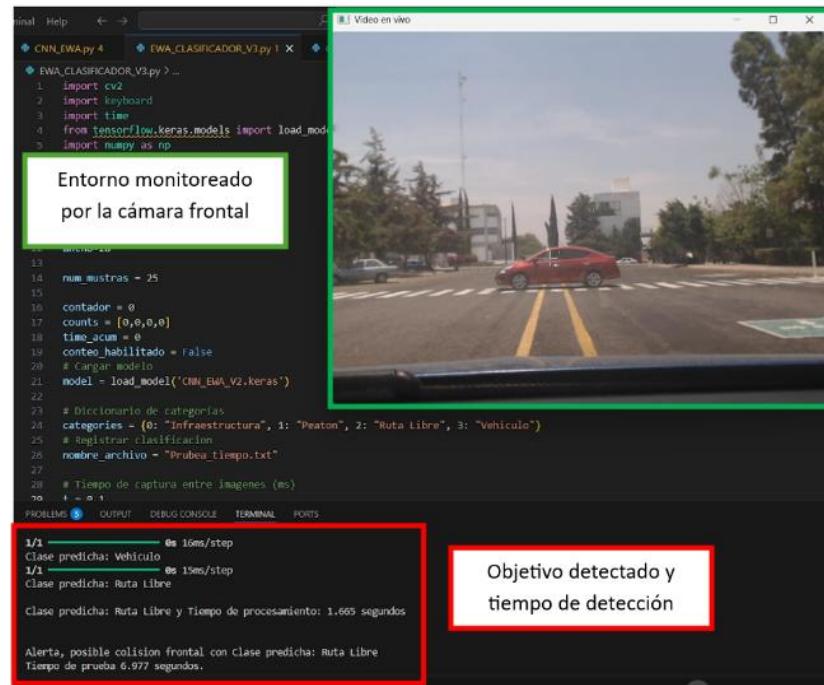
Se puede observar que la matriz de confusión tiene un 97.31% de efectividad en identificar las imágenes y esto es debido a que las imágenes contienen patrones muy diferentes lo que facilita a la CNN identificarlos con alta precisión. Sin embargo, todavía no es posible predecir una posible colisión en un entorno real, en consecuencia, es necesario integrar la CNN al sistema EWA para la identificación oportuna de los entornos descritos en la puesta experimental y medir así su efectividad.

4.2 Resultados del desempeño del sistema en pista de pruebas

En esta sección se presenta la interfaz usada para cada una de las pruebas, así como, los resultados obtenidos por cada clase. Al implementar el sistema EWA en el vehículo hospedador, se realiza la puesta experimental para ver la precisión al momento de detectar la presencia o ausencia de algún objetivo. En la **Figura 36** se muestra un ejemplo de la interfaz usada para las pruebas del sistema, en este caso corresponde a una prueba de la clase “vehículo”.



a)



b)

Figura 36. Interfaz usada para las pruebas del sistema a) inicio de la prueba a 40 m de distancia del objetivo y b) Parte final de la prueba en zona de detección con el vehículo hospedador moviéndose a una velocidad de 15 m/s.

En la **Figura 36a** se muestra que el sistema clasifica la entrada como “ruta libre” mientras se encuentra en el inicio de la pista a 40 m del objetivo, lo cual es correcto dado que no hay una situación de colisión inminente. experimentalmente, se obtuvo un tiempo promedio de 50 a 100 ms para capturar una imagen y realizar su clasificación. Y con un numero de muestras $n = 25$ para realizar el histograma se obtuvo un tiempo de detección entre 1.6s. En la Figura 36b se muestra el momento en que el vehículo es detectado y genera la alerta de manera visual en la interfaz y también de manera sonora. En este ejemplo en particular, el tiempo total de la prueba es de 6.97s que representa el tiempo desde que el vehículo hospedador empieza a moverse hasta que se levanta la alerta, esta prueba se repite 40 veces por cada clase con las velocidades descritas en la sección 3.5, el criterio para definir una prueba como *pasada* o *fallida* está basado en un umbral de tiempo para colisión mayor o igual a 1.33 s, como se explica en la sección 3.5.

4.2.1 Resultados de desempeño del sistema en la clase Peatón.

En esta sección, se muestra los resultados obtenidos en la puesta experimental descrita en la sección 3.5.1.

En la **Figura 37** se muestra de manera gráfica una comparación del TTC contra el número de prueba realizada. En la Figura 40 se muestran las comparaciones del número de pruebas contra el tiempo estimado de colisión, además de contemplar el criterio de alerta oportuna siendo el tiempo ante una colisión inminente, representado con una línea horizontal con el valor de 1.33s. Siendo los tiempos menores a este los que no son detectados a tiempo por el sistema EWA, además de resaltar con un ovalo rojo aquellos que a pesar de cumplir con el criterio de TTC, no cumple con el criterio de detección de la clase en cuestión a analizar y tomándolo así como una prueba fallada.

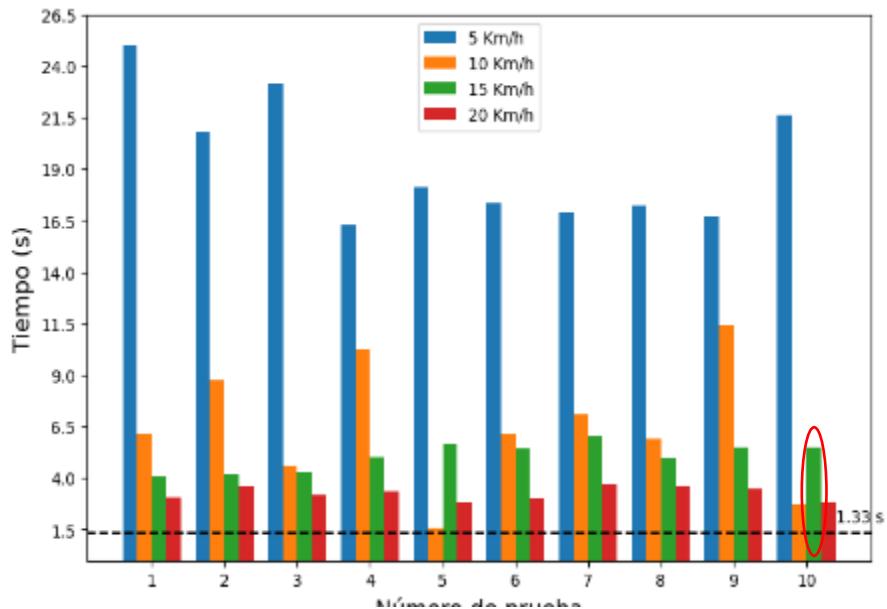


Figura 37. TTC obtenido por el sistema EWA en la clase “Peatón”.

Como se puede observar en la **Figura 37** que los TTC en las pruebas con todas las velocidades son superiores al umbral establecido. Sin embargo, en la prueba 10 correspondiente a 15 Km/h se cataloga como una prueba fallida, ya que a pesar de haber cumplido con el criterio de TTC, el sistema identificó a la clase como “Vehículo”, como se esperaba si se toma en cuenta los resultados de la matriz de confusión en la **Figura 35**, esto debido a la similitud de patrones que existen en la captura de estas imágenes. En la **Tabla 1** se muestran los resultados numéricos obtenidos la prueba.

Tabla 1: Resultados de tiempo de detección en clase “peatón” con una distancia inicia de 40m a velocidades definidas

Prueba	Velocidad km/h (m/s)	Tiempo de prueba (s)	Alerta	Tiempo de detección (s)	Distancia de detección (m)	Tiempo a colisión (s)	Resultado
1	5 (1.39)	28.8	Sí	3.81	34.708	24.99	Aprobado
2			Sí	8.01	28.875	20.79	Aprobado
3			Sí	5.62	32.194	23.18	Aprobado
4			Sí	12.5	22.639	16.3	Aprobado
5			Sí	10.65	25.208	18.15	Aprobado
6			Sí	11.45	24.097	17.35	Aprobado
7			Sí	11.93	23.431	16.87	Aprobado
8			Sí	11.55	23.958	17.25	Aprobado
9			Sí	12.07	23.236	16.73	Aprobado
10			Sí	7.16	30.056	21.64	Aprobado
11	10 (2.78)	14.4	Sí	8.23	17.139	6.17	Aprobado
12			Sí	5.65	24.306	8.75	Aprobado
13			Sí	9.79	12.806	4.61	Aprobado
14			Sí	4.15	28.472	10.25	Aprobado
15			Sí	12.82	4.389	1.58	Aprobado
16			Sí	8.23	17.139	6.17	Aprobado
17			Sí	7.28	19.778	7.12	Aprobado
18			Sí	8.51	16.361	5.89	Aprobado
19			Sí	2.97	31.750	11.43	Aprobado
20			Sí	11.68	7.556	2.72	Aprobado
21	15 (4.16)	9.6	Sí	5.55	16.875	4.05	Aprobado
22			Sí	5.4	17.500	4.2	Aprobado
23			Sí	5.29	17.958	4.31	Aprobado
24			Sí	4.56	21.000	5.04	Aprobado
25			Sí	3.91	23.708	5.69	Aprobado
26			Sí	4.16	22.667	5.44	Aprobado
27			Sí	3.54	25.250	6.06	Aprobado
28			Sí	4.61	20.792	4.99	Aprobado
29			Sí	4.13	22.792	5.47	Aprobado
30			No	4.13	22.792	5.47	Falló
31	20 (5.16)	7.2	Sí	4.14	17.000	3.06	Aprobado
32			Sí	3.57	20.167	3.63	Aprobado
33			Sí	4.04	17.556	3.16	Aprobado
34			Sí	3.83	18.722	3.37	Aprobado
35			Sí	4.37	15.722	2.83	Aprobado
36			Sí	4.18	16.778	3.02	Aprobado
37			Sí	3.49	20.611	3.71	Aprobado
38			Sí	3.59	20.056	3.61	Aprobado
39			Sí	3.69	19.500	3.51	Aprobado
40			Sí	4.35	15.833	2.85	Aprobado

Cabe mencionar que el Sistema en la clase “Peatón” detectó correctamente 39 veces y solamente 1 prueba fue fallida según los casos tomados en cuenta.

4.2.2 Resultados de desempeño del sistema en la clase Infraestructura

En este apartado se discuten los resultados obtenidos del sistema EWA con la clase “infraestructura”. En la **Figura 38** se muestra de manera gráfica una comparación del TTC contra el número de prueba realizada.

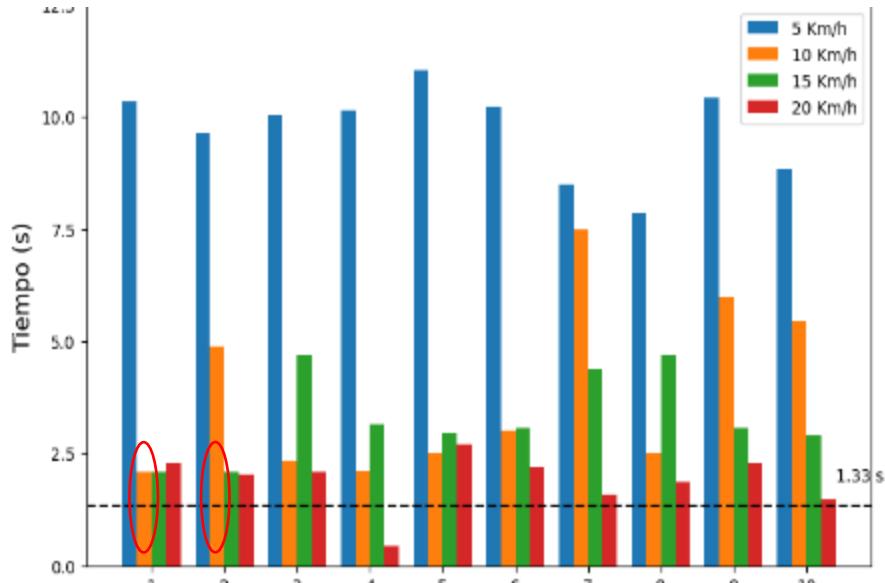


Figura 38. TTC obtenido por el sistema EWA en la clase “Peatón”.

La **Figura 38** muestra que el escenario de la clase “infraestructura” tiene dos errores en cuanto a la clasificación detectando la clase “Vehículo” correspondientes a la velocidad de 10 km/h en las pruebas 1 y 2, esto debido a que en la matriz de confusión (**Figura 35**) muestra que con los conjuntos de prueba tiene una efectividad del 100%, se observa también que se clasifica la clase de “vehículo” en la clase de “infraestructura”, de igual forma se deduce que es por el parecido de los patrones similares que comparten estas clases. Otro resultado muestra en la prueba 4 a la velocidad de 20 Km/h no cumple con el criterio del TTC siendo este menor al umbral de 1.33 s. En la tabla 2 se muestra el resultado numérico obtenido de las 40 pruebas realizadas del sistema EWA con la clase “Infraestructura”.

Tabla 2: Resultados de tiempo de detección en clase infraestructura con una distancia inicia de 40m a velocidades definidas

Prueba	Velocidad km/h (m/s)	Tiempo de prueba (s)	Alerta	Tiempo de detección (s)	Distancia de detección (m)	Tiempo a colisión (s)	Resultado
1	5 (1.39)	28.8	Sí	18.45	14.375	10.35	Aprobado
2			Sí	19.17	13.375	9.63	Aprobado
3			Sí	18.77	13.931	10.03	Aprobado
4			Sí	18.66	14.083	10.14	Aprobado
5			Sí	17.764	15.328	11.036	Aprobado
6			Sí	18.56	14.222	10.24	Aprobado
7			Sí	20.32	11.778	8.48	Aprobado
8			Sí	20.95	10.903	7.85	Aprobado
9			Sí	18.37	14.486	10.43	Aprobado
10			Sí	19.97	12.264	8.83	Aprobado
11	10 (2.78)	14.4	Sí	12.32	5.778	2.08	Aprobado
12			Sí	9.5	13.611	4.9	Aprobado
13			Sí	12.05	6.528	2.35	Aprobado
14			Sí	12.27	5.917	2.13	Aprobado
15			Sí	11.9	6.944	2.5	Aprobado
16			Sí	11.39	8.361	3.01	Aprobado
17			Sí	6.9	20.833	7.5	Aprobado
18			Sí	11.9	6.944	2.5	Aprobado
19			Sí	8.43	16.583	5.97	Aprobado
20			Sí	8.95	15.139	5.45	Aprobado
21	15 (4.16)	9.6	No	7.5	8.750	2.1	Falló
22			No	7.5	8.750	2.1	Falló
23			Sí	4.9	19.583	4.7	Aprobado
24			Sí	6.46	13.083	3.14	Aprobado
25			Sí	6.64	12.333	2.96	Aprobado
26			Sí	6.53	12.792	3.07	Aprobado
27			Sí	5.21	18.292	4.39	Aprobado
28			Sí	4.9	19.583	4.7	Aprobado
29			Sí	6.53	12.792	3.07	Aprobado
30			Sí	6.7	12.083	2.9	Aprobado
31	20 (5.16)	7.2	Sí	4.9	12.778	2.3	Aprobado
32			Sí	5.16	11.333	2.04	Aprobado
33			Sí	5.1	11.667	2.1	Aprobado
34			Sí	6.76	2.444	0.44	Falló
35			Sí	4.5	15.000	2.7	Aprobado
36			Sí	5	12.222	2.2	Aprobado
37			Sí	5.61	8.833	1.59	Aprobado
38			Sí	5.33	10.389	1.87	Aprobado
39			Sí	4.9	12.778	2.3	Aprobado
40			Sí	5.73	8.167	1.47	Aprobado

Con base en la Tabla 2 se muestra que da como resultado un total de 3 pruebas fallidas y 37 pruebas exitosas.

4.2.3 Resultados de desempeño del sistema en la clase Vehículo

Finalmente, se analizan los resultados de la Figura 39 de la clase “Vehículo”.

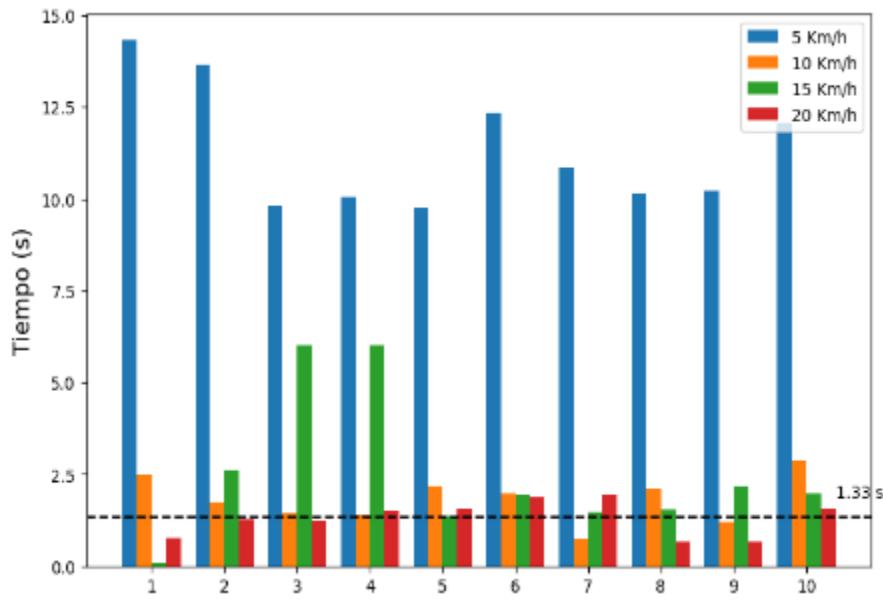


Figura 39: Resultados de tiempo estimado a colisión en pruebas.

La clase de vehículo tiene 6 errores debido al criterio de TTC y en contraste con lo que se esperaría en la matriz de confusión, este fue la principal razón de que el sistema fallara, y siempre alertó por la presencia de la clase “vehículo”. El motivo por lo que tuvo más error en esta clase es debido a que la iluminación del vehículo cambia por su material metálico reflectivo del que está hecho, además de afectar su perspectiva. En la Tabla 3 se muestran los resultados generales en los cuales el sistema EWA logró pasar las pruebas y los datos recabados durante la puesta experimental de la clase “Vehículo”.

Tabla 3: Resultados de tiempo de detección en clase infraestructura con una distancia inicia de 40m a velocidades definidas

Prueba	Velocidad km/h (m/s)	Tiempo de prueba (s)	Alerta	Tiempo de detección (s)	Distancia de detección (m)	Tiempo a colisión (s)	Resultado
1	5 (1.39)	28.8	Sí	14.47	19.903	14.33	Aprobado
2			Sí	15.13	18.986	13.67	Aprobado
3			Sí	18.97	13.653	9.83	Aprobado
4			Sí	18.76	13.944	10.04	Aprobado
5			Sí	19.05	13.542	9.75	Aprobado
6			Sí	16.46	17.139	12.34	Aprobado
7			Sí	17.95	15.069	10.85	Aprobado
8			Sí	18.63	14.125	10.17	Aprobado
9			Sí	18.58	14.194	10.22	Aprobado
10			Sí	16.73	16.764	12.07	Aprobado
11	10 (2.78)	14.4	Sí	11.93	6.861	2.47	Aprobado
12			Sí	12.65	4.861	1.75	Aprobado
13			Sí	12.95	4.028	1.45	Aprobado
14			Sí	13	3.889	1.4	Aprobado
15			Sí	12.23	6.028	2.17	Aprobado
16			Sí	12.43	5.472	1.97	Aprobado
17			Sí	13.68	2.000	0.72	Falló
18			Sí	12.29	5.861	2.11	Aprobado
19			Sí	13.21	3.306	1.19	Falló
20			Sí	11.51	8.028	2.89	Aprobado
21	15 (4.16)	9.6	Sí	9.5	0.417	0.1	Falló
22			Sí	6.97	10.958	2.63	Aprobado
23			Sí	3.59	25.042	6.01	Aprobado
24			Sí	3.59	25.042	6.01	Aprobado
25			Sí	8.22	5.750	1.38	Aprobado
26			Sí	7.66	8.083	1.94	Aprobado
27			Sí	8.14	6.083	1.46	Aprobado
28			Sí	8.05	6.458	1.55	Aprobado
29			Sí	7.43	9.042	2.17	Aprobado
30			Sí	7.62	8.250	1.98	Aprobado
31	20 (5.16)	7.2	Sí	6.42	4.333	0.78	Falló
32			Sí	5.92	7.111	1.28	Falló
33			Sí	5.97	6.833	1.23	Falló
34			Sí	5.68	8.444	1.52	Aprobado
35			Sí	5.63	8.722	1.57	Aprobado
36			Sí	5.33	10.389	1.87	Aprobado
37			Sí	5.25	10.833	1.95	Aprobado
38			Sí	6.54	3.667	0.66	Falló
39			Sí	6.53	3.722	0.67	Falló
40			Sí	5.62	8.778	1.58	Aprobado

Con estas pruebas, el sistema con la clase “vehículo” tuvo 8 pruebas fallidas y 32 pruebas exitosas. Para la clase de “ruta libre” se realizaron las 40 pruebas como se describieron en la sección 3.5.2 y todas resultaron exitosas al no detectar nada mientras se conducía a las

velocidades establecidas, dando un total global de 160 pruebas del sistema y un total de 14 pruebas fallidas contando todos los casos, teniendo una efectividad de 91.25% en la predicción de colisiones frontales con la puesta experimental planteada.

4.3 Comparación del sistema con trabajos similares

En la Tabla 4 se muestra una comparativa de los sistemas similares encontrado en la literatura con el sistema EWA propuesto.

Tabla 4. Comparativa de sistema EWB con trabajos similares

Autor	Sensor(es) Utilizado(s)	Tiempo de latencia (s)	Simulación o entorno real	Velocidad(es) de prueba (km/h)	Número de clases	Pruebas Realizadas	Probado en vehículo eléctrico	Precisión
Deo et al. (2021)	Cámara y LiDAR	0.0485	Simulación	30 – 70	1	3	✗	67–70%
Gulino et al. (2023)	LiDAR / RaDAR / Cámaras / GNSS / IMU	0.3	Simulación	50	1	5000>	✗	100%
Lai & Yang (2023)	Radar frontal (del módulo PreScan) y cámaras	No registrado	Simulación	50 y 60	1	18	✗	100%
Dai et al. (2024)	LiDAR RoboSense RS-16 y Cámara monocular Q20	No registrado	Simulación	< 30	2	No registrado	✗	94%
Shaout & Castaneda-Trejo (2025)	Ultrasonido (HC-SR04)	0.0011 – 0.0015	Simulación	No registrado	1	10	✗	95%
Liu et al. (2025)	Fusión de cámara + radar (City Safety)	1.79	Ambos	20 – 40	2	10	✗	100%
Propuesta	Cámara digital	1.66	Ambos	5, 10, 15 y 20	4	160	✓	91.25 %

De la Tabla 4 se muestran trabajos similares y más relevantes que abordan sistemas similares al desarrollado en este Trabajo de Tesis. Como se puede observar, en 5 de ellos fueron llevados a cabo en simulación, y si bien sus tiempos de latencia muy rápidos, estos toman en cuenta los tiempos entre captura o el tiempo que tarda en actualizarse su simulación y realizar una predicción de los datos obtenidos en entornos reales y después analizados en “offline”. En cuanto al trabajo realizado por Liu et al. Fue similar en cuestión de puesta experimental, además de enfocarse en la dinámica del vehículo y tomando en cuenta tiempos de monitoreo con sensores de proximidad y cámaras, análisis e intervención del sistema, lo que tiene concordancia con el tiempo obtenido en sus pruebas reales. Además, se observa que el sistema propuesto mejoró en un tiempo de 13 ms con 2 objetivos más que el trabajo en cuestión.

4.4 Pruebas realizadas en vehículo eléctrico.

Una vez que el sistema EWA fue probado en condiciones controladas, se realiza una serie de pruebas finales donde se instala el sistema en el vehículo eléctrico de la Facultad de Ingeniería EFACI. Las pruebas en esta sección consisten en conducir el vehículo dentro de las instalaciones de la Universidad Autónoma de Querétaro, Campus San Juan del Río. En la **Figura 40** se muestra el vehículo eléctrico acondicionado.

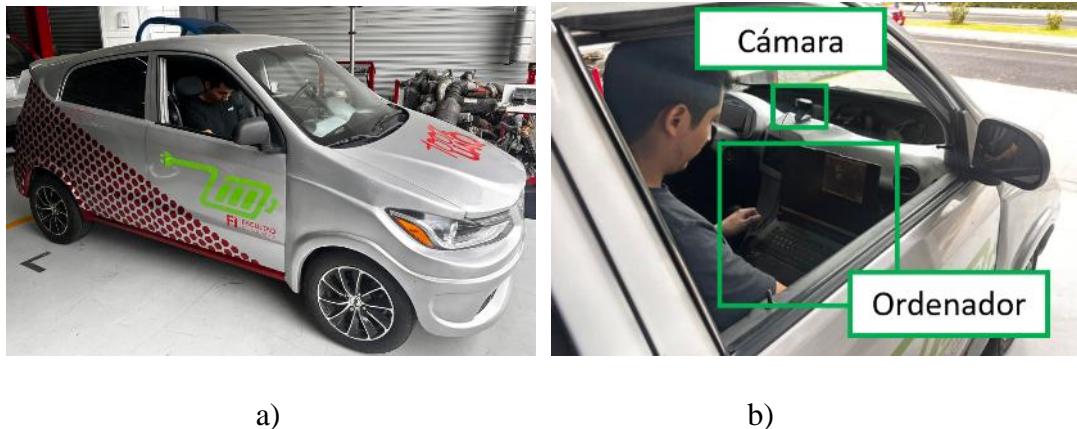


Figura 40. a) Vehículo Eléctrico Acondicionado para pruebas en escenarios reales.

En la **Figura 40 a)** se muestra el vehículo eléctrico que se utiliza en la prueba y en la Figura 35b se muestra el sistema EWA montado en el vehículo. Se realiza una única prueba que consiste en iniciar el sistema EWA en el circuito universitario donde hay tráfico concurrente

de peatones y vehículos, bajo condiciones no controladas. Se realiza solamente un único recorrido para no sobrepasar el límite de uso de la batería del vehículo, y además de realizar esta prueba puesto que se espera registrar el desempeño del sistema EWA en situaciones no controladas en el vehículo eléctrico, ya que el sistema fue probado en situaciones controladas. En la Figura 36 se muestra el circuito que sigue el vehículo eléctrico para esta de prueba.

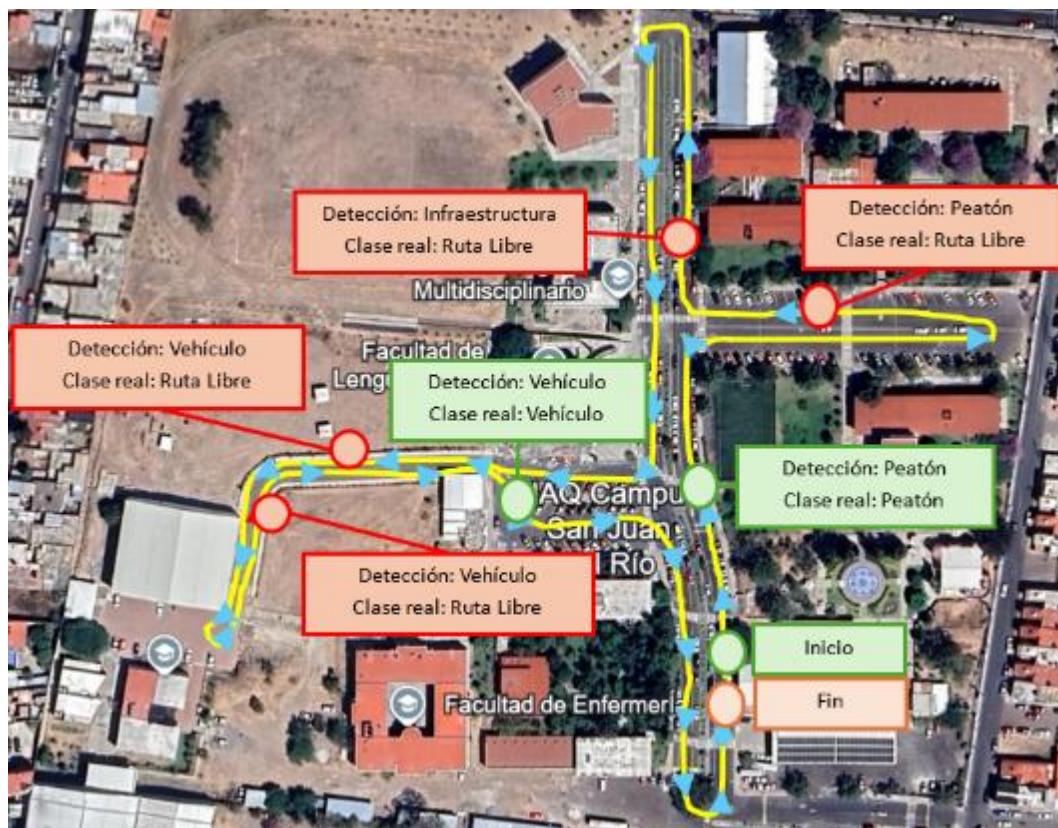


Figura 41. Circuito donde se conduce el vehículo eléctrico dentro de las instalaciones de la Universidad Autónoma de Querétaro, Campus San Juan del Río.

En la **Figura 41** se muestra el recorrido que sigue el vehículo hospedador, así como, se muestra mediante puntos el instante en que el sistema detecta algún obstáculo. Al realizar la conducción del vehículo eléctrico con el sistema EWA se registraron 6 detecciones: dos de ellas fueron detecciones acertadas (color verde) mientras que en los 4 restantes alertó cuando no se presentaba ningún objetivo, también se contempla el hecho que las veces que no detectó un

obstáculo se contaron como pruebas exitosas. Es importante resaltar que, aunque parezca que tenga un bajo desempeño en situaciones reales, el sistema fue acertado en detectar la ausencia de obstáculos, es decir, en detectar una “ruta libre” cuando no había objetivos cerca, esto es el 90% del tiempo. Otra cosa que hay que tener en cuenta es que la prueba se realizó con condiciones de iluminación distinta a en la que se adquirieron las imágenes de los objetivos, además de mostrar una gran densidad diferente de peatones y vehículos de la esperada.

CAPÍTULO V: CONCLUSIONES

En este proyecto de tesis, se realizó un sistema de alerta al conductor EWA, ante posibles colisiones frontales con imágenes y DL, acondicionando un vehículo para pruebas controladas y después un vehículo eléctrico para realizar pruebas en capo, para alertar sobre posibles colisiones frontales. Se comenzó por definir la puesta experimental, en la cual se delimitó en el estacionamiento de la Facultad de Ingeniería, además se acondicionó un vehículo con un sistema de visión para adquirir imágenes para entrenar un algoritmo de clasificación que integra el sistema EWA, y probarlo antes de ser puesto a prueba en un entorno escolar. Para esto, se creó un programa de adquisición de imágenes para capturar 2,000 imágenes por cada uno de los 4 objetivos: Vehículo, Peatón, Vehículo y Ruta libre que serían los casos que se tomarían en cuenta para el sistema EWA y se ejecutó dicho programa con el sistema de adquisición de imágenes. Posteriormente, se realizó un post-procesamiento de imágenes que reducía las imágenes obtenidas a un tamaño de 28×28 pixeles en escala de grises para disminuir la carga computacional a la hora de implementar en tiempo real la clasificación. Se implementó una topología CNN propuesta para la clasificación de las imágenes de los objetivos obteniendo un 97.1% de precisión en la etapa de validación y un tiempo promedio de 50 ms entre clasificación. Se integró la CNN al sistema EWA, junto con la adquisición de imágenes y un filtrado propuesto por histograma, obteniendo una latencia entre predicción de 1.6 s sobre la presencia alguno de los objetivos en el camino. Las pruebas en campo para el sistema EWA en el vehículo de pruebas consisten en; se conduce el vehículo con el sistema EWA ejecutándose y se realizaron 40 pruebas en una pista de 40 m ubicando al final del recorrido uno de los 4 objetivos, a velocidades de 5, 10, 15 y 20 m/s, habiendo probado el sistema un total de 160 veces, logrando un TTC superior al umbral de 1.33s y teniendo una precisión del 91.25% en condiciones controladas. Finalmente, se realizó una prueba en condiciones no controladas en la cual el sistema realizó 2 detecciones correctas y 4 detecciones erróneas. En cuanto a las detecciones erróneas, se mostraba desde las pruebas en condiciones controladas que las variaciones de luz, enfoque, vibraciones afectaban la precisión del sistema, en investigaciones futuras se pretende analizar más casos para crear un sistema más robusto con una topología de CNN más completa y optimización en tiempo de detecciones, además de realizar una prueba

con fusión de sensores, además de integrar un actuador que permita crear la función de EBA y frenar de manera activa.

BIBLIOGRAFÍA

Aoki, S., Fujinami, Y., Raksincharoensak, P., & Henze, R. (2024). Proactive Braking Control System for Collision Avoidance during Right Turns with Occluded Vision at an Intersection. *Applied Sciences (Switzerland)*, 14(6). <https://doi.org/10.3390/app14062661>

Cheng, C. H., Lin, H. H., & Luo, Y. Y. (2025). Research on Autonomous Vehicle Lane-Keeping and Navigation System Based on Deep Reinforcement Learning: From Simulation to Real-World Application. *Electronics (Switzerland)*, 14(13). <https://doi.org/10.3390/electronics14132738>

Dai, Z., Guan, Z., Chen, Q., Xu, Y., & Sun, F. (2024). Enhanced Object Detection in Autonomous Vehicles through LiDAR—Camera Sensor Fusion. *World Electric Vehicle Journal*, 15(7). <https://doi.org/10.3390/wevj15070297>

Deo, A., Palade, V., & Huda, M. N. (2021). Centralised and decentralised sensor fusion-based emergency brake assist. *Sensors*, 21(16). <https://doi.org/10.3390/s21165422>

ENCAP. (2023). *AUTONOMOUS EMERGENCY BRAKING SYSTEM [FOR PEDESTRIAN DAYTIME] PERFORMANCE TEST PROCEDURE*.

Gonzalez, R. C., & Woods, R. E. (2018). *Digital Image Processing* (4th Edition).

Gulino, M. S., Vangi, D., & Damaziak, K. (2023). Performance Assessment in a “Lane Departure” Scenario of Impending Collision for an ADAS Logic Based on Injury Risk Minimisation. *Designs*, 7(3). <https://doi.org/10.3390/designs7030059>

Gunjate, S. S., & Khot, S. A. (2023). A Systematic Review of Emergency Braking Assistant System to Avoid Accidents Using Pulse Width Modulation and Fuzzy Logic Control Integrated with Antilock Braking. In *International Journal of Automotive and Mechanical Engineering* (Vol. 20, Issue 2, pp. 10457–10479). Universiti Malaysia Pahang. <https://doi.org/10.15282/ijame.20.2.2023.10.0808>

Hu, G., Li, K., Lu, W., Chen, O., Sun, C., & Zhao, Y. (2025). A Proactive Collision Avoidance Model for Connected and Autonomous Vehicles in Mixed Traffic Flow. *World Electric Vehicle Journal*, 16(7). <https://doi.org/10.3390/wevj16070394>

Jain, A., Bansal, R., Kumar, A., & Singh, K. (2015). A comparative study of visual and auditory reaction times on the basis of gender and physical activity levels of medical first year students. *International Journal of Applied and Basic Medical Research*, 5(2), 124. <https://doi.org/10.4103/2229-516x.157168>

Jeong, H.-S., & Kim, J.-H. (2025). Empirical Validation of a Multidirectional Ultrasonic Pedestrian Detection System for Heavy-Duty Vehicles Under Adverse Weather Conditions. *Sensors*, 25(17), 5287. <https://doi.org/10.3390/s25175287>

Jeong, Y. (2022). Interactive Lane Keeping System for Autonomous Vehicles Using LSTM-RNN Considering Driving Environments. *Sensors*, 22(24). <https://doi.org/10.3390/s22249889>

Kim, W., Yang, H., & Kim, J. (2023). Blind Spot Detection Radar System Design for Safe Driving of Smart Vehicles. *Applied Sciences (Switzerland)*, 13(10). <https://doi.org/10.3390/app13106147>

Lai, F., & Yang, H. (2023). Integrated Longitudinal and Lateral Control of Emergency Collision Avoidance for Intelligent Vehicles under Curved Road Conditions. *Applied Sciences (Switzerland)*, 13(20). <https://doi.org/10.3390/app132011352>

Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444. <https://doi.org/10.1038/nature14539>

Lee, K., & Lee, C. (2025). String Stability Analysis and Design Guidelines for PD Controllers in Adaptive Cruise Control Systems. *Sensors*, 25(11). <https://doi.org/10.3390/s25113518>

Lee, Y., & Park, M. (2025). Rearview Camera-Based Blind-Spot Detection and Lane Change Assistance System for Autonomous Vehicles. *Applied Sciences (Switzerland)*, 15(1). <https://doi.org/10.3390/app15010419>

Liu, D., Huang, W., Chu, R., Fan, Y., Fu, W., Tang, X., Li, Z., Jin, X., Zhang, H., & Wang, Y. (2025). A Comprehensive Performance Evaluation Method Based on Dynamic Weight Analytic Hierarchy Process for In-Loop Automatic Emergency Braking System in Intelligent Connected Vehicles. *Machines*, 13(6). <https://doi.org/10.3390/machines13060458>

Losada, Á., Páez, F. J., Luque, F., & Piovano, L. (2023). Effectiveness of the Autonomous Braking and Evasive Steering System OPREVU-AES in Simulated Vehicle-to-Pedestrian Collisions. *Vehicles*, 5(4), 1553–1569. <https://doi.org/10.3390/vehicles5040084>

Manolis, D., Spiliopoulou, A., Vandorou, F., & Papageorgiou, M. (2020). Real time adaptive cruise control strategy for motorways. *Transportation Research Part C: Emerging Technologies*, 115. <https://doi.org/10.1016/j.trc.2020.102617>

Masello, L., Castignani, G., Sheehan, B., Murphy, F., & McDonnell, K. (2022). On the road safety benefits of advanced driver assistance systems in different driving contexts. *Transportation Research Interdisciplinary Perspectives*, 15. <https://doi.org/10.1016/j.trip.2022.100670>

Muzammel, M., Yusoff, M. Z., Saad, M. N. M., Sheikh, F., & Awais, M. A. (2022). Blind-Spot Collision Detection System for Commercial Vehicles Using Multi Deep CNN Architecture. *Sensors*, 22(16). <https://doi.org/10.3390/s22166088>

Na, H., Kim, D. G., Kang, J. M., & Lee, C. (2025). The Development of a Lane Identification and Assessment Framework for Maintenance Using AI Technology †. *Applied Sciences (Switzerland)*, 15(13). <https://doi.org/10.3390/app15137410>

Neumann, T. (2024). Analysis of Advanced Driver-Assistance Systems for Safe and Comfortable Driving of Motor Vehicles. *Sensors*, 24(19). <https://doi.org/10.3390/s24196223>

Salud, S. de. (2021). *INFORME SOBRE LA SITUACIÓN DE LA SEGURIDAD VIAL MÉXICO*. <https://www.gob.mx/salud/acciones-y-programas/informes-sobre-la-situacion-de-la-seguridad-vial>

Shaout, A., & Castaneda-Trejo, L. (2025). Fuzzy Logic Control for Adaptive Braking Systems in Proximity Sensor Applications. *Electronics (Switzerland)*, 14(14). <https://doi.org/10.3390/electronics14142858>

Shrestha, A., & Mahmood, A. (2019). Review of deep learning algorithms and architectures. *IEEE Access*, 7, 53040–53065. <https://doi.org/10.1109/ACCESS.2019.2912200>

Tan, H., Zhao, F., Hao, H., Liu, Z., Amer, A. A., & Babiker, H. (2020a). Automatic emergency braking (AEB) system impact on fatality and injury reduction in China. *International Journal of Environmental Research and Public Health*, 17(3). <https://doi.org/10.3390/ijerph17030917>

Tan, H., Zhao, F., Hao, H., Liu, Z., Amer, A. A., & Babiker, H. (2020b). Automatic emergency braking (AEB) system impact on fatality and injury reduction in China. *International Journal of Environmental Research and Public Health*, 17(3). <https://doi.org/10.3390/ijerph17030917>

Taye, M. M. (2023). Theoretical Understanding of Convolutional Neural Network: Concepts, Architectures, Applications, Future Directions. In *Computation* (Vol. 11, Issue 3). MDPI. <https://doi.org/10.3390/computation11030052>

UAQ. (2024). *Vehículo Eléctrico*. <https://ingenieria.uaq.mx/index.php/oferta-educativa/licenciaturas/ingenieria-electromecanica>

Wang, Z., Huang, X., & Hu, Z. (2025). Attention-Based LiDAR–Camera Fusion for 3D Object Detection in Autonomous Driving. *World Electric Vehicle Journal*, 16(6). <https://doi.org/10.3390/wevj16060306>

Wei, W., Li, L., Li, Q., Zhang, S., Fan, C., & Liang, L. (2025). Dynamic Weight Model Predictive Control for Longitudinal Adaptive Cruise Systems in Electric Vehicles. *Applied Sciences (Switzerland)*, 15(12). <https://doi.org/10.3390/app15126715>

World Health Organization, W. (2023). *Global Status report on Road Safety*.

Xie, T., Yin, M., Zhu, X., Sun, J., Meng, C., & Bei, S. (2023). A Fast and Robust Lane Detection via Online Re-Parameterization and Hybrid Attention. *Sensors*, 23(19). <https://doi.org/10.3390/s23198285>

Yang, L., Yang, Y., Wu, G., Zhao, X., Fang, S., Liao, X., Wang, R., & Zhang, M. (2022). A Systematic Review of Autonomous Emergency Braking System: Impact Factor, Technology, and Performance Evaluation. In *Journal of Advanced Transportation* (Vol. 2022). Hindawi Limited. <https://doi.org/10.1155/2022/1188089>

Yang, Z., Wang, Z., & Yan, M. (2021). An optimization design of adaptive cruise control system based on mpc and adrc. *Actuators*, 10(6). <https://doi.org/10.3390/act10060110>

Yu, T., Tang, Y., Chen, R., & Zhao, S. (2025). Optimization of Adaptive Cruise Control Strategies Based on the Responsibility-Sensitive Safety Model. *Vehicles*, 7(2). <https://doi.org/10.3390/vehicles7020028>

Zhao, Y., Bai, L., Lyu, Y., & Huang, X. (2019). Camera-based blind spot detection with a general purpose lightweight neural network. *Electronics (Switzerland)*, 8(2). <https://doi.org/10.3390/electronics8020233>

Ziebinski, A., Cupek, R., Grzechca, D., & Chruszczyk, L. (2017). Review of advanced driver assistance systems (ADAS). *AIP Conference Proceedings*, 1906. <https://doi.org/10.1063/1.5012394>