Ing. Maria Laura Miranda Leon

Detección de somnolencia enfocado a conductores de automóviles con procesamiento de imágenes y CNN.

2025

Universidad Autónoma de Querétaro

Facultad de Ingeniería

Detección de somnolencia enfocado a conductores de automóviles con procesamiento de imágenes y CNN.

Tesis

Que como parte de los requisitos para obtener el Grado de

Maestra en Ciencias en Inteligencia Artificial

Presenta

Ing. Maria Laura Miranda Leon

Dirigido por:
Dr. Jesús Carlos Pedraza Ortega

Querétaro, Qro. a Mayo de 2025

Universidad Autónoma de Querétaro

Facultad de Ingeniería

Maestría en Ciencias en Inteligencia Artificial

Detección de somnolencia enfocado a conductores de automóviles con procesamiento de imágenes y CNN.

Tesis

Que como parte de los requisitos para obtener el Grado de

Maestra en Ciencias en Inteligencia Artificial

Presenta

Ing. Maria Laura Miranda Leon

Dirigido por:

Dr. Jesús Carlos Pedraza Ortega

Dr. Jesús Carlos Pedraza Ortega
Presidente
Dr. Juan Manuel Ramos Arreguin
Secretario
Dr. Marco Antonio Aceves Fernández
Vocal
Dr. Saúl Tovar Arriaga
Suplente
MCC. Luis Antonio Salazar Licea
Suplente

Centro Universitario, Querétaro, Qro. México
Junio, 2025

DEDICATORIAS

A mi esposo,

Mi compañero de aventuras, de retos y emprendimientos. Gracias por ser más que mi complemento, por ser mi detonante, ese impulso que me reta y me desafía, pero también mi refugio cuando la tormenta se aproxima. Contigo comparto risas, victorias y rivalidades que nos hacen fuertes, porque en nuestra complicidad encontramos el equilibrio perfecto entre el amor y el desafío constante. Eres mi hogar, mi adversario en el mejor sentido, y mi mejor aliado en este viaje.

A mis mascotas,

Quizá ustedes no comprendan con palabras cuánto los amo, pero en cada caricia, en cada mirada y en cada momento juntos, llevan un pedazo de mi corazón. Ustedes son y siempre serán mi mayor motor en esta vida. Sin su amor incondicional, mi mundo se sentiría vacío y sin rumbo. Si hoy estoy logrando mis sueños, es porque me acompañan en cada paso, dándome la fuerza para seguir adelante. Gracias por ser mi refugio, mi alegría y mi razón para nunca rendirme. A ti, mi Artemis, mi gordita preciosa, siempre tan intrépida y amorosa. A ti, mi Haru, mi güerito corajudo, con tu espíritu indomable. Y a ti, mi Putin, mi nerviosito prestish, con tu ternura inigualable. Pucca, Mion, Negra peluda, Momo, ustedes son quienes a pesar de sus tiernos 10 añitos viven conmigo desde hace 2 años que me case con su papa y me han dado también muchas alegrías. Y a ustedes Glucarda y Adolfo Domínguez fueron los de pilón pero espero que estén conmigo muchos años más. A ustedes 9 los amo como si fueran mis hijos.

Los amo con todo mi ser, hoy y siempre.

A mi padre y hermano,

En esta vida tengo cuatro pilares fundamentales, y dos de ellos son ustedes. Papá, sin tu amor y apoyo incondicional, no estaría donde estoy hoy. Gracias por ser mi guía, por impulsarme a terminar mis estudios, por asegurarte de que nunca me faltara nada y por enseñarme que con esfuerzo y perseverancia puedo alcanzar cualquier meta. Todo lo que he logrado tiene una parte de ti, y por eso te estaré eternamente agradecida.

Y a ti, mi hermanito lindo. Mi gordis, gracias, por soportar mis días difíciles y por hacerme reír con cada una de tus ocurrencias, incluso las más absurdas. Gracias por crecer a mi lado, por recordarme siempre que no estoy sola y por ser ese alguien en quien sé que siempre puedo confiar y con quien puedo actuar de manera inmadura sin ser juzgada. No importa qué pase, siempre estaré aquí para ti, así como sé que tú estarás para mí.

Los amo, con todo mi corazón, mis dos grandes hombres.

A mi madre y mi abuela,

Ustedes son los otros dos pilares fundamentales en mi vida. Las mujeres que han moldeado mi corazón y mi espíritu. Mamá, gracias por enseñarme todo lo que necesito saber para enfrentar el mundo. De ti aprendí a ser fuerte, a no dejarme doblegar ante nada ni nadie. Gracias a ti, soy una mujer que no teme luchar por lo que quiere. Porque me diste las herramientas para enfrentar cualquier obstáculo con valentía y determinación. Tu amor y tu ejemplo son mi mayor herencia.

Y a ti, mi querida Manaty, gracias por cada abrazo lleno de amor, por cada platillo preparado con cariño, por ser mi refugio y mi segunda madre. Contigo siempre puedo ser esa niña mimada por su abuela, sintiéndome protegida y amada en cada etapa de mi vida. Tu amor incondicional es un tesoro que llevo en el alma.

Las amo, con todo mi corazón, hoy y siempre.

AGRADECIMIENTOS

Expreso mi más sincero agradecimiento al Consejo Nacional de Humanidades, Ciencias y Tecnologías (CONAHCYT) por el apoyo brindado a través de la beca otorgada, la cual fue fundamental para la realización de este trabajo.

A mis sinodales, por su tiempo, guía y valiosas observaciones, que contribuyeron significativamente al desarrollo de esta investigación. Su conocimiento y retroalimentación fueron clave para mejorar y fortalecer este proyecto.

A la Universidad Autónoma de Querétaro, por proporcionarme la formación académica y los recursos necesarios para llevar a cabo esta investigación. Su compromiso con el desarrollo del conocimiento y la ciencia ha sido una fuente constante de inspiración.

A todas las personas que contribuyeron en la creación de la base de datos utilizada en este trabajo, por su disposición, paciencia y colaboración. Sin su apoyo, este proyecto no habría sido posible.

A todos aquellos que, de una u otra manera, han formado parte de este camino, mi más profundo agradecimiento.

RESUMEN

La seguridad vial es una de las principales preocupaciones a nivel global. Debido al alto número de accidentes de tráfico, los cuales representan una causa significativa de lesiones y muertes. De acuerdo con el Anuario Estadístico de Colisiones en Carreteras Federales en México, los accidentes relacionados con la fatiga y la somnolencia presentan una tendencia fluctuante. Por lo que la presente investigación aborda el desafío de la detección de somnolencia enfocada en conductores de automóviles.

Esta investigación se centra en la detección de somnolencia en conductores mediante el uso de modelos de inteligencia artificial y procesamiento de imágenes. Para ello, el estudio se estructuró en tres etapas principales:

1. Evaluación preliminar y experimentación con datasets públicos

En esta fase inicial, se analizaron bases de datos públicas mediante herramientas de detección de objetos como Liner.ai, lo que permitió una exploración inicial del rendimiento y la calidad de los datos. A partir de esta evaluación, se identificó la necesidad de construir un dataset acorde a las necesidades del proyecto. Incorporando técnicas de procesamiento de imágenes para enriquecer su calidad y diversidad.

2. Evaluación de modelos tradicionales de detección de objetos

Se implementó y analizó el rendimiento del modelo Faster R-CNN, considerado un referente por su velocidad y precisión. Esta etapa permitió establecer una línea base para comparar el desempeño de modelos más recientes. Además, se descartaron aquellos experimentos con resultados poco prometedores y se seleccionaron los modelos con mejor desempeño para la siguiente fase.

3. Comparación de modelos YOLOv8 y YOLOv10

En la etapa final, se implementaron y compararon los modelos YOLOv8 y YOLOv10, esto con el objetivo de determinar cuál ofrece un mejor desempeño para la detección de somnolencia en conductores. Los resultados mostraron una precisión de

hasta el 99.3%, lo que demuestra el potencial de estos modelos para futuras aplicaciones en seguridad vial.

ABSTRACT

Road safety is one of the main global concerns due to the high number of traffic accidents, which represent a significant cause of injuries and deaths. According to the Statistical Yearbook of Collisions on Federal Highways in Mexico, accidents related to fatigue and drowsiness show a fluctuating trend, so this research addresses the challenge of drowsiness detection focused on automobile drivers.

This research focuses on detecting drowsiness in drivers through the use of artificial intelligence models and image processing. To this end, the study was structured in three main stages:

1. Preliminary evaluation and experimentation with public datasets

In this initial phase, public databases were analyzed using object detection tools such as Liner.ai, allowing for an initial exploration of data performance and quality. Based on this assessment, the need to build a dataset tailored to the project's needs was identified, incorporating image processing techniques to enhance its quality and diversity.

2. Evaluation of Traditional Object Detection Models

The performance of the Faster R-CNN model, considered a benchmark for its speed and accuracy, was implemented and analyzed. This stage established a baseline for comparing the performance of more recent models. Furthermore, experiments with unpromising results were discarded, and the best-performing models were selected for the next phase.

3. Comparison of YOLOv8 and YOLOv10 Models

In the final stage, the YOLOv8 and YOLOv10 models were implemented and compared to determine which offers better performance for detecting drowsiness in

drivers. The results showed an accuracy of up to 99.3%, demonstrating the potential of these models for future applications in road safety.

ABREVIATURAS Y SIGLAS

Acrónimo	Español	English
AI	Inteligencia Artificial	Artificial Intelligence
CNN	Redes Neuronales Convolucionales	Convolutional Neural Networks
ADAS	Sistemas Avanzados de Asistencia al Conductor	Advanced Driver Assistance Systems
EEG	Electroencefalograma	Electroencephalography
EOG	Electrooculograma	Electrooculography
ECG	Electrocardiograma	Electrocardiogram
EMG	Electromiograma	Electromyography
R-CNN	Red Neuronal Convolucional Basada en Regiones	Region-based Convolutional Neural Network
YOLO	Solo Miras Una Vez	You Only Look Once
YOLOv	-	You Only Look Once version
ResNet	-	Residual Networks
FPN	-	Feature Pyramid Networks
Faster R-CNN	Red Neuronal Convolucional Basada en Regiones Acelerada	Faster Region-based Convolutional Neural Network
SSD	Detección en un Solo Disparo	Single Shot Detection
mAP	Precisión Media Promediada	Mean Average Precision
IoU	Intersección sobre Unión	Intersection over Union
VOC	Clases de Objetos Visuales (refiriéndose al formato Pascal VOC)	Visual Object Classes (referring to Pascal VOC format)
TP	Verdadero Positivo	True Positive
FP	Falso Positivo	False Positive
FN	Falso Negativo	False Negative

ROC	Curva Característica Operativa del Receptor	Receiver Operating Characteristic
AUC	Área Bajo la Curva	Area Under the Curve
FPS	Cuadros por Segundo	Frames Per Second
RGB	Rojo, Verde, Azul	Red, Green, Blue
ROI	Región de Interés	Region of Interest
DNN	Red Neuronal Profunda	Deep Neural Network
ML	Aprendizaje Automático	Machine Learning
DL	Aprendizaje Profundo	Deep Learning

ÍNDICE GENERAL

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

I. INTRODUCCIÓN

1.1 Somnolencia

De acuerdo con numerosos estudios realizados por expertos en el campo de la medicina y profesionales clínicos del sueño, la somnolencia es considerada una necesidad fisiológica fundamental para el ser humano (Carrillo-Mora et al., 2013; Rosales Mayor & Rey De Castro Mujica, 2010). Esta necesidad es comparable, en importancia, a otras funciones biológicas esenciales como la alimentación y la hidratación, lo cual la hace crucial para el adecuado funcionamiento del organismo. Sin la satisfacción de esta necesidad, el ser humano no podría mantener su salud ni su capacidad para realizar actividades diarias.

Dentro del estudio de la somnolencia se reconocen dos tipos: la objetiva y la subjetiva(Rosales Mayor & Rey De Castro Mujica, 2010).

1.2 Somnolencia Objetiva y Subjetiva

La somnolencia objetiva se refiere a la tendencia fisiológica de una persona a quedarse dormida. Esta condición se evalúa mediante mediciones estandarizadas como el Test de Latencia Múltiple del Sueño (TLMS), el cual determina cuánto tiempo tarda una persona en conciliar el sueño en diferentes momentos del día(Instituto Peruano de Neurociencias (IPN), 2017). Este tipo de somnolencia es un fenómeno medible, basado en respuestas fisiológicas observables y cuantificables.

Por otro lado, según el consenso de investigadores y clínicos en el área, la somnolencia subjetiva es la percepción personal de tener sueño o de encontrarse en un estado de transición entre estar despierto y dormido (Shen et al., 2006). Esta forma de somnolencia se asocia con sensaciones como:

- Bostezo
- Cierre involuntario de los ojos
- Pérdida de tono muscular en el cuello
- Menor atención
- Disminución del rendimiento cognitivo y psicomotor

1.3 Privación del sueño

La privación del sueño puede ser causada por diversos factores, que van desde los más simples hasta los más complejos. Entre las causas más comunes se encuentran la falta de sueño debido a hábitos poco saludables, el estrés o los compromisos laborales. No obstante, también existen factores más complejos que afectan la calidad del sueño, como los trastornos del sueño (por ejemplo, apnea del sueño e insomnio).

Cuando la privación del sueño se prolonga, puede surgir disritmia circadiana, un trastorno que ocurre cuando el ciclo natural de sueño-vigilia se ve alterado debido a cambios abruptos en los hábitos de sueño. Este desfase afecta la capacidad del cuerpo para sincronizarse con el entorno y altera funciones cognitivas como la atención, la memoria y el tiempo de reacción (Ortiz et al., 2007). Todo esto puede disminuir el rendimiento intelectual y aumentar la probabilidad de cometer errores que podrían derivar en accidentes, desde leves hasta mortales.

Algunos estudios realizados en 1999 respaldan esta afirmación: en dicho estudio se encontró una relación significativa entre la somnolencia y los accidentes de tráfico, al utilizar el Índice de Apnea-Hipopnea (AHI) para medir la gravedad de la somnolencia (Terán-Santos et al., 1999). Se demostró que, en la mayoría de los casos, los conductores reportaron sentirse cansados y somnolientos antes de sufrir un accidente vehicular.

1.4 Somnolencia vehicular

Desde siempre, el ser humano ha buscado formas de ahorrar tiempo al trasladar materiales o pasajeros de un punto a otro, ya sea para trabajar, ir a la escuela o visitar algún lugar. Antes de los automóviles motorizados, existían carros tirados por animales robustos o incluso por personas. No fue sino hasta la Revolución Industrial que comenzaron las innovaciones en este ámbito, primero con vehículos a vapor, luego con motores de combustión interna, y más recientemente con motores eléctricos (Lucendo, 2019).

Si bien la llegada del automóvil ha facilitado muchas actividades humanas, no es beneficiosa cuando las condiciones del conductor no son óptimas. Por ello, la detección de

somnolencia en conductores se ha vuelto un campo de estudio cada vez más relevante, dada la importancia de prevenir accidentes de tráfico relacionados con la fatiga o somnolencia.

La somnolencia vehicular se refiere al estado de cansancio extremo o somnolencia que experimenta un conductor mientras maneja. Los síntomas más comunes coinciden con los observados en la somnolencia subjetiva: bostezos frecuentes y ptosis por fatiga lo cual dificulta enfocar la vista en el camino y, por tanto, impide mantener la atención en la carretera (Fernández Carrión, 2021). Esto puede provocar movimientos bruscos del vehículo y aumentar el riesgo de accidentes.

1.5 Planteamiento del problema

La somnolencia vehicular constituye un riesgo significativo para la seguridad vial, ya que afecta la capacidad de atención, reacción y toma de decisiones de los conductores, aumentando la probabilidad de accidentes de tráfico(Carrillo-Mora et al., 2013; Fernández Carrión, 2021; Ortiz et al., 2007). A pesar de los avances en la seguridad automotriz, los accidentes relacionados con la fatiga continúan siendo una de las principales causas de siniestros viales en México.

Los síntomas asociados a la somnolencia, como los bostezos frecuentes y el cierre prolongado de los ojos, disminuyen la concentración y afectan directamente la habilidad de los conductores para mantener el control del vehículo. Esto puede derivar en accidentes graves o incluso fatales.

Aunque se ha documentado la prevalencia de la somnolencia como factor contribuyente en accidentes, su detección temprana y precisa sigue representando un reto. Los sistemas tradicionales de detección de somnolencia suelen requerir la interacción directa con el conductor o la instalación de tecnologías avanzadas en los vehículos. Lo cual resulta en soluciones costosas y poco prácticas para una implementación generalizada.

Si bien algunos modelos de detección han mostrado un alto rendimiento, la mayoría de ellos dependen de técnicas específicas. Por ello, este trabajo busca abordar el problema mediante enfoques innovadores que integren el procesamiento de imágenes y la inteligencia artificial. Este enfoque no solo pretende mejorar la precisión de la detección, sino también

simplificar el proceso y hacerlo más accesible para su replicación en futuros proyectos de seguridad vial.

1.6 Justificación

La seguridad vial constituye una de las preocupaciones más apremiantes a nivel nacional y global, ya que los accidentes de tráfico siguen siendo una de las principales causas de lesiones y muertes prematuras. Un factor determinante en la ocurrencia de estos accidentes es la somnolencia o fatiga del conductor, la cual puede comprometer gravemente su capacidad de reacción, atención y juicio durante la conducción.

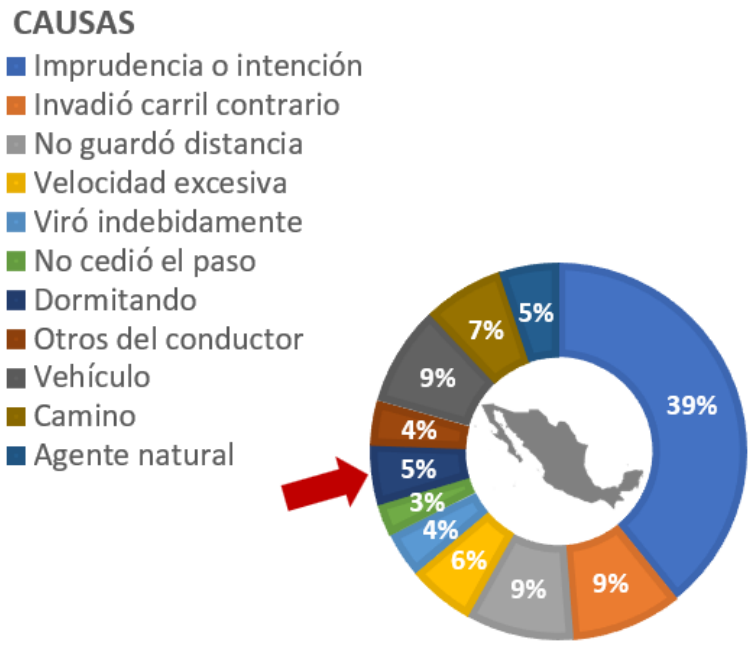


Figura 1. Principales causas de accidentes viales en México (Cuevas-Colunga & Silva Rivera, 2024).

Según datos oficiales, en 2021 se registraron 19,889 accidentes de tráfico, de los cuales el 3.76 % fueron causados por conductores que se encontraban dormitando (Cuevas-Colunga et al., 2022). Para el año 2022, aunque se reportaron 12,618 accidentes (una disminución en términos absolutos) el 3.62 % de ellos también fueron provocados por conductores somnolientos, lo que representa únicamente una reducción del 0.14 % en los accidentes atribuibles a la somnolencia. Sin embargo, en 2023, a pesar de registrarse 3,977 accidentes

menos, el porcentaje de accidentes causados por somnolencia aumentó en un 1.48 % (ver Figura 1) (Cuevas-Colunga & Silva Rivera, 2024).

Este comportamiento sugiere una tendencia irregular en la incidencia de accidentes relacionados con la fatiga, a pesar de predicciones optimistas que indicaban una reducción del 10 % en los siniestros vehiculares con la implementación de sistemas avanzados de asistencia al conductor (ADAS) para 2025 (EQUIPO SITEC, 2023).

Considerando lo anterior, es evidente que la fatiga y la falta de sueño adecuado pueden tener efectos devastadores sobre las capacidades cognitivas de los conductores. Tales como:

- Tiempos de reacción más lentos
- Deterioro en la atención y la percepción
- Pérdida temporal de la conciencia

Lo que incrementa el riesgo de accidentes graves y fatales. Estos problemas se agravan aún más por la naturaleza impredecible de la somnolencia, lo que hace más difícil su prevención.

Frente a este desafío, resulta crucial el desarrollo e implementación de sistemas eficaces para la detección en de somnolencia en los conductores. De este modo permitiendo intervenir antes de que ocurra un accidente. En este contexto, la inteligencia artificial y, en particular, las redes neuronales artificiales, emergen como una opción prometedora, ya que han demostrado ser herramientas eficientes en el análisis y detección de patrones complejos en imágenes y videos.

La implementación de estas tecnologías permitiría no solo una mejor comprensión de los patrones asociados con la somnolencia al volante, sino también dar un paso adelante para la creación de sistemas de alerta temprana alternos que puedan prevenir accidentes en tiempo real. De este modo, se espera que este proyecto aporte una contribución significativa al avance tecnológico en el campo de la seguridad vial, con el objetivo de reducir la incidencia de accidentes causados por la fatiga del conductor.

II. ANTECEDENTES

En la actualidad, existen diversas técnicas para la detección de somnolencia, las cuales pueden clasificarse en tres grupos principales:

1. Aquellas que utilizan el comportamiento del vehículo como modelo predictivo.
2. Aquellas que se basan en señales biológicas del cuerpo humano.
3. Aquellas que emplean señales visuales con ayuda de procesamiento de imágenes.

2.1 Detección de somnolencia por comportamiento del vehículo

El primer grupo corresponde a los Sistemas Avanzados de Asistencia al Conductor, comúnmente conocidos como sistemas ADAS, los cuales utilizan una combinación de sensores para recopilar información sobre el comportamiento del vehículo, como el giro del volante y la desviación del carril (Muñoz et al., 2024).

La somnolencia afecta directamente la forma de conducir, haciendo que el comportamiento del vehículo se vuelva irregular y peligroso. Por ejemplo, un conductor somnoliento tiende a reducir la frecuencia de los movimientos correctivos del volante, lo que incrementa la probabilidad de giros bruscos y desvíos del carril, aumentando el riesgo de accidentes.

Este tipo de cambios o irregularidades en el volante se comunican a través del sistema CAN BUS, el cual permite obtener medidas como el ángulo y la velocidad de giro. Estos datos, junto con el seguimiento de los bordes del carril mediante cámaras integradas y algoritmos de visión por computadora, permiten detectar patrones asociados a la somnolencia (García Daza, 2011).

Con parámetros como la distancia entre el vehículo y las líneas del carril, el ángulo del volante, la velocidad de giro y el tiempo estimado para cruzar de carril, el sistema ADAS puede realizar una detección eficaz de estados de fatiga. No obstante, estos sistemas solo están presentes en vehículos recientes y suelen implicar un costo elevado.

2.2 Detección de somnolencia por señales biológicas

El segundo grupo corresponde a la detección por señales biológicas, también llamadas bioseñales (Anrango Farinango, 2022). Estas señales pueden ser eléctricas o no eléctricas y se obtienen mediante sensores o transductores que detectan cambios en el entorno biológico.

Este enfoque permite medir de forma objetiva el grado de somnolencia, utilizando técnicas como (Shen et al., 2006):

- Electroencefalograma (EEG): Registra la actividad eléctrica cerebral. Las ondas se vuelven lentas en el sueño profundo y más rápidas en el sueño REM.
- Electrooculograma (EOG): Registra el movimiento ocular, que también varía según la fase del sueño.
- Electrocardiograma (ECG): Registra la actividad eléctrica del corazón. Las variaciones en los patrones cardiacos pueden indicar somnolencia.
- Electromiograma (EMG): Mide la actividad muscular. Durante la somnolencia, los músculos tienden a relajarse considerablemente.

La mayoría de estas técnicas son invasivas o requieren contacto directo con el usuario, mediante electrodos conectados a sensores especializados. Por ello, son más comunes en el estudio clínico del sueño que en aplicaciones móviles o vehiculares.

Sin embargo, el nivel de exactitud en la detección del sueño mediante redes neuronales tradicionales como Decision Trees (DT), Random Forest (RF), K-Neighbors y GaussianNB, por mencionar algunas, suele oscilar entre un 60 % y un 73% de precisión. Un trabajo destacado de 2021 logró mejorar estos resultados mediante el uso del modelo EEGNet, específicamente diseñado para análisis de electroencefalografía. Este modelo fue comparado con algoritmos como SVM y QDA, cada uno entrenado durante seis épocas, mostrando un rendimiento superior en términos de precisión y eficiencia para detectar somnolencia (Cui et al., 2022).

2.3 Detección de somnolencia por señales visuales

El tercer grupo contempla la detección mediante señales visuales (Cluydts et al., 2002). Existen cambios en la apariencia y el comportamiento de una persona somnolienta que pueden observarse visualmente, como:

- Cierre parcial o total de los ojos
- Cambios en la expresión facial
- Modificación de la postura corporal
- Respuesta lenta a estímulos
- Desconexión visual o disociación

La transición de un estado activo a uno somnoliento suele ser progresiva y perceptible. La persona puede adoptar posturas más relajadas, aflojar los músculos faciales y cerrar los ojos de forma característica (Carrillo-Mora et al., 2013; Ortiz et al., 2007; Rosales Mayor & Rey De Castro Mujica, 2010), lo cual facilita la detección visual de somnolencia subjetiva.

2.4 Detección de somnolencia usando inteligencia artificial con señales visuales

Actualmente, existen diversos modelos de detección de somnolencia basados en la identificación de objetos mediante redes neuronales artificiales, como YOLO, Mask R-CNN y SSD. Estos enfoques aprovechan el análisis de señales visuales mediante algoritmos de visión por computadora, combinados con técnicas de aprendizaje profundo.

Uno de los primeros desarrollos relevantes fue realizado en la Universidad Autónoma de Querétaro, donde se utilizó un clasificador en cascada para localizar el rostro del conductor. Posteriormente, se analizó un conjunto de puntos característicos faciales (landmarks) con el objetivo de identificar regiones clave como los ojos y la boca. El sistema fue probado en sujetos con características faciales diversas y se evaluó su rendimiento bajo diferentes resoluciones (Alcántara-Montiel et al., 2019; Montiel-Alcántara, 2018). Se observó una relación inversa entre la resolución y la velocidad de procesamiento: a menor resolución, el sistema era más rápido, pero con pérdida de precisión en la interpretación de los landmarks.

En otro estudio, se utilizaron redes neuronales denominadas Robust SleepNets para detectar somnolencia mediante la identificación del rostro y el análisis del parpadeo. Este

modelo fue entrenado exclusivamente con imágenes de ojos abiertos y cerrados, alcanzando métricas destacables: una precisión promedio del 94 %, un recall del 95 % y un puntaje F1 de 94.25 % (Alparslan & Kim, 2021). No obstante, según investigaciones posteriores se encontró que, a pesar del buen desempeño en métricas cuantitativas, el modelo enfrentaba dificultades para localizar con precisión los ojos y la boca en tiempo real(Singh et al., 2023).

Adicionalmente, se propuso un sistema de detección de somnolencia basado en la arquitectura VGG combinada con una red neuronal convolucional (CNN) preentrenada específicamente para el análisis de rasgos faciales humanos. Este modelo logró una precisión del 98.02 %(Kepesiova et al., 2020) , aunque presentó limitaciones al momento de reconocer nuevos rostros. Esto sugiere que el conjunto de datos utilizado era insuficiente en términos de diversidad y que no se aplicaron técnicas de aumentación de imágenes, factores esenciales para mejorar la capacidad de generalización del modelo.

El trabajo más reciente en este campo corresponde a un experimento realizado en 2023, que empleó una cámara infrarroja para detectar somnolencia en condiciones nocturnas. El modelo utilizado, basado en la arquitectura YOLOv4, alcanzó una precisión del 88 % y un recall del 93 % (Ramírez-Arriaga, 2023). No obstante, su desempeño quedó limitado por el hecho de que fue entrenado únicamente con imágenes nocturnas, lo cual reduce su aplicabilidad en contextos diurnos o mixtos.

III. FUNDAMENTACIÓN TEÓRICA

La tecnología, a lo largo de la historia, ha sido un motor clave en el desarrollo de herramientas y sistemas que transforman la forma de interacción de los seres humanos hacia su entorno. Muchos avances emblemáticos tienen sus raíces en épocas de conflicto. Tales como los vehículos automotores o el célebre algoritmo de Turing (Benítez Rojas, 2023; Lucendo, 2019),. El cual fue desarrollado durante la Segunda Guerra Mundial. Sin embargo, estos inventos no se limitaron a aplicaciones militares, con el tiempo, se adaptaron y evolucionaron para facilitar múltiples aspectos de la vida cotidiana, contribuyendo significativamente al progreso de la sociedad.

En las últimas décadas, hemos sido testigos de un crecimiento exponencial en el desarrollo tecnológico, lo que ha llevado a la creación de dispositivos cada vez más “exactos” y, en muchos casos, calificados como "inteligentes". Pero ¿qué significa realmente que un aparato sea inteligente? Detrás de esta "inteligencia" se encuentra la combinación y aplicación de tecnologías como los algoritmos de inteligencia artificial, la visión por computadora y el procesamiento de imágenes.

3.1 Procesamiento de imágenes

Dentro del campo de la visión por computadora, exístete una disciplina fundamental que es el procesamiento de imágenes. Esta área se ocupa del análisis y manipulación de datos visuales con el fin de extraer información relevante o transformarlos para un propósito específico (Wiley & Lucas, 2018). Se ha consolidado como una herramienta clave en el avance de múltiples sectores tecnológicos como: la robótica, la salud, la interacción humano-computadora, la seguridad y la inteligencia artificial.

El procesamiento de imágenes no solo permite identificar patrones visuales, sino que también facilita el análisis de comportamientos, características o eventos en entornos dinámicos (Satapathy et al., 2020). Por ejemplo, puede detectar señales de fatiga como el bostezo o el estado de los ojos (con ptosis o sin ptosis), esto incluso en condiciones de baja iluminación o con presencia de sombras.

Este tipo de análisis resulta clave para el desarrollo de sistemas inteligentes que mejoren la interacción humano-máquina, desde aplicaciones médicas hasta monitoreo vehicular en tiempo real.

3.2 Inteligencia Artificial

Aunque el procesamiento de imágenes es una herramienta poderosa para analizar datos visuales, por sí solo tiene limitaciones. Sus principales funciones incluyen: resaltar bordes, ajustar brillo, contraste, color y operar en distintos espacios de color. Estas capacidades, si bien útiles, funcionan como apoyo para tareas más complejas.

Para reconocer y comprender patrones de forma autónoma, es necesaria la integración con la inteligencia artificial (IA). En los últimos años, la IA ha adquirido gran relevancia debido

a su capacidad para resolver tareas complejas: generación de texto, predicciones, clasificación de imágenes y detección de objetos, entre otras (Angelov et al., 2021).

La sinergia entre procesamiento de imágenes e inteligencia artificial ha dado lugar a sistemas avanzados que van más allá de la manipulación visual: ahora pueden interpretar y tomar decisiones basadas en datos visuales, de manera autónoma y eficiente (Angelov et al., 2021). Esta combinación ha impulsado avances significativos en áreas como la salud, la robótica, la seguridad, el transporte y más.

3.3 Redes Neuronales Artificiales

En el campo de la inteligencia artificial, uno de los conceptos más destacados es el de las redes neuronales artificiales (Artificial Neural Networks, ANN por sus siglas en inglés). Su nombre proviene de la inspiración en la anatomía y funcionamiento de las neuronas humanas. En el sistema nervioso, las neuronas transmiten información entre el cerebro y el cuerpo mediante señales eléctricas. Este proceso inicia cuando una neurona recibe estímulos desde otras neuronas a través de sus dendritas. Una vez procesada la información, la neurona genera pequeños pulsos eléctricos llamados potenciales de acción. Estos viajan por una estructura llamada axón hasta llegar a la sinapsis, donde se comunican con otras neuronas (Gershenson, 2003).

Este mecanismo ha servido como modelo conceptual para diseñar las redes neuronales artificiales. En ellas, cada "neurona" artificial recibe una o varias entradas, que son ponderadas por coeficientes llamados pesos. Estas entradas ponderadas se combinan y pasan por una función de activación. Esta función determina si la neurona se activa o no, generando así una salida (Gershenson, 2003). Este proceso permite que las redes neuronales simulen procesos de aprendizaje, reconocimiento de patrones y toma de decisiones.

En términos más formales, el procesamiento que realiza una neurona artificial puede describirse mediante la siguiente operación matemática:

$$y = f(\sum(w_i \cdot x_i) + b) \qquad (1)$$

Donde:

- x_i representa las entradas,
- w_i los pesos asociados,
- b es el sesgo (bias),
- y f es la función de activación (por ejemplo, ReLU, sigmoide, tanh).

Esta operación permite que la red ajuste los pesos en función de los errores cometidos, mejorando su rendimiento con cada iteración (Goodfellow et al., 2016). El aprendizaje se logra a través de la retropropagación del error, combinado con un algoritmo de optimización como el descenso de gradiente (Gershenson, 2003).

La similitud estructural y funcional entre neuronas biológicas y artificiales ha sido representada gráficamente para facilitar su comprensión. En la Figura 2, puede observarse una neurona biológica y su contraparte artificial, destacando sus elementos equivalentes: dendritas → entradas, núcleo → función de activación, axón → salida.

Gracias a esta analogía, se ha permitido desarrollar sistemas computacionales que imitan funciones cognitivas humanas como la clasificación de información, predicción de eventos o reconocimiento de patrones. Es por ello que las redes neuronales artificiales se han convertido en uno de los pilares fundamentales de la inteligencia artificial moderna.

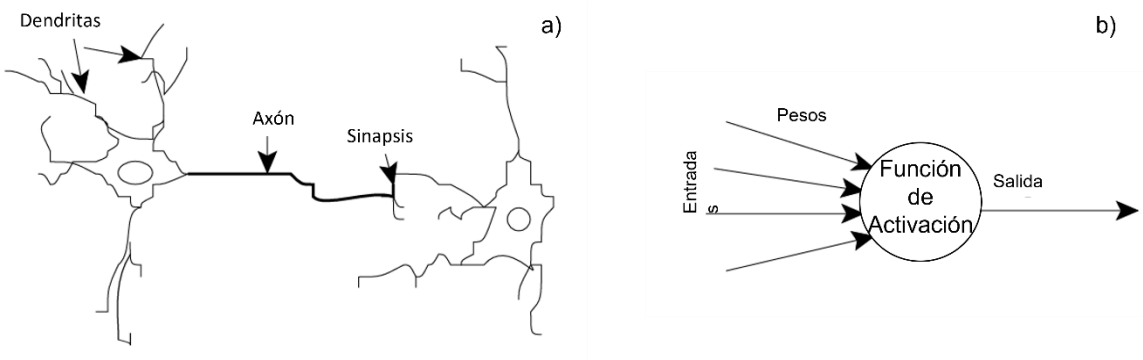


Figura 2. Concepto visual de una a) Neurona; b) Neurona Artificial. (Gershenson, 2003).

3.4 Redes Neuronales Convolucionales

Dentro del campo de las ANNs, las redes neuronales convolucionales (Convolutional Neural Networks, CNN por sus siglas en inglés) destacan por su capacidad para procesar datos en formato de imágenes o videos (Guo et al., 2017). Este tipo de redes es ampliamente utilizado en tareas de visión por computadora como el reconocimiento, detección y clasificación de objetos.

Su eficacia radica en la habilidad para identificar patrones visuales directamente en los datos de entrada. Estos suelen representarse como matrices de píxeles (Guo et al., 2017). Las CNN han demostrado ser especialmente útiles para detectar estructuras espaciales jerárquicas. Lo cual que las convierte en herramientas eficientes para el análisis de imágenes.

Una red convolucional típica está compuesta por tres tipos principales de capas: Cada una con funciones específicas que contribuyen al procesamiento y análisis de la información visual:

3.4.1 Capas de convolución

Las capas de convolución son las responsables de extraer las características visuales más relevantes de las imágenes de entrada. Su funcionamiento se basa en el uso de filtros o núcleos (kernels), que recorren la imagen en pasos definidos (strides) y detectan patrones específicos como bordes, texturas o formas (Guo et al., 2017).

Cada filtro se aplica sobre una pequeña región de la imagen. Este va multiplicando sus valores por los correspondientes valores de los píxeles y sumando los resultados para generar una nueva matriz conocida como mapa de características (feature map). Tras cada operación de convolución, se aplica una función de activación (como ReLU (Rectified Linear Unit), sigmoide o tanh) introduciendo no linealidad en el modelo, permitiéndole aprender patrones complejos (Guo et al., 2017).

El uso de múltiples filtros permite a la red detectar diferentes características a lo largo de toda la imagen, lo que enriquece la representación interna de los datos.

3.4.2 Capas pooling

Las capas pooling (o de submuestreo) tienen como función reducir la resolución espacial del feature map resultante de la convolución. Disminuyendo así la cantidad de parámetros y la complejidad computacional del modelo.

Este tipo de capas utiliza funciones de agregación como máximo (max-pooling) o promedio (average-pooling) para seleccionar la información más significativa de una región determinada del mapa de características (Guo et al., 2017). En términos simples, estas capas permiten destacar los patrones más importantes mientras se descartan detalles irrelevantes o ruido.

Además, el pooling contribuye a hacer que el modelo sea más robusto frente a pequeñas variaciones en la posición, escala o iluminación de los objetos en la imagen.

3.4.3 Capas totalmente conectadas

Las capas totalmente conectadas (fully connected layers) suelen ubicarse al final de la red neuronal y son las responsables de la toma de decisiones. Cada neurona de estas capas está conectada con todas las salidas de la capa anterior, formando una estructura densa.

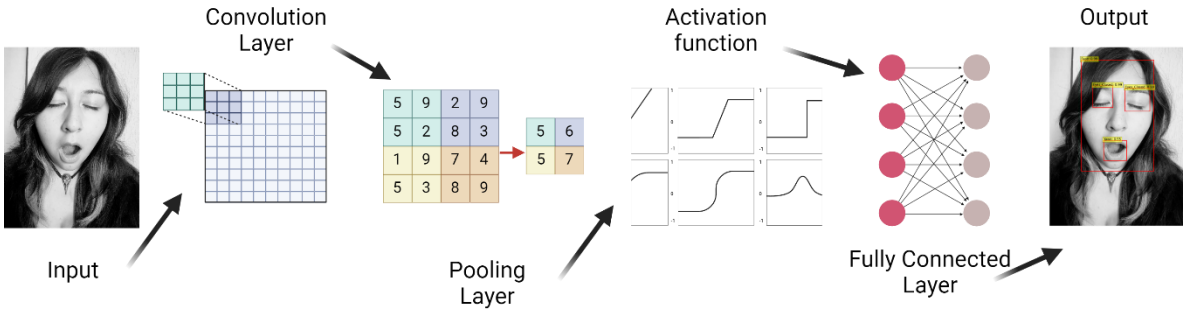


Figura 3. Arquitectura conceptual de una Red Neuronal Convolucional.

En esta etapa, se realiza una operación de suma ponderada y se aplica una función de activación, lo que permite clasificar o detectar objetos con base en las características extraídas por las capas anteriores (Guo et al., 2017). Este proceso permite que la red produzca salidas interpretables, como una etiqueta de clase o una coordenada de ubicación para objetos detectados.

Estas tres capas (convolucionales, de pooling y totalmente conectadas) conforman la estructura básica de las redes neuronales convolucionales y permiten el aprendizaje jerárquico a partir de imágenes. Gracias a esto, las CNN han demostrado ser altamente eficaces para resolver problemas de visión por computadora. Los ejemplos: reconocimiento facial, análisis médico, inspección industrial, entre muchos otros.

En el ámbito del aprendizaje automático aplicado a la visión, las CNN se han convertido en el núcleo de diversos modelos de detección de objetos. Algunas de las arquitecturas más representativas que integran redes convolucionales como su componente principal (también conocido como backbone) son:

- Faster R-CNN (2015): Pionera en incorporar regiones de interés para mejorar la precisión y eficiencia en la detección.
- YOLO (You Only Look Once): Uno de los enfoques más recientes y populares por su velocidad y capacidad de detección en tiempo real.

3.5 Arquitectura Faster R-CNN

El modelo Faster R-CNN con ResNet-50 y FPN es una arquitectura ampliamente utilizada en el ámbito de la detección de objetos. La cual se encuentra disponible en librerías de código abierto como PyTorch (PyTorch, s/f). Esta arquitectura combina técnicas avanzadas que mejoran la precisión y eficiencia en tareas de visión por computadora (Miranda-Leon et al., 2024).

Su estructura principal se compone de los siguientes módulos:

3.5.1 Backbone (ResNet50)

ResNet-50 es una red neuronal profunda. Se basa en bloques residuales, diseñados para facilitar el entrenamiento de modelos con muchas capas. Su principal innovación radica en las conexiones de atajo (skip connections) (Koonce, 2021). Las que permiten que el gradiente fluya sin problemas durante el proceso de retropropagación, evitando el problema de la degradación del rendimiento en redes profundas.

Esta arquitectura consta de 50 capas que actúan como un extractor de características. Proporcionando representaciones detalladas de las imágenes de entrada. Gracias a su diseño eficiente, ResNet-50 es una de las arquitecturas más utilizadas como base (backbone) en modelos de detección avanzados.

3.5.2 Feature Pyramid Network (FPN)

La Feature Pyramid Network (FPN) se integra con ResNet-50 para construir una pirámide de características multiescala: Este permite al modelo detectar objetos de distintos tamaños (Zhu et al., 2022). El enfoque combina características de diferentes niveles de la red convolucional, generando mapas de características enriquecidos que mejoran la detección, en especial de objetos pequeños.

En otras palabras la FPN mejora la capacidad de generalización del modelo, ya que le permite aprender de estructuras visuales tanto locales como globales.

3.5.3 Region Proposal Network (RPN):

La Region Proposal Network (RPN) es una subred que tiene como objetivo generar regiones candidatas donde probablemente se encuentren objetos. La RPN escanea todo el

mapa de características de la imagen y predice regiones de interés (Regions of Interest, RoI), mediante el uso de anclas de diferentes escalas y proporciones (Li, 2021).

Este proceso es rápido y eficiente, ya que reutiliza las características previamente extraídas por el backbone. Evitando así cálculos redundantes. Las regiones propuestas por la RPN son refinadas en etapas posteriores del modelo.

3.5.4 Region of Interest Pooling (RoI)

Las propuestas generadas por la RPN tienen diferentes dimensiones, por lo que deben uniformarse en tamaño antes de ser procesadas por las capas finales del modelo. Para ello, se utiliza la técnica de RoI Pooling, que convierte cada región en una matriz de tamaño fijo (Gholamalinezhad & Khosravi, 2020).

Este paso garantiza que todas las regiones candidatas puedan ser procesadas por las capas completamente conectadas, sin importar su forma o escala original (L. Zhao & Zhang, 2024; X. Zhao et al., 2024).

3.5.5 Bounding Box Regression (BBR):

Una vez que las regiones propuestas han sido ajustadas, se procesan a través de una red totalmente conectada que cumple dos funciones esenciales (Li, 2021; Miranda-Leon et al., 2024):

- Clasificación: Determina la clase del objeto contenido en cada región propuesta, utilizando funciones como softmax.
- Regresión de cajas delimitadoras: Ajusta las coordenadas de las cajas (o bounding boxes) para mejorar la precisión en la localización del objeto dentro de la imagen.

Este mecanismo de doble salida permite que Faster R-CNN no solo detecte objetos, sino que los clasifique correctamente y los delimite con gran exactitud .

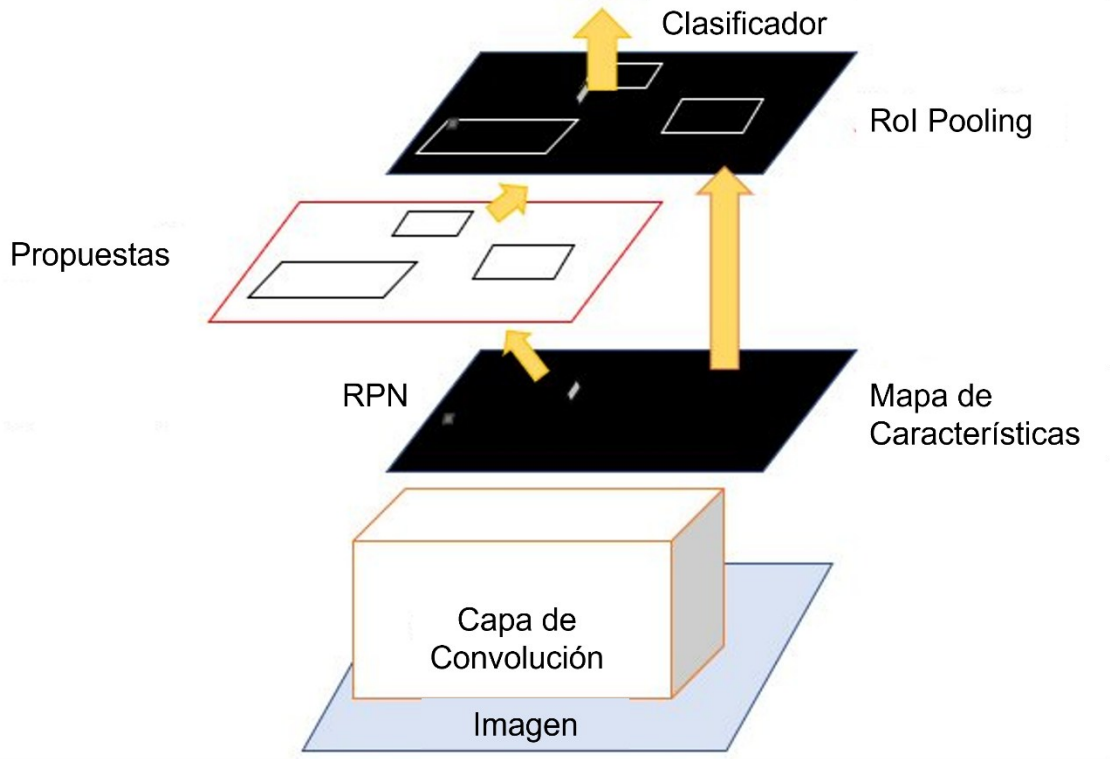


Figura 4. Arquitectura Faster R-CNN (Du et al., 2020)

3.6 Arquitectura YOLO

El modelo YOLO (You Only Look Once) es una de las arquitecturas más influyentes en el campo de la detección de objetos en tiempo real. A diferencia de los modelos por etapas como Faster R-CNN. YOLO adopta un enfoque unificado, tratando la detección de objetos como un único problema de regresión. En el que se predicen directamente las clases y las coordenadas de los objetos en una sola pasada por la red (Jocher et al., 2023).

Este diseño permite que YOLO alcance altas velocidades de procesamiento. Lo cual lo hace ideal para aplicaciones donde el tiempo de respuesta es crítico, como la conducción asistida, vigilancia o monitoreo en video en tiempo real.

3.6.1 Backbone de YOLO

El backbone de YOLO es una red convolucional profunda que se encarga de extraer las características de la imagen. Dependiendo de la versión de YOLO (Bochkovskiy et al., 2020; Jocher et al., 2023), se han utilizado diferentes arquitecturas como:

- Darknet en las versiones YOLOv3 y YOLOv4.
- CSPDarknet en YOLOv5 y YOLOv8.

Este componente transforma la imagen de entrada en un mapa de características rico en información visual, que sirve de base para las etapas posteriores de detección.

3.6.2 Neck

El neck es la parte intermedia de la arquitectura YOLO, ubicada entre el backbone y la cabeza de detección. Su función es mejorar el flujo de información entre diferentes niveles de la red (Bochkovskiy et al., 2020; Jocher et al., 2023), permitiendo una mejor detección de objetos de distintos tamaños.

Entre los componentes clave del neck se encuentran:

- Feature Pyramid Network (FPN): Combina características de múltiples escalas para mejorar la detección multiescala.
- Path Aggregation Network (PAN): Mejora la propagación de características ascendentes y descendentes para fortalecer la precisión del modelo.

Gracias al neck, el modelo puede integrar tanto características globales como locales, lo cual mejora la capacidad de detección sin sacrificar velocidad.

3.6.3 Head

La cabeza de detección (head) es la encargada de generar las predicciones finales. Divide la imagen en una cuadrícula y, para cada celda, predice varias cajas delimitadoras (bounding boxes) (Bochkovskiy et al., 2020; Jocher et al., 2023), cada una con:

- Una puntuación de confianza, que indica la probabilidad de que exista un objeto.
- Una predicción de clase, que especifica de qué objeto se trata.
- Las coordenadas del objeto (centro, ancho y alto).

Esta estructura permite que el modelo procese toda la imagen en una sola pasada, manteniendo la detección en tiempo real.

3.6.4 Predicciones conjuntas

A diferencia de modelos por etapas, YOLO realiza predicciones conjuntas para múltiples tareas de detección. Estas predicciones incluyen:

- Ubicación del objeto: Coordenadas precisas de la caja delimitadora.
- Confianza: Probabilidad de que la región contenga un objeto.
- Clasificación: Etiqueta correspondiente al tipo de objeto detectado.

Este enfoque simultáneo permite que YOLO mantenga un balance eficaz entre precisión y velocidad (Bochkovskiy et al., 2020; Jocher et al., 2023), lo que ha consolidado su uso en diversas aplicaciones prácticas.

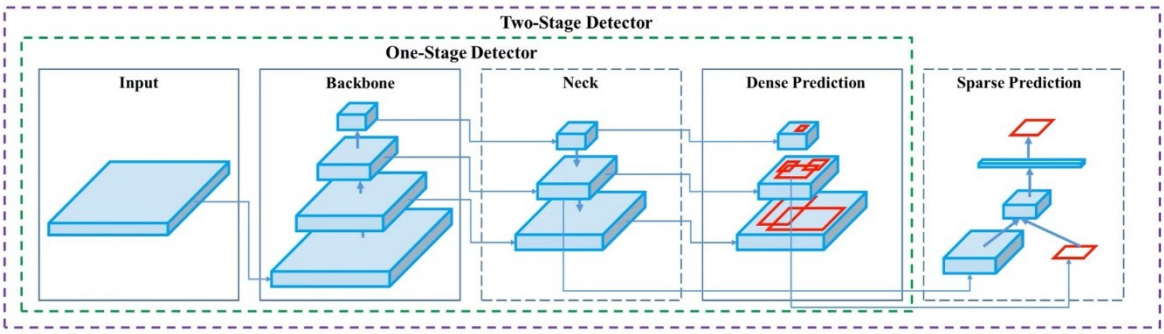


Figura 5. Presentación del complejo diseño de red YOLOv4 (Jocher et al., 2023).

IV. HIPÓTESIS

Mediante la aplicación de técnicas de procesamiento de imágenes sobre un conjunto de datos conformado por imágenes de rostros humanos, y con el uso de algoritmos de inteligencia artificial basados en redes neuronales convolucionales (CNN), es posible lograr una detección de somnolencia en conductores con una precisión comparable o superior a la de estudios previos.

V. OBJETIVOS

Implementar un modelo de red neuronal convolucional para la detección de somnolencia en conductores de vehículos, mediante la aplicación de técnicas de procesamiento de imágenes y algoritmos de inteligencia artificial. El modelo será entrenado con un conjunto de imágenes de rostros humanos con características visuales asociadas a la somnolencia (ojos cerrados, bostezo, etc.).

5.1 Objetivos específicos

- Configurar un entorno de trabajo para el desarrollo de modelos CNN, garantizando las condiciones adecuadas para su entrenamiento y validación.
- Reunir una base de datos pública que contenga imágenes de rostros humanos con características de somnolencia, para crear diferentes conjuntos mediante diversas técnicas de procesamiento de imagen.
- Utilizar inteligencia artificial para entrenar un modelo de detección basado en redes neuronales convolucionales para cada conjunto de imágenes.
- Evaluar los modelos de detección utilizando métricas de validación estandarizadas y pruebas, con el fin de identificar los modelos con mejores resultados en la detección de somnolencia a través de señales visuales, y así comparar los resultados con estudios y trabajos similares.

5.2 Cronograma de actividades

Tabla I. Cronograma de Actividades.

	2024												2025						
	Enero	Febrero	Marzo	Abril	Mayo	Junio	Julio	Agosto	Septiembre	Octubre	Noviembre	Diciembre	Enero	Febrero	Marzo	Abril	Mayo	Junio	Julio
Recopilación de base de datos																			
Investigación sobre modelos de detección de objetos																			
Selección de arquitecturas y entrenamiento de modelos																			
Evaluación de modelos																			
Redacción de tesis																			

VI. METODOLOGÍA

La metodología propuesta en este estudio, inicialmente concebida como una aproximación sencilla, consistía en la recopilación y el procesamiento básico de una base de datos, seguida

por la aplicación de modelos con estructuras inspiradas en redes neuronales convolucionales (CNN), su validación y comparación con el estado del arte, como se muestra en la siguiente figura.

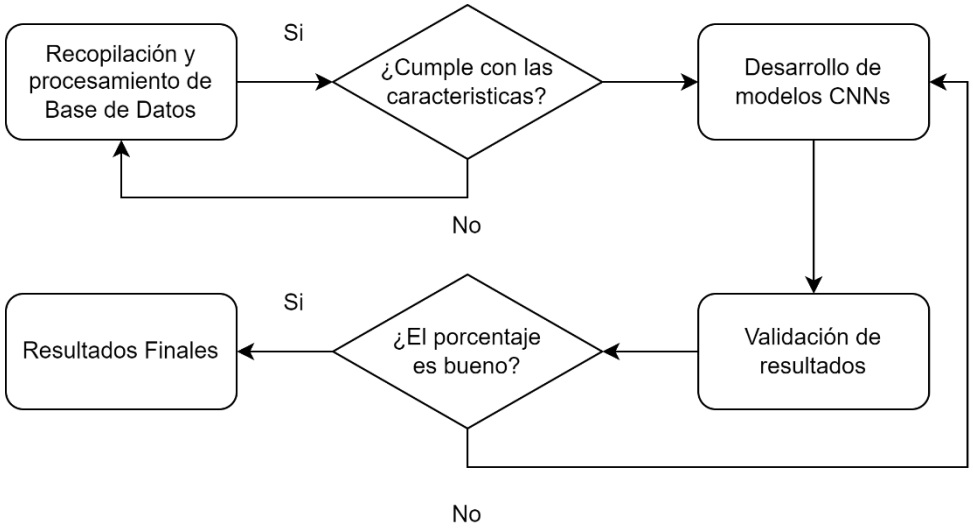


Figura 6. Metodología propuesta inicialmente.

Durante el desarrollo del proyecto surgieron diversos desafíos e imprevistos que motivaron la implementación de estrategias adicionales. Una de ellas, la construcción de un conjunto de datos diseñado específicamente para el entrenamiento de distintos modelos de redes neuronales. A partir del análisis del desempeño de estos modelos, se estableció un proceso iterativo. Lo que permitió seleccionar tanto los conjuntos de imágenes utilizados en las distintas fases del estudio como los modelos con mejor rendimiento.

En la figura siguiente se ilustra de forma más precisa el enfoque metodológico adoptado. Se muestra la progresión del trabajo desde sus fases iniciales hasta la conclusión del proyecto. Esta representación ofrece una visión integral del flujo metodológico seguido y su relación con los objetivos planteados.

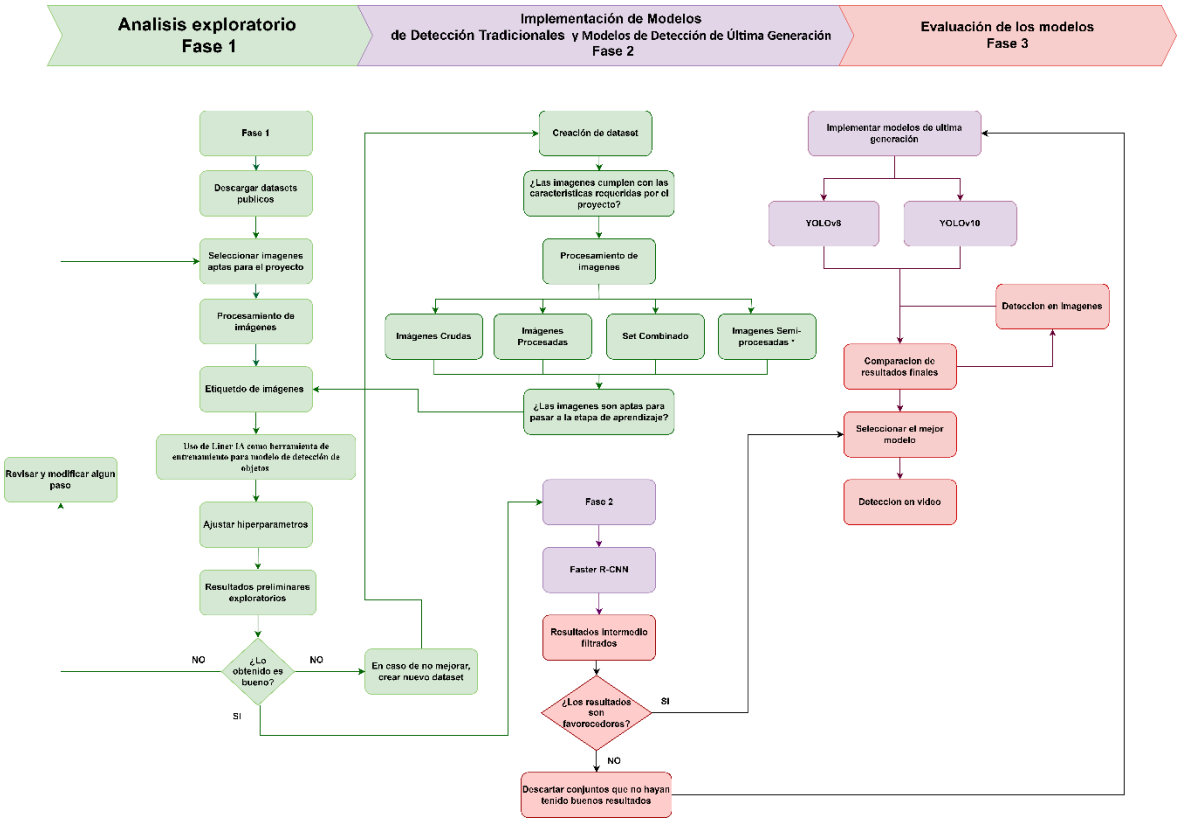


Figura 7. Metodología empleada.

6.1 Fase 1: Análisis exploratorio con datasets y Liner.ai

En esta etapa inicial, el propósito fue utilizar herramientas y datasets existentes en el área. Para ello, se eligió la plataforma Liner.ai. Este proporciona una interfaz sencilla, con instrucciones claras para la carga de datos y ejecución de modelos de detección.

Se seleccionaron y combinaron tres datasets públicos de la plataforma Kaggle (Dutta, 2022; Patil, 2025; Perumandla, 2025) con el objetivo de desarrollar un corpus unificado para el análisis. Del primer y segundo conjunto se extrajeron imágenes de ojos abiertos y cerrados; del tercero, imágenes de rostros en estado de bostezo y sin bostezo.

Como resultado del filtrado y combinación de los datos, se obtuvo el siguiente conjunto:

- Ojos abiertos: 2,595 imágenes
- Ojos cerrados: 2,726 imágenes
- Personas bostezando: 1,999 imágenes

- Personas sin bostezo: 2,523 imágenes

Este proceso permitió abordar de forma inicial los objetivos del estudio, asegurando una diversidad mínima para entrenar modelos preliminares.

6.1.1 Procesamiento del conjunto de datos y etiquetado de imágenes

Durante esta fase, se definieron dos conjuntos principales: uno compuesto con imágenes públicas (Kaggle) y otro creado provisionalmente para este estudio. Cada conjunto incluía 400 imágenes y fue procesado y etiquetado de distintas formas. Generando cinco versiones con características diferenciadas. Estas versiones se utilizaron para evaluar cómo afecta el formato, el contenido y el etiquetado a la capacidad de los modelos para detectar somnolencia.

A continuación, se presenta una tabla resumen con nombres más descriptivos para cada versión del conjunto de datos. Esta clasificación busca evitar confusiones y facilitar la comprensión de las diferencias entre versiones.

Tabla II. Resumen de bases de datos y sus características.

Nombre descriptivo	Tipo de origen	Nº de imágenes	Formato de color	Enfoque principal	Clases etiquetadas
Básico sin rostro	Público (Kaggle)	400	Escala de grises	Omisión del rostro; contiene 4 clases	Closed, Open, Yawn, No_Yawn
Básico reducido	Público (Kaggle)	400	Escala de grises	Solo ojos cerrados y bostezo	Closed, Yawn
Intermedio con rostro	Público (Kaggle)	400	Escala de grises	Incluye detección de rostro	Closed, Open, Yawn, No_Yawn, Face
Perfil facial RGB	Propio (creado)	400	RGB	Imágenes con rostros en diferentes perfiles	Closed, Open, Yawn, No_Yawn, Face
Perfil facial en escala de grises	Propio (creado)	400	Escala de grises	Versión en gris del conjunto Perfil facial RGB	Closed, Open, Yawn, No_Yawn, Face

6.1.2 Estandarización y etiquetado de imágenes

Cada conjunto de imágenes fue sometido a un proceso de estandarización para facilitar su manejo y garantizar consistencia durante las etapas de entrenamiento y validación de los modelos.

Primero, todas las imágenes fueron renombradas sistemáticamente siguiendo un esquema uniforme con formato secuencial:

img1.png, img2.png, ..., img400.png.

Esta organización estandarizada permitió una gestión más sencilla de los archivos y facilitó su integración con las herramientas de anotación y entrenamiento.

Posteriormente, según lo requerido por cada conjunto (ver Tabla II), algunas versiones fueron convertidas de RGB a escala de grises utilizando la función rgb2gray de MATLAB (MathWorks, 2025b). Esta transformación se aplicó para uniformar el espacio de color, dado que algunas imágenes estaban originalmente en escala de grises, mientras que otras se mantenían en color.

Posteriormente el proceso de etiquetado se realizó utilizando VoTT (Visual Object Tagging Tool) (Microsoft, 2020), una herramienta especializada en anotación de imágenes para tareas de detección de objetos. Cada imagen fue etiquetada manualmente siguiendo una taxonomía de clases definida previamente, que incluye:

- Closed (ojos cerrados)
- Open (ojos abiertos)
- Yawn (bostezo)
- No_Yawn (ausencia de bostezo)
- Face (rostro)

La etiqueta “Face” fue añadida solo cuando la región completa del rostro era visible. En el caso de imágenes de sonrisas o rostros serios sin señales de bostezo, estas se etiquetaron como “No_Yawn”, con el fin de retar al modelo a diferenciar entre una sonrisa y un bostezo. Lo cual representa un reto visual significativo por su similitud morfológica.

Una vez completado el etiquetado, los datos fueron exportados en formato Pascal VOC. Lo cual permitió estructurar la información de forma jerárquica y es ampliamente compatible con arquitecturas de redes neuronales modernas.

6.1.3 Uso de Liner.ai como herramienta de entrenamiento

En esta etapa preliminar, el objetivo principal fue realizar un análisis exploratorio. De esta forma evaluar la viabilidad del conjunto de datos diseñado y examinar su comportamiento en un entorno de entrenamiento inicial. Para ello, se utilizó la plataforma Liner.ai, que permite probar arquitecturas avanzadas de aprendizaje profundo enfocadas en tareas de clasificación y detección de objetos, con una interfaz amigable y configuración simplificada.

Entre los modelos disponibles, se eligió EfficientDet (Liner.ai, s/f). Esta arquitectura está basada en redes convolucionales que equilibra eficiencia computacional con alto rendimiento en tareas de detección. Esta opción resultó adecuada para el análisis rápido de los distintos subconjuntos de datos descritos anteriormente.

Se diseñaron múltiples experimentos configurando diferentes combinaciones de hiperparámetros disponibles en la plataforma. Entre los ajustes considerados se incluyeron:

- Número de iteraciones
- Aplicación o no de aumentación de datos (data augmentation)
- Variantes del conjunto de datos procesado

El propósito de estas pruebas fue identificar fortalezas y debilidades tanto del modelo como de las distintas versiones de los conjuntos de datos (ver Tabla II), observando cómo respondían a distintos niveles de procesamiento de imagen y etiquetado.

Los resultados se registraron en la Tabla III, donde se incluyen métricas de desempeño, observaciones cualitativas y comentarios sobre la capacidad de generalización del modelo.

Tabla III. Desempeño y evaluación de modelos en EfficientDet con diferentes configuraciones.

Conjunto evaluado	**Iteraciones**	**Aumentación de datos**	**mAP (%)**	**Observaciones clave**
Básico sin rostro	6,000	No	80.37	Buen mAP, pero sobreajuste evidente al no generalizar fuera del conjunto.
Básico sin rostro	1,000	Sí	21.36	Aumentación + pocas iteraciones = bajo rendimiento; incapaz de adaptarse.
Básico reducido	6,000	No	96.26	mAP alto, pero problemas con clase "Closed"; falta de diversidad en los datos.
Básico reducido	1,000	Sí	53.51	Pobre desempeño; configuración inadecuada para imágenes complejas.
Intermedio con rostro	6,000	Sí	18.14	La aumentación afectó negativamente. Se sugiere eliminarla

				para este conjunto.
Intermedio con rostro	6,000	No	52.96	A pesar de alto número de iteraciones, el aprendizaje fue insuficiente.
Perfil facial RGB	6,000	No	97.51	Resultados significativamente mejores; detección eficaz y estable.
Perfil facial en escala grises	6,000	No	98.57	Precisión muy alta; solo algunas imágenes generaron errores menores.

Los resultados obtenidos proporcionaron información significativa. En este proceso se identificó que, pese a que algunas versiones (como las basadas en datos públicos) arrojaban métricas aceptables de precisión media promedio (mAP), presentaban serias limitaciones al generalizar a nuevas imágenes. Esto sugirió que el tipo de dataset no era el más adecuado para los fines del estudio.

Por tal motivo, se tomó la decisión de crear un nuevo conjunto de imágenes, diseñado específicamente con base en las necesidades del proyecto, basado en el conjunto de datos “Perfil facial RGB” y “Perfil facial en escala de grises” (véase Tabla II): diversidad facial, etiquetado claro, variedad de poses, y un control más estricto en las condiciones de captura y procesamiento.

6.2 Creación de dataset

Con base en los resultados obtenidos durante el análisis exploratorio, se identificó la necesidad de construir un conjunto de datos más robusto, diverso y adaptado específicamente a los objetivos de esta investigación. A este conjunto se le denominará a partir de aquí como conjunto de datos crudo.

La principal innovación de este conjunto radica en la inclusión de imágenes de rostros humanos en diversas condiciones faciales: personas con ojos abiertos, ojos cerrados, en estado de bostezo, sin bostezo, con expresión neutra y con sonrisas (Figura 9). Además, para favorecer la variabilidad, se capturaron imágenes desde diferentes ángulos faciales: frontal, perfil izquierdo y perfil derecho abarcando una sumatoria de 180 grados del rostro (Figura 8). Esto permitió ampliar la diversidad morfológica y la representatividad del conjunto.

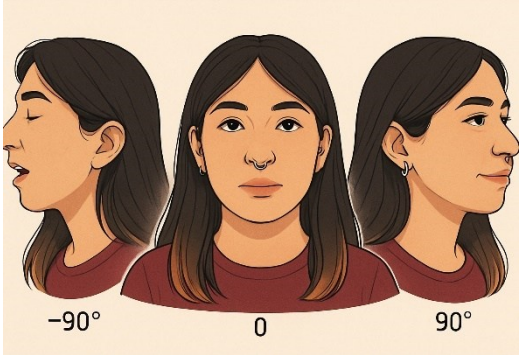


Figura 8. Ilustración generada con IA representando el rostro humano a -90°, 0° y 90°.
Fuente: Elaboración propia con ChatGPT (OpenAI, 2025)

El proceso de planeación, captura y distribución se esquematizó cuidadosamente, asegurando que todos los participantes dieran su consentimiento informado. Las imágenes se tomaron bajo diferentes condiciones de iluminación, fondo, distancia, capturadas con distintas cámaras de teléfonos celulares, y fueron etiquetadas manualmente. Este conjunto fue publicado posteriormente en Kaggle y Zenodo (MIRANDA-LEON, 2025; Miranda-Leon & Angole-Tierrablanca, 2024) para su disponibilidad pública.

Figura 9. Conjunto de imágenes faciales para diferentes expresiones y orientaciones etiquetadas en VOTT *(Miranda-Leon & Angole-Tierrablanca, 2024)*.

El conjunto de datos crudo se conformó por imágenes de 11 individuos, en 12 expresiones distintas, generando un total de 1,880 imágenes. Cada imagen fue recortada para aislar la región de interés (el rostro), utilizando un formato cuadrado de 512 x 512 píxeles, tamaño que garantiza la conservación de detalles relevantes sin exigir demasiados recursos computacionales. La elección del valor 512 responde a que es un múltiplo de 29, lo cual favorece el procesamiento en GPU y TPU.

Después del recorte, las imágenes fueron renombradas secuencialmente (img1.png, img2.png, ..., img1880.png) y se prepararon para su procesamiento y etiquetado, siguiendo un mismo formato estandarizado.

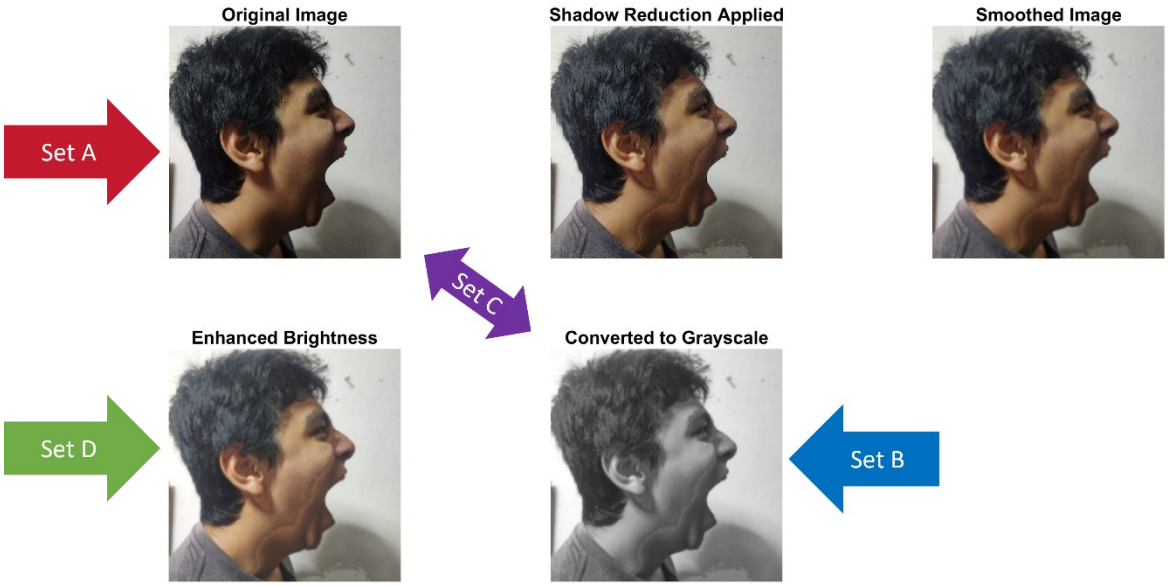


Figura 10. Diferentes etapas del procesamiento de imágenes: Set A) conjunto de imágenes crudas; Set B) conjunto de imágenes procesadas; Set C) conjunto híbrido; Set D) conjunto de imágenes semiprocesadas.

Durante el análisis exploratorio, se observaron resultados favorables en dos conjuntos de datos preliminares: uno en formato RGB y otro convertido a escala de grises. Por consiguiente, se propuso replicar estos conjuntos utilizando el conjunto de datos crudo, ahora extendido a 1,880 imágenes por conjunto. Además, se decidió crear dos nuevos conjuntos adicionales, también derivados del conjunto de datos crudo, aplicando diferentes tipos de procesamiento de imágenes.

En total, se generaron cuatro conjuntos de imágenes, cada uno con una configuración visual específica, con el objetivo de evaluar cómo estas transformaciones afectan el entrenamiento y rendimiento de los modelos de detección. En la Tabla IV se describe el tipo de procesamiento aplicado a cada conjunto, y en la Figura 10 se ilustra gráficamente el flujo de etapas involucradas.

Tabla IV. Tipos de Procesamiento en los diferentes sets empleados.

Conjuntos de Imágenes	**Tipo de Procesamiento**
Imágenes Crudas	No se aplicó ningún procesamiento de imágenes (únicamente se aplicó recordé de región de interés, cambio de nombre y de formato).
Imágenes Procesadas	• Reducción de sombras • Suavizado de imagen • Regulación de brillo • Convertido a escala de grises
Conjunto Hibrido	50 % imágenes crudas + 50 % imágenes procesadas.
Imágenes Semiprocesadas	• Reducción de sombras • Suavizado de imagen • Regulación de brillo • Se mantiene formato RGB

Cada conjunto fue etiquetado manualmente utilizando la herramienta VoTT (Visual Object Tagging Tool) (Microsoft, 2020), como se detalló previamente en la sección 6.1.3 (véase también la Figura 8). El etiquetado se basó en la presencia o ausencia de características visuales asociadas a la somnolencia subjetiva, según las siguientes clases:

- eyes_open: Ojos abiertos
- eyes_closed: Ojos cerrados
- face: Rostro
- yawn: Presencia de bostezo
- no_yawn: Ausencia de bostezo

De manera particular, las imágenes que mostraban sonrisas (con o sin ojos cerrados) fueron etiquetadas como no_yawn, al igual que los rostros neutros sin expresión. Esta decisión se tomó con la intención de agregar un nivel de complejidad al modelo, obligándolo a aprender a distinguir entre una sonrisa y un bostezo, expresiones que pueden presentar similitudes morfológicas en ciertas configuraciones faciales.

6.2.1 Imágenes Crudas

El conjunto denominado imágenes crudas, fue concebido principalmente como un grupo de control. Las imágenes de este conjunto corresponden directamente al conjunto de datos crudo. Este conjunto fue concebido como grupo de control, con el propósito de evaluar cómo responde el modelo al trabajar con datos en su estado más natural posible.

Las únicas modificaciones aplicadas fueron las mínimas necesarias para su compatibilidad con los modelos de detección:

- Recorte de la región de interés (rostro)
- Cambio de nombre y formato
- Redimensionamiento a 512 x 512 píxeles

Este conjunto se mantuvo en su formato original RGB (color), sin aplicar filtros, suavizados, ni transformaciones de color. El objetivo fue observar si el modelo era capaz de aprender directamente de los datos sin intervención previa, lo cual sirve como línea base para contrastar el impacto de los conjuntos procesados (Figura 9, conjunto: imágenes crudas).

6.2.2 Imágenes procesadas

Este conjunto representa la versión con mayor grado de procesamiento aplicado. A partir del conjunto de datos crudo, se realizaron las siguientes transformaciones:

- Reducción de sombras utilizando el espacio de color HSV
- Suavizado mediante filtro gaussiano
- Ajuste de brillo sumando/restando una constante

- Conversión a escala de grises

Estas operaciones se llevaron a cabo en MATLAB, utilizando funciones especializadas para procesamiento de imágenes. El propósito de estas transformaciones fue mejorar la calidad visual del conjunto, reduciendo ruido y estandarizando condiciones de iluminación, con la hipótesis de que esto podría facilitar el aprendizaje del modelo (Figura 9, conjunto: imágenes procesadas).

6.2.2.1 Reducción de sombras

La reducción de sombras se realizó utilizando el espacio de color HSV (Hue, Saturation, Value) (MathWorks, 2025a), ya que este permite aislar y trabajar específicamente con la componente de intensidad (V) sin afectar directamente los valores de tono ni saturación.

Los pasos implementados fueron:

1. Conversión de RGB a HSV: La imagen original se transformó al espacio HSV.
2. Generación de máscara de sombras: Se definió un umbral de intensidad $0.25 \leq V \leq 0.5$ para identificar las zonas oscuras.
3. Ajuste de intensidad: Se incrementó el valor de los píxeles que estaban dentro del rango definido para reducir visualmente las sombras.
4. Conversión inversa a RGB: La imagen fue devuelta al espacio RGB para continuar con los pasos siguientes.

Figura 11. Reducción de sombras en MATLAB: a) Imagen original; b) Imagen con reducción de sombras.

Este proceso permitió disminuir el efecto de sombras en áreas críticas como los ojos o la boca, sin distorsionar los colores originales (Figura 11).

6.2.2.2 Suavizado de imágenes

El suavizado se aplicó utilizando un filtro gaussiano, a través de la función imgaussfilt de MATLAB. Esta técnica es ideal para eliminar ruido de alta frecuencia, preservando los bordes relevantes.

Para el caso particular ilustrado en la Figura 12, se utilizó una desviación estándar σ=1.3 (1) (MathWorks, 2025c) para equilibrar suavizado y detalle. La fórmula general del filtro gaussiano bidimensional es:

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{\frac{x^2+y^2}{2\sigma^2}} \qquad (1)$$

Donde:

- Donde: x,y son las coordenadas del píxel
- σ controla el grado de desenfoque.

Figura 12. Suavizado de imagen en MATLAB: a) Imagen con reducción de sombras; b) Imagen con filtro gaussiano.

Este suavizado facilitó que el modelo se concentrara en los rasgos faciales dominantes sin verse afectado por texturas o ruido visual.

6.2.2.3 Ajuste de brillo

El ajuste de brillo se realizó mediante la suma o resta de una constante a cada elemento de la matriz que representa la imagen. Esto permite aumentar o disminuir la luminosidad general sin alterar la estructura subyacente de la imagen (véase la Figura 13). La operación básica (Gonzalez & Woods, 2018) se define como:

$$I'(x,y) = I(x,y) + \beta \qquad (2)$$

Donde:

- I(x,y) es la intensidad original del píxel en las coordenadas

- I’(x,y) es la intensidad ajustada.
- β es la constante de ajuste (positiva para incrementar el brillo, negativa para reducirlo).

Figura 13 Ajuste de Brillo en una imagen.

6.2.2.4 Conversión a escala de grises

La conversión a escala de grises se realizó mediante la función estándar de MATLAB (MathWorks, 2025b), que utiliza la fórmula ponderada comúnmente aceptada en el procesamiento de imágenes:

$$\mathrm{EG}(x,y) = 0.2989 \cdot \mathrm{R}(x,y) + 0.5870 \cdot \mathrm{G}(x,y) + 0.1140 \cdot \mathrm{B}(x,y) \qquad (3)$$

Donde:

- EG(x,y) es la intensidad en escala de grises.
- R(x,y), G(x,y), y B(x,y) son las intensidades de los canales Rojo, Verde y Azul, respectivamente.

Esta fórmula considera la sensibilidad del ojo humano a los distintos canales de color. La conversión a grises permitió reducir la dimensionalidad de las imágenes, mejorando la eficiencia computacional y eliminando posibles variaciones cromáticas no relevantes para la

detección de somnolencia. Esta conversión solo se aplicó al conjunto imágenes procesadas, no al conjunto semiprocesado RGB (Figura 9, conjunto: imágenes procesadas).

6.2.3 Conjunto hibrido

Este conjunto fue diseñado como un híbrido entre los dos conjuntos anteriores, para evaluar cómo afecta al rendimiento del modelo el mezclar imágenes en diferentes condiciones visuales.

Se construyó seleccionando aleatoriamente el 50 % de las imágenes del conjunto de imágenes crudas y el 50 % del conjunto de imágenes procesadas, garantizando que ambas partes contuvieran una representación balanceada de todas las clases etiquetadas.

La intención fue probar si el modelo podía beneficiarse de la diversidad presente en las imágenes sin comprometer la precisión ni provocar confusión durante el entrenamiento (Figura 9, conjunto: conjunto hibrido).

6.2.4 Imágenes semiprocesadas

Este conjunto representa una estrategia intermedia entre los extremos anteriores. Se partió nuevamente del conjunto de datos crudo y se aplicaron los siguientes pasos:

- Reducción de sombras (sección 6.2.2.1)
- Suavizado (sección 6.2.2.2)
- Ajuste de brillo (sección 6.2.2.3)

A diferencia del conjunto de imágenes procesadas, en este caso no se convirtió a escala de grises. Las imágenes se conservaron en formato RGB, con la intención de evaluar si la información de color podía aportar valor al modelo en combinación con las mejoras visuales derivadas del preprocesamiento.

Este conjunto fue diseñado bajo la hipótesis de que una optimización visual moderada sin pérdida de información cromática podría ofrecer un mejor equilibrio entre generalización y precisión (Figura 9, conjunto: imágenes semiprocesadas).

6.3 Métricas de Evaluación

Para evaluar el rendimiento de los modelos de detección de somnolencia implementados en este estudio, fue necesario establecer un conjunto de métricas estandarizadas y ampliamente reconocidas por la comunidad científica en el ámbito del aprendizaje profundo.

Estas métricas se aplicaron de manera transversal a lo largo de las distintas fases de experimentación (EfficientDet en Liner.ai, Faster R-CNN, YOLOv8 y YOLOv10), permitiendo comparaciones consistentes entre configuraciones, conjuntos de datos y versiones procesadas.

Además de servir como criterios de validación cuantitativa, las métricas también fueron la base para la generación de gráficas de pérdida y precisión, las cuales ofrecieron una representación visual del comportamiento de los modelos a lo largo de su proceso de entrenamiento.

Métricas empleadas:

- Accuracy (exactitud): Mide la proporción de clasificaciones correctas en relación con el total de predicciones (Developers, 2025). Es útil como visión general del rendimiento, especialmente en conjuntos de datos balanceados.
- Precision (precisión): Evalúa cuántas de las predicciones positivas fueron realmente correctas (Developers, 2025). Es crítica cuando se requiere evitar falsos positivos, como en la detección de señales de somnolencia no presentes.
- Recall (sensibilidad o exhaustividad): Indica cuántos de los casos positivos reales fueron correctamente identificados por el modelo .(Developers, 2025) Es clave cuando el costo de omitir un caso (falso negativo) es alto.

- F1-Score: Combina precisión y recall en una sola métrica armónica (developers, 2025). Es particularmente útil cuando se busca un equilibrio entre evitar falsos positivos y falsos negativos.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (4) \qquad Precision = \frac{TP}{TP + FP} \quad (5)$$

$$Recall = \frac{TP}{TP + FN} \quad (6) \qquad F1 - score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (7)$$

Donde:

- TP es igual al número de verdaderos positivos
- TN es igual al número de verdaderos negativos
- FP es igual al número de falsos positivos
- FN es igual al número de falsos negativos

Además de las métricas anteriores, se utilizaron medidas especializadas para modelos de detección de objetos, como Faster R-CNN y YOLO, que requieren no solo clasificar correctamente, sino también ubicar con precisión los objetos (regiones de interés).

- IoU (Intersection over Union): Mide el grado de superposición entre la caja predicha y la real (Ultralytics, 2025). Se calcula como la intersección dividida entre la unión de ambas cajas.

- mAP@50 (Mean Average Precision con umbral 0.50): Promedia la precisión de detección para todas las clases, considerando como correcta una predicción si el

IoU ≥ 0.50 (Ultralytics, 2025). Es una métrica clave para detección en condiciones estándar.

- mAP@50-95: Igual que mAP, pero calculado a través de 10 umbrales desde 0.50 hasta 0.95 en intervalos de 0.05 (Ultralytics, 2025). Proporciona una visión más exigente y completa del desempeño del modelo.
- FPS (Frames per Second): Mide la velocidad de procesamiento del modelo. Es especialmente relevante cuando se busca aplicar la detección en tiempo real, como en imágenes o video (Redmon & Farhadi, 2018; Szeliski, 2010).

$$mAP@50-95 = \frac{1}{10}\sum_{IoU=0.5}^{0.95} mAP@IoU \quad (8) \qquad mAP@50 = \frac{\sum_{i=1}^{N} AP_i}{N} \quad (9)$$

$$IoU = \frac{Área\ de\ superposición}{Área\ de\ unión} \quad (10) \qquad AP = \frac{1}{N}\sum_{i=1}^{N} Precision_i \quad (11)$$

Donde:

N es igual al número de clases

i es igual a la clase en curso

Estas métricas fueron fundamentales no solo para evaluar modelos, sino también para tomar decisiones informadas respecto a:

- Qué conjunto de datos utilizar en fases posteriores
- Qué arquitectura conservar como base para pruebas en video
- Qué configuraciones evitaban el sobreajuste o la pérdida de generalización

Su inclusión permite respaldar cuantitativamente la validez de los resultados obtenidos y garantizar la reproducibilidad y rigor científico del trabajo

6.4 Recursos computacionales

La implementación, entrenamiento y validación de los distintos modelos de detección de somnolencia requerían recursos computacionales considerables. Por ello, se combinaron dos entornos de trabajo: uno local y otro basado en la nube, con el objetivo de equilibrar eficiencia, accesibilidad y capacidad de procesamiento.

Entorno local:

Durante las fases iniciales del proyecto, se empleó un equipo de cómputo personal con las siguientes características:

- Procesador: AMD Ryzen 5 5600H con gráficos Radeon, frecuencia base de 3.30 GHz
- Memoria RAM: 8 GB (5.86 GB utilizables)
- Sistema operativo: Windows 11 Home Single Language, versión 23H2
- Arquitectura: 64 bits

Este entorno fue utilizado principalmente para la planificación del proyecto, procesamiento de imágenes en MATLAB y ejecución de scripts preliminares en Python. Sin embargo, sus limitaciones en memoria y capacidad de GPU restringieron su uso para el entrenamiento de modelos más complejos.

Entorno en la nube – Google Colab:

Para las fases de entrenamiento y evaluación de los modelos Faster R-CNN, YOLOv8 y YOLOv10, se empleó la plataforma Google Colab, aprovechando sus recursos gratuitos de aceleración por GPU. Las especificaciones disponibles durante el desarrollo del proyecto fueron:

- Memoria RAM del sistema: 53.0 GB (uso promedio: 1.7 GB)
- Memoria RAM de GPU: 22.5 GB disponibles

- Espacio en disco: 201.2 GB disponibles (uso promedio: 27.3 GB)

Este entorno permitió entrenar los modelos con el conjunto de datos completo (1,880 imágenes) en tiempos razonables, reduciendo significativamente el tiempo de cómputo en comparación con el entorno local.

Otros:

- MATLAB, utilizado para el preprocesamiento visual (reducción de sombras, suavizado, ajuste de brillo), fue ejecutado en entorno local.
- Python, junto con bibliotecas como PyTorch y Ultralytics, se utilizó en Colab para entrenar y comparar los modelos.

6.5 Fase 2: Implementación de Modelos de Detección Tradicionales y Modelos de Detección de Última Generación

Una vez construido y procesado el conjunto de datos crudo y sus variantes. La fase 2 del estudio se enfocó en la implementación de modelos de detección de objetos, tanto tradicionales como de última generación. Esta elección no fue arbitraria ya que se buscó deliberadamente contrastar diferentes enfoques arquitectónicos. De forma que permitiera analizar no solo el desempeño cuantitativo, sino también los retos y ventajas de cada tipo de modelo.

Por un lado, se implementó el modelo Faster R-CNN, una arquitectura de tipo two-stage, ampliamente reconocida por su precisión. Por otro lado, se integraron modelos de tipo one-stage, como YOLOv8 y YOLOv10, conocidos por su alta velocidad de detección y aplicabilidad en tiempo real.

Este diseño experimental permitió:

- Observar el comportamiento del aprendizaje entre modelos two-stage vs. one-stage
- Identificar las ventajas de precisión en arquitecturas más complejas como Faster R-CNN

- Contrastar la eficiencia computacional de los modelos YOLO en escenarios donde la detección rápida es prioritaria
- Explorar qué tanto influye el procesamiento visual de los datos en modelos con arquitecturas distintas

Cabe mencionar que no se buscó realizar una comparación directa e injusta entre estas familias de modelos, dado que sus estructuras, objetivos y aplicaciones prácticas son diferentes. Más bien, se pretendió obtener una visión complementaria del desempeño de cada enfoque dentro del contexto del problema: la detección de somnolencia mediante señales visuales en condiciones similares a la realidad.

6.5.1 Modelo Faster R-CNN

Esta elección se basó en los hallazgos obtenidos durante la fase exploratoria, donde se concluyó que los conjuntos de datos públicos no ofrecían la diversidad ni la estructura necesarias para una detección efectiva. Por ello, se emplearon cuatro conjuntos de datos específicos (ver sección 6.2) y se entrenó el modelo en condiciones controladas.

El modelo utilizado fue una versión de Faster R-CNN con ResNet-50 y Feature Pyramid Network (FPN), tal como se encuentra disponible en la librería PyTorch. Esta arquitectura fue elegida por su precisión comprobada y su papel como referente clásico en detección de objetos.

Para adaptar el modelo a los datos del estudio sin requerir entrenamiento desde cero, se utilizó la técnica de aprendizaje por transferencia (transfer learning), que consiste en reutilizar un modelo previamente entrenado y adaptarlo a un nuevo conjunto de datos. Esto permitió:

- Aprovechar pesos previamente entrenados en conjuntos de datos generales
- Reducir tiempos de entrenamiento
- Mejorar la capacidad de generalización del modelo

La capa de clasificación fue modificada para ajustarse a las seis clases definidas: eyes_open, eyes_closed, yawn, no_yawn, face y background.

Para evaluar sistemáticamente el rendimiento del modelo, se establecieron cuatro experimentos independientes, uno por cada conjunto de datos:

- Experimento A: Imágenes crudas
- Experimento B: Imágenes procesadas
- Experimento C: Conjunto combinado
- Experimento D: Imágenes semiprocesadas

A todos los experimentos se les aplicó la misma configuración de entrenamiento para asegurar comparabilidad. A continuación, se detallan los hiperparámetros utilizados:

Tabla V. Hiperparámetros empleados en los Experimentos A, B, C y D (Faster R-CNN con ResNet-50 y FPN).

Hiperparámetro	**Valor**
Tamaño del lote (batch size)	4
Número de workers	2
Tasa de aprendizaje inicial	0.01
Momentum	0.9
Decaimiento de pesos (weight decay)	0.005
Tamaño del paso (step size)	3
Gamma	0.1
Épocas	100

6.5.2 Resultados intermedios filtrados

Debido a las limitaciones computacionales del entorno disponible, el modelo Faster R-CNN no pudo ser entrenado con el 100 % de las imágenes de cada conjunto de datos. Considerando el tamaño total de 1,880 imágenes por conjunto, se optó por utilizar únicamente el 25 % de las imágenes en cada experimento, es decir, 470 imágenes por conjunto (dichas imágenes fueron seleccionadas heurísticamente).

Esta decisión permitió mantener tiempos de entrenamiento razonables sin comprometer la posibilidad de obtener resultados comparativos entre conjuntos. Además, se conservaron sin cambios los hiperparámetros definidos previamente (ver Tabla V), lo que aseguró condiciones experimentales uniformes.

Tabla VI. Resultados de evaluación con Faster R-CNN (25 % del conjunto de datos).

Conjunto de datos	**Épocas**	**Tiempo de entrenamiento (h)**	**Loss**	**Precisión**	**Recall**	**F1-score**	**Accuracy**	**FPS**
Experimento A: Imágenes crudas	100	11.2	0.999	0.020	0.996	0.9985	0.9944	0.15
Experimento B: Imágenes procesadas	100	9.5	0.999	0.021	0.997	0.9987	0.9952	0.15
Experimento C: Conjunto combinado	100	11.08	0.999	0.021	0.995	0.9983	0.993	0.15
Experimento D: Imágenes semiprocesadas	100	11.13	0.999	0.020	0.997	0.9986	0.995	0.15

A pesar de entrenarse con solo una fracción del total de datos, los resultados obtenidos fueron altamente consistentes entre conjuntos (Tabla VI), con variaciones mínimas en las métricas clave. Esto demuestra que Faster R-CNN fue capaz de adaptarse a los diferentes formatos visuales, manteniendo una alta precisión y capacidad de detección incluso en condiciones reducidas de entrenamiento.

La Figura 14 muestra las curvas de pérdida y precisión obtenidas durante el entrenamiento de Faster R-CNN sobre los cuatro conjuntos de datos. En todos los casos, las curvas muestran una convergencia estable, lo que indica que los hiperparámetros empleados fueron adecuados y que el modelo fue capaz de aprender de manera efectiva sin síntomas de sobreajuste.

En particular, se observa que los conjuntos procesados y semiprocesados muestran una tasa de convergencia ligeramente más rápida que el conjunto de imágenes crudas. Esto puede atribuirse al hecho de que las técnicas de procesamiento (como la reducción de sombras y el suavizado) contribuyen a eliminar ruido visual y facilitan que el modelo se concentre en los rasgos más importantes, como ojos y boca.

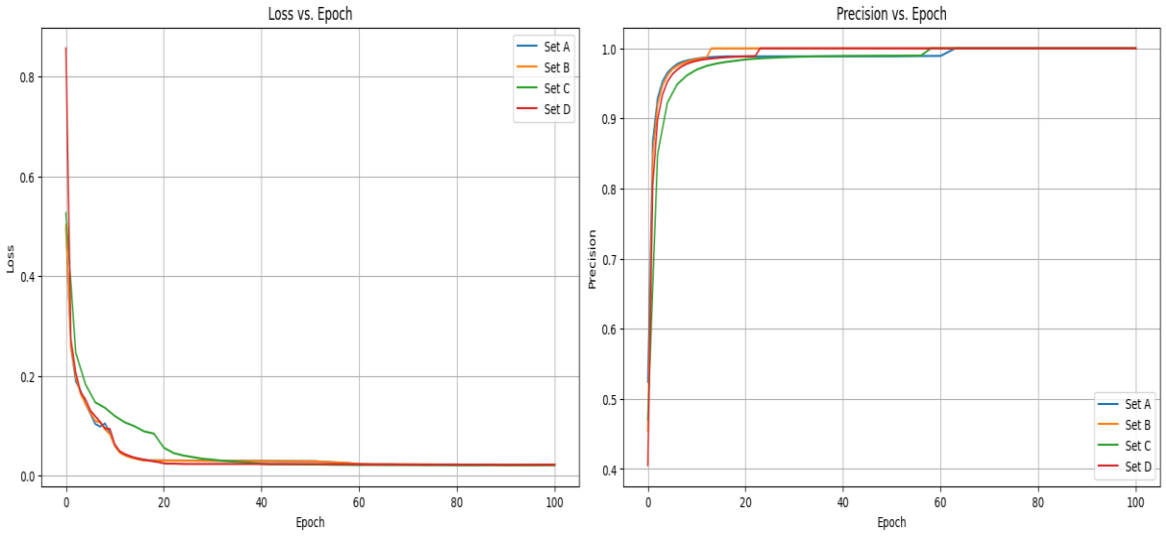


Figura 14. Gráficas de pérdida y precisión obtenidas en los experimentos realizados con el modelo Faster R-CNN utilizando los siguientes conjuntos de imágenes: Set A) conjunto de imágenes crudas; Set B) conjunto de imágenes procesadas; Set C) conjunto híbrido; Set D) conjunto de imágenes semiprocesadas.

Un hallazgo relevante fue que el conjunto híbrido (compuesto por el 50 % de imágenes crudas y 50 % de imágenes procesadas) obtuvo el desempeño más bajo entre los cuatro conjuntos evaluados. Este resultado sugiere que, aunque en teoría la mezcla de imágenes con

distintos niveles de procesamiento podría ofrecer mayor diversidad de entrenamiento, en la práctica puede introducir una señal ambigua al modelo.

La razón probable es que las imágenes crudas y procesadas, al presentar diferencias en textura, iluminación y color, generan inconsistencias estadísticas dentro del mismo conjunto. Esto puede dificultar que el modelo aprenda patrones claros y generalizables, especialmente tratándose de una arquitectura como Faster R-CNN, que es sensible a variaciones visuales cuando el conjunto no está suficientemente balanceado o normalizado.

Este resultado refuerza la idea de que, para modelos de tipo two-stage, la consistencia visual del conjunto de datos es más crítica que la variabilidad superficial. Por esta razón, el conjunto híbrido fue descartado en las siguientes fases de experimentación con modelos YOLO.

6.6 Fase 3: Modelos YOLOv8 y YOLOv10

Tras la evaluación del modelo Faster R-CNN, cuya arquitectura de dos etapas ofreció resultados altamente precisos, pero con alto costo computacional, se procedió a implementar una fase complementaria utilizando modelos YOLOv8 y YOLOv10, conocidos por su velocidad, eficiencia y capacidad de detección en tiempo real.

Estos modelos representan una evolución significativa en las técnicas de visión por computadora basadas en redes neuronales convolucionales. A diferencia de las arquitecturas por etapas, YOLO es un modelo de detección one-stage, que realiza la localización y clasificación de objetos en una sola pasada por la red. Lo cual lo hace especialmente adecuado para aplicaciones en video, dispositivos embebidos o entornos donde la latencia es un factor crítico.

La selección de YOLOv8 y YOLOv10 se basó en tres criterios fundamentales:

- Velocidad de inferencia: Son capaces de procesar múltiples cuadros por segundo sin comprometer severamente la precisión.

- Compatibilidad con detección multiclase: Ambos modelos permiten detectar simultáneamente diferentes señales visuales asociadas a la somnolencia (ojos abiertos/cerrados, rostro, bostezo, etc.).
- Arquitectura moderna y optimizada: YOLOv10 incorpora mejoras estructurales frente a versiones anteriores, permitiendo una mayor eficiencia de entrenamiento y ejecución.

A diferencia de Faster R-CNN, los modelos YOLO requieren que las anotaciones estén estructuradas en formato TXT, compatible con su esquema de entrenamiento. Por tanto, fue necesario realizar una conversión de las etiquetas desde el formato Pascal VOC a formato YOLO, asegurando que todas las coordenadas y clases fueran correctamente traducidas.

Posteriormente, cada conjunto de datos fue dividido en dos subconjuntos:

- 80 % para entrenamiento
- 20 % para validación

Esta partición fue aplicada a los tres conjuntos seleccionados crudo, procesado y semiprocesado. Como consecuencia de un desempeño inferior en comparación con los demás conjuntos experimentales, se omitió el conjunto hibrido.

Los hiperparámetros empleados en el proceso de entrenamiento de los modelos YOLOv8 y YOLOv10 se presentan a continuación en una lista detallada con sus respectivos valores en la Tabla VIII.

Tabla VII. Hiperparámetro empleados en los experimentos con YOLOv8 y YOLOv10.

Hiperparámetro	**Valor**
Optimizador	AdamW
Tasa de aprendizaje (lr)	0.001111
Momentum	0.9
Weight decay (pesos)	0.0005
Bias decay	0.0
Tamaño de imagen (entrenamiento)	512 × 512
Tamaño de imagen (validación)	512 × 512
Número de workers del dataloader	8
Épocas	100

6.6.1 Comparación de Resultados de YOLOv8 y YOLOv10

Una vez definidos y entrenados los experimentos con los modelos YOLOv8 y YOLOv10 sobre los tres conjuntos de datos seleccionados (crudo, procesado y semiprocesado), se procedió a comparar sus resultados en términos de precisión, recall, pérdida, velocidad de procesamiento y mAP.

A diferencia de la fase anterior con Faster R-CNN, en esta etapa fue posible entrenar con el 100 % de las imágenes de cada conjunto (1,880 imágenes), debido a la eficiencia computacional superior de las arquitecturas YOLO. Esto permitió observar el comportamiento completo del modelo en condiciones más realistas y robustas.

Los resultados obtenidos se muestran en la siguiente tabla

Tabla VIII. Métricas de evaluación para YOLOv8 y YOLOv10 entrenados con los conjuntos de imágenes crudas, procesadas y semiprocesadas.

Modelo	**Conjunto de datos**	**Épocas**	**Tiempo (h)**	**Precisión**	**Recall**	**Loss**	**mAP@50**	**FPS**
YOLOv8	Imágenes crudas	100	0.85	0.985	0.9851	0.2480	0.98884	4.07
	Imágenes procesadas	100	0.88	0.982	0.9894	0.2608	0.99202	4.92
	Imágenes semiprocesadas	100	0.85	0.994	0.9963	0.2547	0.99471	5.18
YOLOv10	Imágenes crudas	100	2.51	0.981	0.9822	0.4224	0.98767	4.21
	Imágenes procesadas	100	2.23	0.986	0.9849	0.4487	0.99109	4.66
	Imágenes semiprocesadas	100	2.35	0.985	0.9854	0.4435	0.99396	4.77

Los resultados muestran que YOLOv8 superó a YOLOv10 en eficiencia de entrenamiento, completando los experimentos en menos de la mitad del tiempo requerido por YOLOv10, con un rendimiento igual o superior en la mayoría de las métricas.

El mejor desempeño general se observó en YOLOv8 con el conjunto de imágenes semiprocesadas, donde se obtuvo un mAP@50 de 0.99471, acompañado de una alta precisión y recall, así como una pérdida más baja que la reportada en YOLOv10 bajo las mismas condiciones. Esto sugiere que el modelo se benefició tanto de los ajustes visuales (como la reducción de sombras y el suavizado) como de mantener la información cromática en formato RGB.

Por otro lado, aunque YOLOv10 mostró una leve mejora en algunas métricas de recall con el conjunto de imágenes procesadas, su costo computacional fue mayor, lo que podría ser un factor limitante en aplicaciones en tiempo real o dispositivos con recursos limitados.

Ambos modelos mantuvieron un rendimiento estable en cuanto a velocidad de procesamiento, logrando entre 4 y 5 FPS, lo cual es aceptable para escenarios de detección en video con requisitos de latencia moderada.

Ahora bien, En las curvas de las Figura 15 y 16 se observan varias tendencias clave:

YOLOv8 mostró una pérdida más baja y una convergencia más rápida en las primeras 25 épocas, alcanzando una meseta estable antes de la época 40. Esta caída controlada sugiere que el modelo aprendió rápidamente los patrones visuales relevantes sin grandes oscilaciones, lo cual suele estar asociado a una buena inicialización de pesos y una arquitectura eficiente.

En contraste, YOLOv10 presentó una pérdida inicial más alta y una curva de descenso más lenta. Aunque eventualmente se estabilizó, lo hizo en torno a un valor mayor que YOLOv8, y mostró pequeñas fluctuaciones incluso después de la época 60. Esto podría deberse a que la arquitectura, aunque más reciente, tiene mayor profundidad o parámetros más complejos, lo que exige un tiempo de ajuste mayor o una mayor sensibilidad a los datos de entrada.

En ambos modelos, no se observaron indicios de sobreajuste, ya que las curvas no mostraron repuntes ni aumentos abruptos al final del entrenamiento. Esto es un indicio de que los conjuntos de datos fueron bien construidos y balanceados.

El análisis visual de las curvas de pérdida, en conjunto con las métricas finales, refuerza la conclusión de que YOLOv8 no solo fue más rápido, sino también más eficiente en su proceso de aprendizaje, lo que lo convierte en una opción especialmente atractiva.

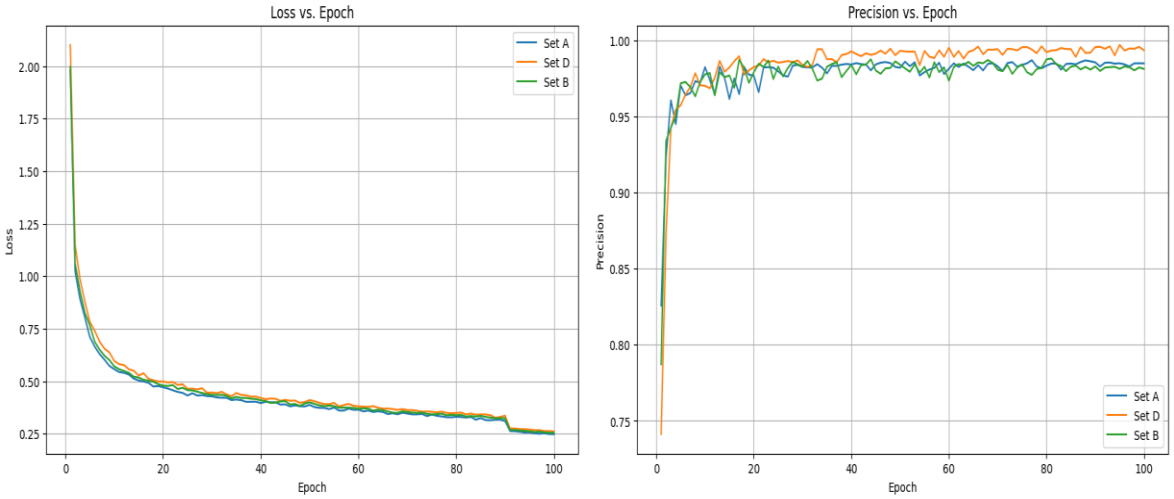


Figura 15. Gráficas de pérdida y precisión del modelo YOLOv8 entrenado con los siguientes conjuntos de imágenes: Set A) conjunto de imágenes crudas; Set B) conjunto de imágenes procesadas; Set D) conjunto de imágenes semiprocesadas.

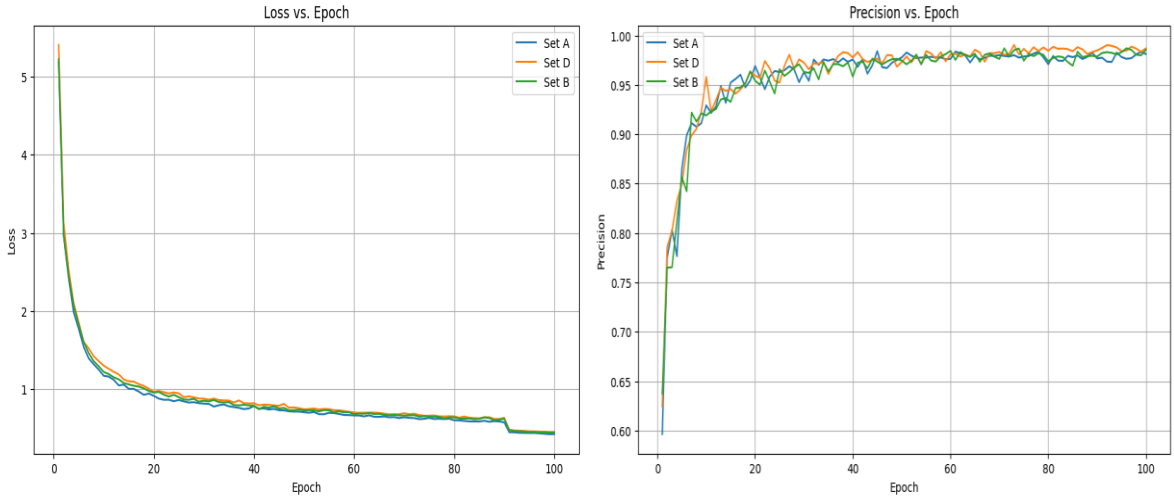


Figura 16. Gráficas de pérdida y precisión del modelo YOLOv10 entrenado con los siguientes conjuntos de imágenes: Set A) conjunto de imágenes crudas; Set B) conjunto de imágenes procesadas; Set D) conjunto de imágenes semiprocesadas.

VII. DISCUSIÓN DE RESULTADOS

Tras analizar en detalle las métricas y gráficas de pérdida durante el entrenamiento, se procedió a evaluar el desempeño de los modelos utilizando imágenes no vistas (es decir, no incluidas en el proceso de entrenamiento ni validación), con el objetivo de verificar su capacidad de generalización ante condiciones nuevas.

En la Figura 17, se presentan los resultados del modelo YOLOv8 aplicados a imágenes que fueron preprocesadas con los mismos parámetros utilizados durante su entrenamiento. De manera análoga, la Figura 18 muestra el comportamiento del modelo YOLOv10 bajo las mismas condiciones experimentales.

Por otro lado, las Figuras 19 y 20 ilustran resultados en imágenes completamente externas, que no pertenecen a ningún conjunto empleado en fases previas del estudio. Estas imágenes no fueron sometidas a ningún tipo de procesamiento y, en algunos casos, provienen de conjuntos de datos públicos. A pesar de ello, ambos modelos mantuvieron un nivel de precisión visual comparable, lo que sugiere un buen nivel de generalización.

La comparación cualitativa entre YOLOv8 y YOLOv10 no evidencia diferencias sustanciales en la detección de objetos. Ambas arquitecturas fueron capaces de identificar correctamente rasgos faciales asociados a la somnolencia, tales como ojos cerrados y bostezo, incluso en condiciones no controladas.

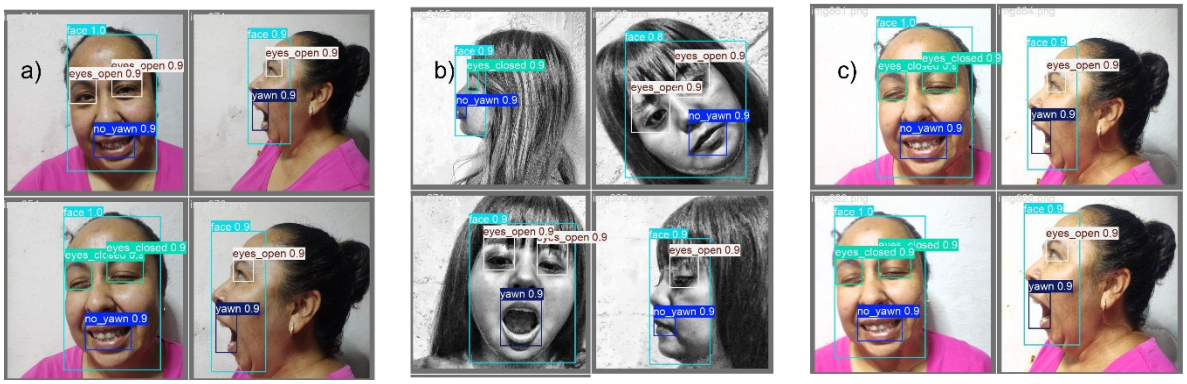


Figura 17. Detección de características de somnolencia en imágenes utilizando el modelo YOLOv8: a) entrenado con imágenes crudas; b) entrenado con imágenes procesadas; c) entrenado con imágenes semiprocesadas.

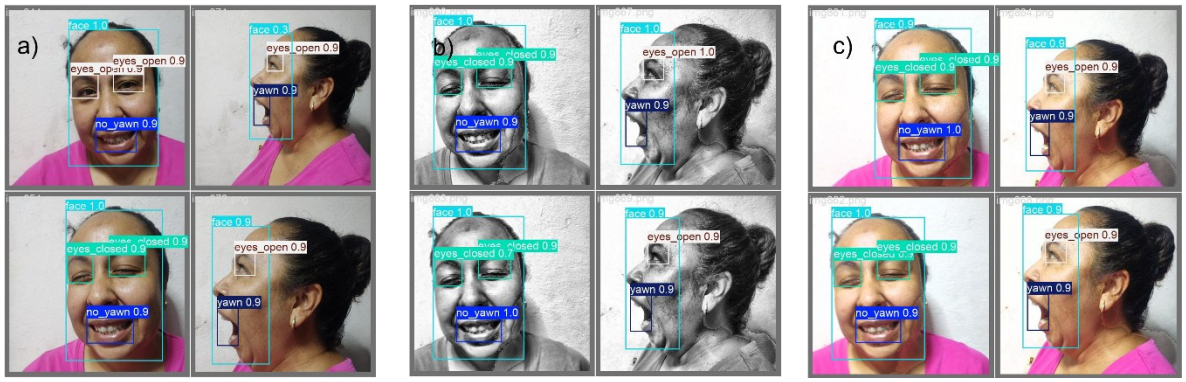


Figura 18. Detección de características de somnolencia en imágenes utilizando el modelo YOLOv10: a) entrenado con imágenes crudas; b) entrenado con imágenes procesadas; c) entrenado con imágenes semiprocesadas.

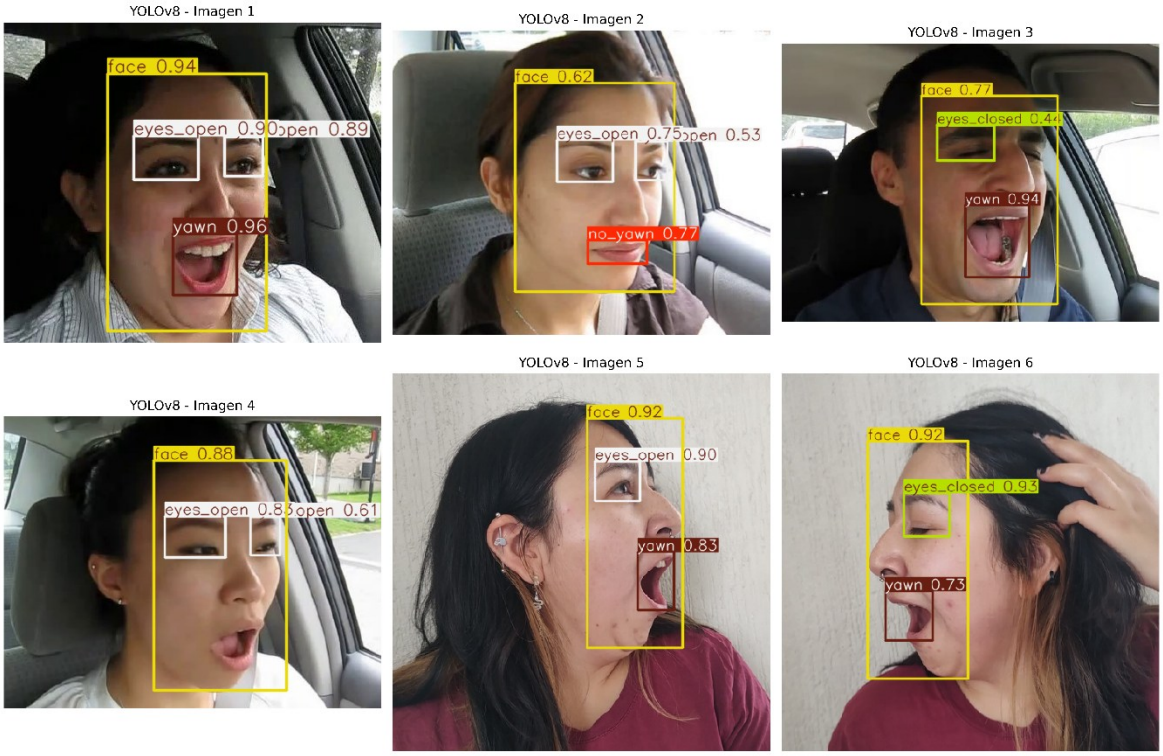


Figura 19. Detección de características de somnolencia en imágenes que no pertenecen a ninguno de los conjuntos utilizados para el entrenamiento o la validación del modelo YOLOv8.

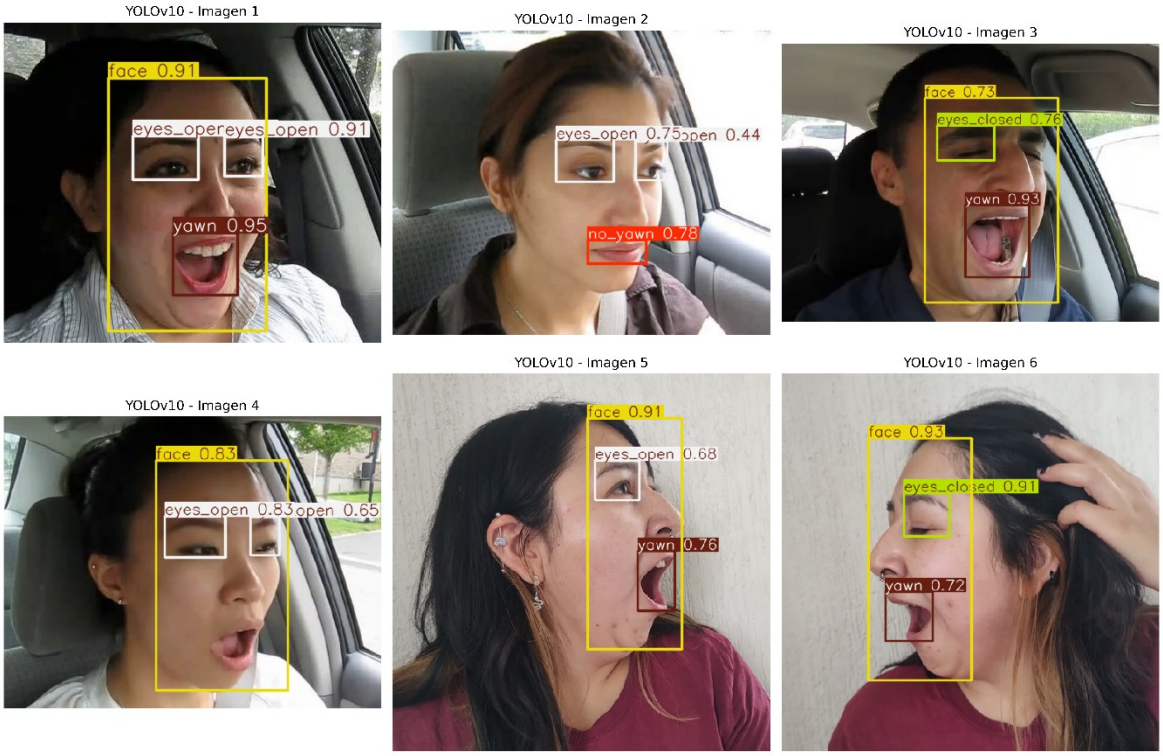


Figura 20. Detección de características de somnolencia en imágenes que no pertenecen a ninguno de los conjuntos utilizados para el entrenamiento o la validación del modelo YOLOv10.

Dado que el desempeño visual fue similar, el criterio final de selección se basó en el análisis cuantitativo de métricas. En este sentido, el modelo YOLOv8 entrenado con imágenes semiprocesadas obtuvo los mejores resultados en precisión, recall y mAP, superando tanto a su versión con imágenes crudas como al modelo YOLOv10.

Además de su alto rendimiento, YOLOv8 presentó tiempos de entrenamiento significativamente menores, lo que representa una ventaja clave para implementaciones futuras en sistemas de detección en tiempo real, especialmente en entornos con recursos computacionales limitados.

Por estas razones, se seleccionó YOLOv8 como la opción más viable para las pruebas de detección en video, descritas en la siguiente sección (véase 6.6).

Una vez identificado el modelo con mejor desempeño (YOLOv8 entrenado con el conjunto de imágenes semiprocesadas), se procedió a probar su efectividad en la detección de señales de somnolencia en secuencias de video, simulando un entorno real de monitoreo vehicular.

El objetivo de esta fase fue evaluar la capacidad del sistema para operar en tiempo real, integrando el modelo de detección en un flujo continuo de imágenes (frames), y generando alertas dinámicas conforme a las condiciones visuales detectadas en el rostro del conductor.

Para llevar a cabo esta tarea, se desarrolló un algoritmo lógico que combina la detección de objetos por parte del modelo con una serie de reglas condicionales basadas en la evaluación de etiquetas por frame. El procedimiento general se describe en el diagrama de flujo de la Figura 21.

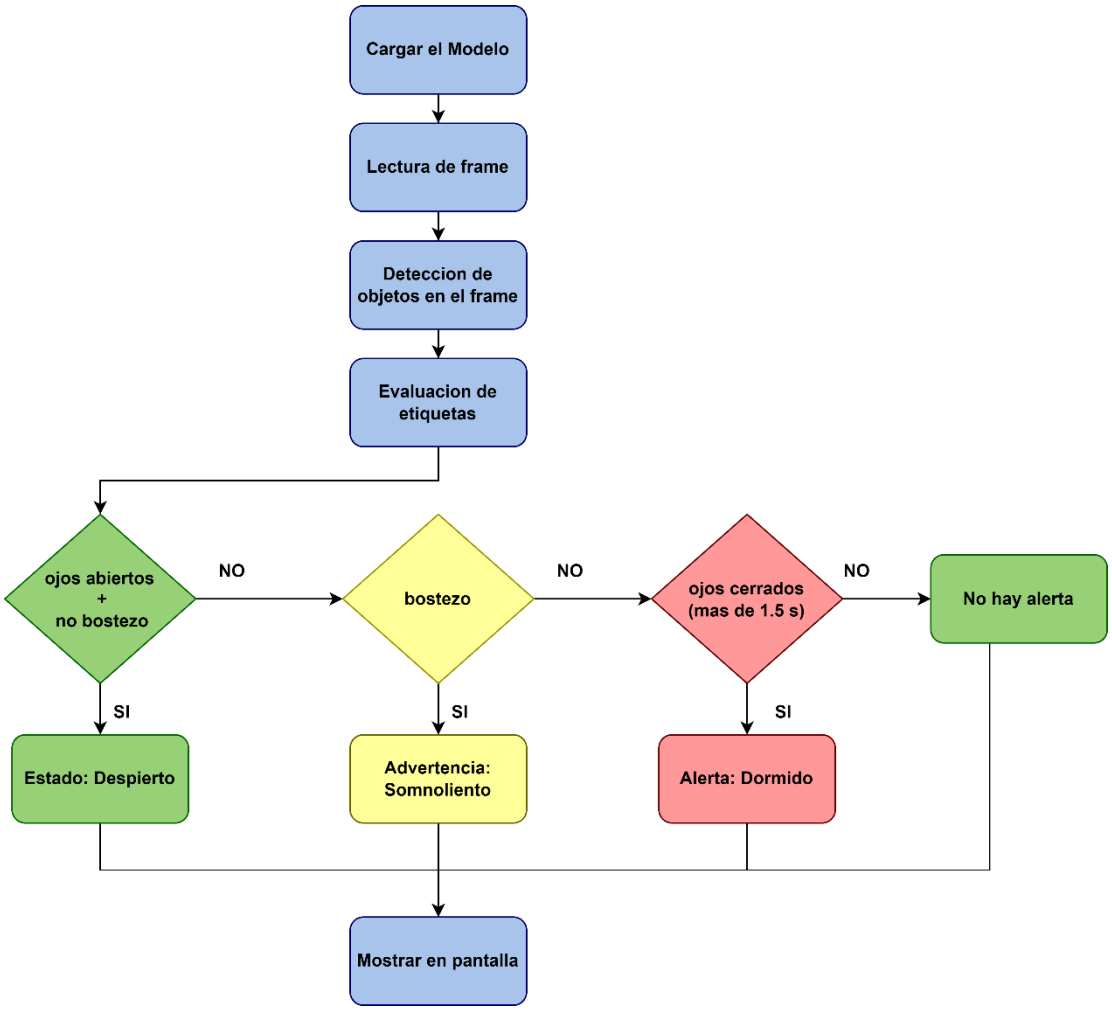


Figura 21. Algoritmo lógico de clasificación de estados de somnolencia.

El proceso inicia con la carga del modelo previamente entrenado, seguido de la lectura secuencial de cada cuadro del video. A cada frame se le aplica detección de objetos, y se recuperan las etiquetas generadas (eyes_open, eyes_closed, yawn, no_yawn, face). Estas etiquetas son evaluadas en tiempo real para determinar el estado del conductor bajo tres posibles condiciones:

- Estado despierto: Cuando se detectan ojos abiertos y no hay presencia de bostezo.
- Estado somnoliento: Si se detecta un bostezo, incluso con los ojos abiertos.
- Estado dormido: Cuando los ojos permanecen cerrados por más de 1.5 segundos consecutivos.

Cada una de estas condiciones activa una respuesta correspondiente: sin alerta, advertencia de somnolencia o alerta de sueño profundo, las cuales se muestran directamente en pantalla mediante anotaciones visuales sobre el video procesado.

Este algoritmo, aunque simple en lógica, fue eficaz para demostrar que es posible construir un sistema de monitoreo en tiempo real a partir del modelo entrenado, siendo adaptable a condiciones dinámicas del entorno vehicular.

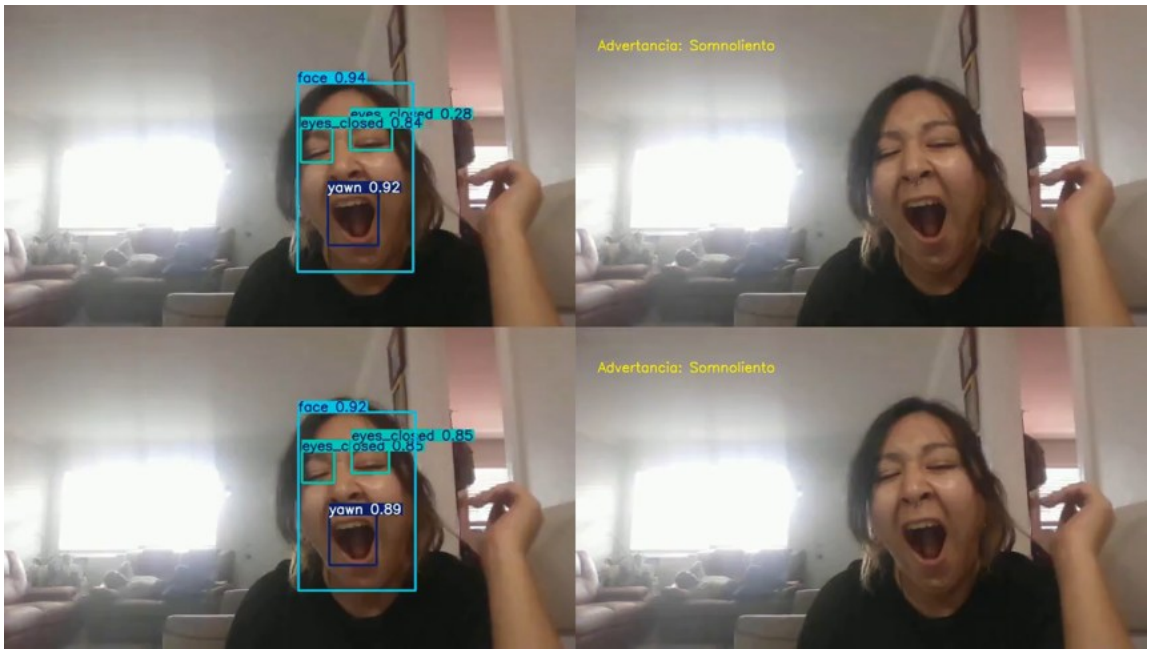


Figura 22. Detección de somnolencia en video en un entorno controlado. (URL:https://youtu.be/13nYQxdoJS8) (Miranda-Leon, 2025).

Figura 23. Detección de somnolencia al volante (URL: https://youtu.be/qjVYWPcaUyY) (Miranda-Leon, 2025).

Aunque el propósito de este trabajo no fue desarrollar un prototipo funcional final, los resultados obtenidos permiten anticipar el potencial práctico de esta tecnología. La Figura 22 muestra la detección de somnolencia en un entorno controlado mediante video, mientras que la Figura 23 representa su aplicación en un contexto vehicular real. Cabe mencionar que fue llevado a cabo bajo condiciones seguras y supervisadas para evitar cualquier tipo de riesgo.

Estas pruebas demuestran que el sistema basado en YOLOv8 puede integrarse en flujos de video en tiempo real. Manteniendo un rendimiento aceptable y una detección precisa de señales visuales de fatiga. Esta funcionalidad podría ser clave para aplicaciones futuras, especialmente en sistemas de asistencia al conductor o soluciones de seguridad vial donde la somnolencia representa un factor de riesgo crítico.

VIII. CONCLUSIONES

A partir de los experimentos realizados y del análisis exhaustivo de los resultados, se establecen las siguientes conclusiones:

1. Creación y validación de un conjunto de datos eficaz:

Se identificó y validó un conjunto de datos crudo personalizado para la detección de somnolencia. El cual fue procesado en distintas variantes y evaluado mediante diversos modelos de aprendizaje profundo. Este conjunto permitió simular condiciones realistas y probar la capacidad de generalización de los modelos. Consolidándose como un recurso sólido para tareas de visión artificial aplicadas a seguridad vial.

2. Desempeño adecuado de los modelos YOLOv8 y YOLOv10:

Ambos modelos demostraron una alta precisión en la detección de objetos faciales relevantes (ojos, rostro, bostezo), tanto en imágenes individuales como en métricas de validación cuantitativa. Su desempeño en entornos controlados sugiere que las arquitecturas YOLO son aptas para tareas de detección multiclase asociadas a somnolencia en escenarios de aplicación real.

3. YOLOv8 como modelo óptimo:

El modelo YOLOv8 entrenado con el conjunto de imágenes semiprocesadas. Allcanzó los mejores resultados en términos de precisión, recall y mAP, destacando también por su tiempo de entrenamiento. El cual fue significativamente menor frente a YOLOv10. Esta combinación de desempeño y eficiencia lo posiciona como la opción más viable para sistemas de detección en tiempo real, especialmente en contextos con restricciones de hardware.

4. Ventajas del conjunto de imágenes semiprocesado:

Se evidenció que el conjunto de imágenes semiprocesadas ofrece un equilibrio ideal entre calidad visual y carga computacional. Al conservar el formato RGB y aplicando mejoras en brillo y sombras, permitió optimizar el aprendizaje del modelo sin perder riqueza de información. Esto convierte al conjunto de imágenes semiprocesado en una opción preferente para futuras implementaciones.

5. Robustez ante variaciones en orientación facial:

El sistema fue capaz de realizar detecciones efectivas incluso cuando la persona cambiaba de ángulo o dirección, lo cual valida la utilidad de haber incluido distintos perfiles faciales en el conjunto de datos crudo. Esta robustez ante variaciones espaciales refuerza la aplicabilidad del modelo en sistemas dinámicos, como monitoreo vehicular en tiempo real.

6. Eficacia del algoritmo lógico de evaluación:

La integración del modelo entrenado con un algoritmo lógico simple, basado en etiquetas y condiciones temporales (bostezo, cierre ocular prolongado), permitió una detección funcional de somnolencia en video, validando su potencial como sistema preventivo. A pesar de su simplicidad, la lógica desarrollada fue suficiente para ofrecer advertencias y alertas de forma precisa y contextualizada.

Si bien el conjunto de datos crudo desarrollado en este trabajo resultó efectivo para entrenar modelos robustos y generalizables. Se identifican áreas de mejora que podrían incrementar su capacidad descriptiva. Una de ellas es la incorporación de mayor complejidad postural. Integrando variaciones de orientación como mirada hacia arriba, abajo o posición neutral, además de los perfiles ya considerados (frontal, lateral izquierdo y derecho). Esta ampliación permitiría enriquecer la diversidad de condiciones faciales simuladas, lo cual podría traducirse en una mayor tolerancia del modelo ante variaciones naturales del comportamiento humano. Particularmente en contextos como el manejo prolongado, donde el rostro del conductor cambia constantemente de posición.

7. Líneas de mejora y proyecciones futuras:

Aunque el conjunto de datos desarrollado demostró ser funcional y bien estructurado. Su capacidad aún podría potenciarse mediante la incorporación de mayor complejidad postural, integrando imágenes dentro del conjunto de imágenes de entrenamiento con orientación hacia arriba, hacia abajo y en posición neutral, además de los perfiles ya considerados. Esta ampliación enriquecería la representación de condiciones reales y contribuiría a una mayor robustez del modelo. Asimismo, el algoritmo lógico empleado para la detección de somnolencia en video, si bien resultó funcional, podría evolucionar mediante el uso de lógica difusa. De manera que permitiese una toma de decisiones más flexible y sensible a transiciones sutiles entre estados de alerta y fatiga. No obstante, esta propuesta queda como una línea de trabajo futura que complementaría los avances aquí presentados.

IX. REFERENCIAS

Alcántara-Montiel, C. V., Pedraza-Ortega, J. C., Ramos-Arreguín, J. M., Gorrostieta-Hurtado, E., Tovar-Arriaga, S., & Vargas-Soto, J. E. (2019). Detección efectiva de rostros en imágenes utilizando descriptores basados en HOG. *Research in Computing Science*, *148*(8), 371–385. https://doi.org/10.13053/rcs-148-8-28

Alparslan, Y., & Kim, E. (2021). *Robust SleepNets*. https://arxiv.org/abs/2102.12555

Angelov, P. P., Soares, E. A., Jiang, R., Arnold, N. I., & Atkinson, P. M. (2021). Explainable artificial intelligence: an analytical review. *WIREs Data Mining and Knowledge Discovery*, *11*(5). https://doi.org/10.1002/widm.1424

Anrango Farinango, E. X. (2022). *Sistema detector de fatiga electrocardiográfico para prevenir el estado de somnolencia en conductores de vehículos*. Universidad Técnica del Norte.

Benítez Rojas, R. V. (2023). *Algoritmos, IA y automatización en comunicación. Prólogo*. Universidad Miguel Hernández de Elche. https://hdl.handle.net/11000/36716

Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). *YOLOv4: Optimal Speed and Accuracy of Object Detection*.

Montiel Alcántara, C. V. (2018). *Sistema embebido de bajo costo para la asistencia al conductor mediante procesamiento de expresiones faciales* [Universidad Autónoma de Querétaro, Facultad de Ingeniería]. https://ri-ng.uaq.mx/handle/123456789/1254

Carrillo-Mora, P., Ramírez-Peris, J., & Magaña-Vázquez, K. (2013). Neurobiología del sueño y su importancia: antología para el estudiante universitario. *Revista de la Facultad de Medicina (México)*, *56*(4), 5–15. http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S0026-17422013000400002&lng=es&nrm=iso

Cluydts, R., De Valck, E., Verstraeten, E., & Theys, P. (2002). Daytime sleepiness and its evaluation. *Sleep Medicine Reviews*, *6*(2), 83–96. https://doi.org/10.1053/smrv.2002.0191

Cuevas-Colunga, A. C., Cadengo-Ramírez, M., Silva-Rivera, M. E., & Mendoza-Díaz, A. (2022). *Anuario estadístico de colisiones en carreteras federales 2021* .

Cuevas-Colunga, A. C., & Silva Rivera, M. E. (2024). *Anuario estadístico de colisiones en carreteras federales, 2023*.

Cui, J., Lan, Z., Liu, Y., Li, R., Li, F., Sourina, O., & Müller-Wittig, W. (2022). A compact and interpretable convolutional neural network for cross-subject driver drowsiness detection from single-channel EEG. *Methods*, *202*, 173–184. https://doi.org/10.1016/j.ymeth.2021.04.017

Developers, G. (2025). *Accuracy, Precision, Recall: Classification Metrics — Machine Learning Crash Course*. https://developers.google.com/machine-learning/crash-course/classification/accuracy-precision-recall?hl=es-419

developers, S. (2025). *f1_score — scikit-learn 1.6.1 documentation*. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html

Du, L., Zhang, R., & Wang, X. (2020). Overview of two-stage object detection algorithms. *Journal of Physics: Conference Series*, *1544*(1), 012033. https://doi.org/10.1088/1742-6596/1544/1/012033

Dutta, G. (2022). *Detect-Yawning*. Kaggle. https://www.kaggle.com/datasets/gauravduttakiit/detectyawning

EQUIPO SITEC. (2023). *Sistemas ADAS en México – SITEC – Soluciones e Integraciones Tecnológicas*. https://sitec.mx/sistemas-adas-en-mexico/

Fernández Carrión, L. A. (2021). *Sistema de detección de somnolencia en conductores mediante técnicas de visión por computadora* [Universidad La Salle - Arequipa]. https://repositorio.ulasalle.edu.pe/handle/20.500.12953/122

García Daza, I. (2011). *Detección de fatiga en conductores mediante fusión de sistemas ADAS*. UNIVERSIDAD DE ALCALÁ ESCUELA POLITÉCNICA SUPERIOR.

Gershenson, C. (2003). *Artificial Neural Networks for Beginners*. https://arxiv.org/abs/cs/0308031

Gholamalinezhad, H., & Khosravi, H. (2020). *Pooling Methods in Deep Neural Networks, a Review*. https://arxiv.org/abs/2009.07485

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

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

Guo, T., Dong, J., Li, H., & Gao, Y. (2017). Simple convolutional neural network on image classification. *2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA)(*, 721–724. https://doi.org/10.1109/ICBDA.2017.8078730

Instituto Peruano de Neurociencias (IPN). (2017). *Test de Latencia Múltiple del Sueño*. https://ipn.pe/servicios/test-de-latencia-multiple-del-sueno/

Jocher, G., Chaurasia, A., & Qiu, J. (2023). *Ultralytics YOLOv8*. https://github.com/ultralytics/ultralytics

Kepesiova, Z., Ciganek, J., & Kozak, S. (2020). Driver Drowsiness Detection Using Convolutional Neural Networks. *2020 Cybernetics & Informatics (K&I)*, 1–6. https://doi.org/10.1109/KI48306.2020.9039851

Koonce, B. (2021). ResNet 50. En *Convolutional Neural Networks with Swift for Tensorflow* (pp. 63–72). Apress. https://doi.org/10.1007/978-1-4842-6168-2_6

Leon, M. L. M. (2025). *Detección de Somnolencia utilizando YOLOv8*. https://www.youtube.com/watch?v=13nYQxdoJS8

Li, W. (2021). Analysis of Object Detection Performance Based on Faster R-CNN. *Journal of Physics: Conference Series*, *1827*(1), 012085. https://doi.org/10.1088/1742-6596/1827/1/012085

Liner.ai. (s/f). *Liner.ai: Train Object Detection Models Without Code*. Recuperado el 30 de junio de 2025, de https://liner.ai/

Lucendo, J. (2019). *Las Edades del Automóvil: Historia del Automóvil*. Jorge Lucendo.

MathWorks. (2025a). *Convert RGB colors to HSV*. https://la.mathworks.com/help/matlab/ref/rgb2hsv.html

MathWorks. (2025b). *Convert RGB image or colormap to grayscale*. https://la.mathworks.com/help/matlab/ref/rgb2gray.html

MathWorks. (2025c). *Gaussian smoothing filter*. https://la.mathworks.com/help/images/ref/imgaussfilt.html

Microsoft. (2020). *Visual Object Tagging Tool (VoTT), Release v2.2.0*. https://github.com/microsoft/VoTT/releases/tag/v2.2.0

Miranda-Leon, M. L. (2025). *Detección de somnolencia al volante (con algoritmo lógico y YOLOv8)*. https://youtu.be/qjVYWPcaUyY

MIRANDA-LEON, M. L. (2025). *DrowsyFace Profiles*. Zenodo. https://doi.org/10.5281/zenodo.14805037

Miranda-Leon, M. L., & Angole-Tierrablanca, J. A. (2024). *Drowsiness Image Dataset*. https://www.kaggle.com/datasets/mariamiranda98/drowsiness-image-dataset

Miranda-Leon, M. L., Pedraza-Ortega, J. C., Aceves-Fernandez, M. A., & Ramos-Arreguin, J. M. (2024). A Comparative Analysis of Object Detection Models for Drowsiness Detection: Evaluating Raw, Processed, and Hybrid Image Datasets with Faster R-CNN. *2024 21st International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE)*, 1–6. https://doi.org/10.1109/CCE62852.2024.10771046

Muñoz, J. E. B., Castillo, J. A. Á., & Gastélum-Barrios, A. (2024). La base de la autonomía vehicular: los sistemas ADAS. *PCT*, *7*(13), 166–184. https://doi.org/10.61820/pct.v7i13.1361

Ortiz, Ó. M., Mora, N. S., Galindo, J. C., Herráez, D. F., & López, C. A. (2007). Alteraciones del sueño en los trastornos psiquiátricos. *Revista Colombiana de Psiquiatría*, *36*(4),

701–717. http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S0034-74502007000400009&lng=en&tlng=es

Patil, P. V. (2025). *MRL Dataset*. https://www.kaggle.com/datasets/prasadvpatil/mrl-dataset

Perumandla, D. (2025). *Drowsiness Dataset*. https://www.kaggle.com/datasets/dheerajperumandla/drowsiness-dataset

PyTorch. (s/f). *Faster R-CNN ResNet50 FPN*. Recuperado el 23 de julio de 2024, de https://pytorch.org/vision/stable/models/generated/torchvision.models.detection.fasterrcnn_resnet50_fpn.html

Ramírez-Arriaga, K. A. (2023). *Sistema de reconocimiento de expresiones faciales para detección de estados de somnolencia con imágenes nocturnas*. Univerisdad Autonoma de Queretaro.

Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. *arXiv preprint arXiv:1804.02767*.

Rosales Mayor, E., & Rey De Castro Mujica, J. (2010). Somnolencia: Qué es, qué la causa y cómo se mide. *Acta Médica Peruana*, *27*(2), 137–143. http://www.scielo.org.pe/scielo.php?script=sci_arttext&pid=S1728-59172010000200010&lng=es&nrm=iso&tlng=es

Satapathy, S. C., Raju, K. S., Shyamala, K., Krishna, D. R., & Favorskaya, M. N. (2020). Advances in decision sciences, image processing, security and computer vision. *International Conference on Emerging Trends in Engineering (ICETE)*, *1*, 1.

Shen, J., Barbera, J., & Shapiro, C. M. (2006). Distinguishing sleepiness and fatigue: focus on definition and measurement. *Sleep Medicine Reviews*, *10*(1), 63–76. https://doi.org/10.1016/j.smrv.2005.05.004

Singh, J., Kanojia, R., Singh, R., Bansal, R., & Bansal, S. (2023). *Driver Drowsiness Detection System: An Approach By Machine Learning Application*. https://arxiv.org/abs/2303.06310

Szeliski, R. (2010). *Computer Vision: Algorithms and Applications*. Springer. http://szeliski.org/Book/

Terán-Santos, J., Jimenez-Gomez, A., & Cordero-Guevara, J. (1999). The Association between Sleep Apnea and the Risk of Traffic Accidents. *New England Journal of Medicine*, *340*(11), 847–851. https://doi.org/10.1056/NEJM199903183401104

Ultralytics. (2025). *YOLO Métricas de rendimiento*. https://docs.ultralytics.com/es/guides/yolo-performance-metrics/#choosing-the-right-metrics

Wiley, V., & Lucas, T. (2018). Computer Vision and Image Processing: A Paper Review. *International Journal of Artificial Intelligence Research*, *2*(1), 22. https://doi.org/10.29099/ijair.v2i1.42

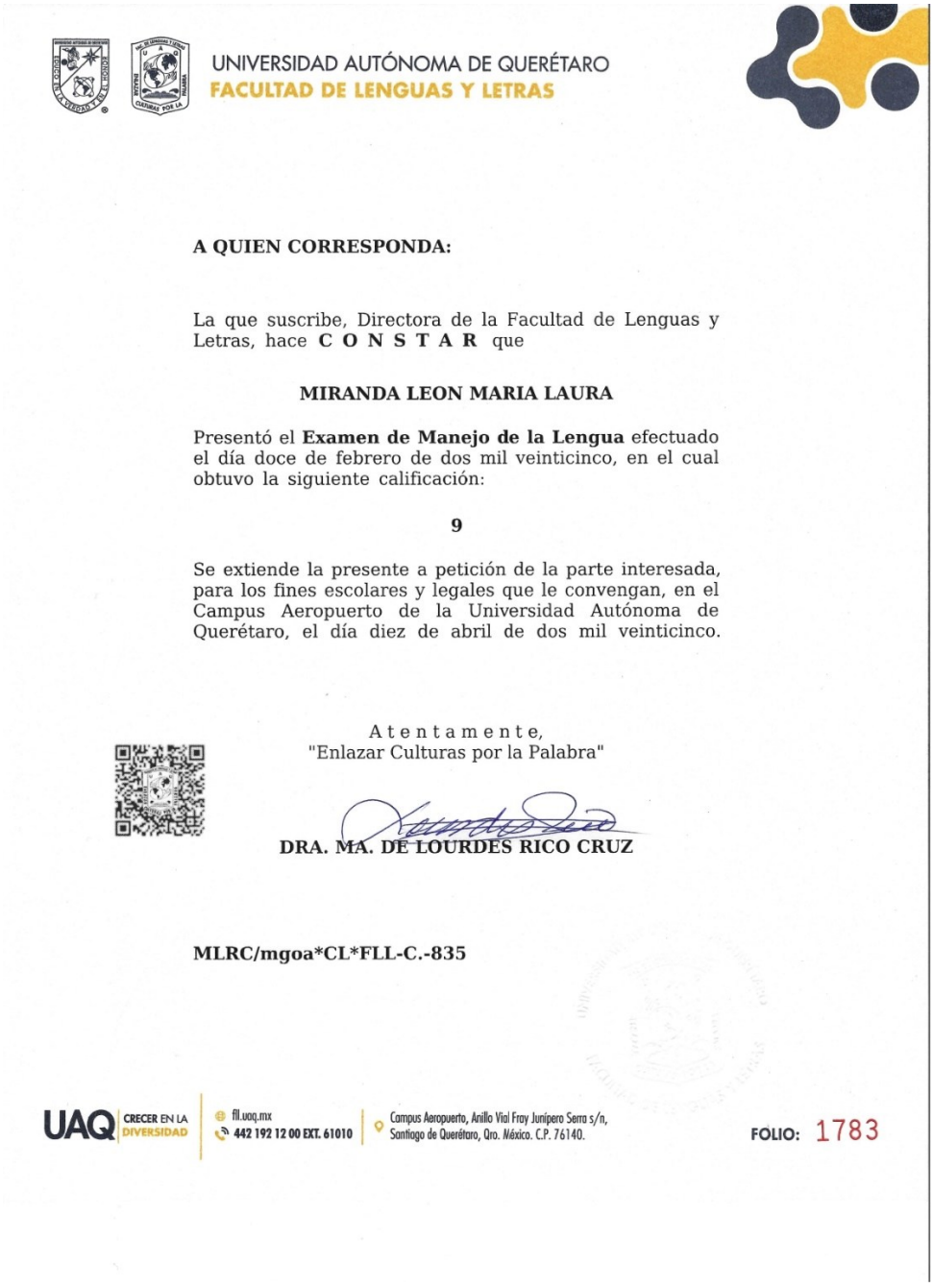
Zhao, L., & Zhang, Z. (2024). A improved pooling method for convolutional neural networks. *Scientific Reports*, *14*(1), 1589. https://doi.org/10.1038/s41598-024-51258-6

Zhao, X., Wang, L., Zhang, Y., Han, X., Deveci, M., & Parmar, M. (2024). A review of convolutional neural networks in computer vision. *Artificial Intelligence Review*, *57*(4), 99. https://doi.org/10.1007/s10462-024-10721-6

Zhu, L., Lee, F., Cai, J., Yu, H., & Chen, Q. (2022). An improved feature pyramid network for object detection. *Neurocomputing*, *483*, 127–139. https://doi.org/10.1016/j.neucom.2022.02.016

X. ANEXOS

Anexo A



UNIVERSIDAD AUTÓNOMA DE QUERÉTARO
FACULTAD DE LENGUAS Y LETRAS

A QUIEN CORRESPONDA:

La que suscribe, Directora de la Facultad de Lenguas y Letras, hace **C O N S T A R** que

MIRANDA LEON MARIA LAURA

Presentó el **Examen de Manejo de la Lengua** efectuado el día doce de febrero de dos mil veinticinco, en el cual obtuvo la siguiente calificación:

9

Se extiende la presente a petición de la parte interesada, para los fines escolares y legales que le convengan, en el Campus Aeropuerto de la Universidad Autónoma de Querétaro, el día diez de abril de dos mil veinticinco.

A t e n t a m e n t e,
"Enlazar Culturas por la Palabra"

DRA. MA. DE LOURDES RICO CRUZ

MLRC/mgoa*CL*FLL-C.-835

UAQ CRECER EN LA DIVERSIDAD
fll.uaq.mx
442 192 12 00 EXT. 61010
Campus Aeropuerto, Anillo Vial Fray Junípero Serra s/n, Santiago de Querétaro, Qro. México. C.P. 76140.

FOLIO: 1783

Figura 24. Constancia de cumplimiento del requisito de ingles del Examen de Manejo de la Lengua emitida por la Facultad de Lengua y Letras de la Universidad Autónoma de Querétaro.

Anexo B

UNIVERSIDAD AUTÓNOMA DE QUERÉTARO
FACULTAD DE LENGUAS Y LETRAS

A QUIEN CORRESPONDA:

La que suscribe, Directora de la Facultad de Lenguas y Letras, hace **C O N S T A R** que

MIRANDA LEON MARIA LAURA

Presentó y acreditó el **Examen de Comprensión de Textos en Inglés** efectuado el día cuatro de diciembre de dos mil veinticuatro.

Se extiende la presente a petición de la parte interesada, para los fines escolares y legales que le convengan, en el Campus Aeropuerto de la Universidad Autónoma de Querétaro, el día diez de abril de dos mil veinticinco.

A t e n t a m e n t e,
"Enlazar Culturas por la Palabra"

DRA. MA. DE LOURDES RICO CRUZ

MLRC/mgoa*CL*FLL-C.-834

UAQ CRECER EN LA DIVERSIDAD
fll.uaq.mx
442 192 12 00 EXT. 61010
Campus Aeropuerto, Anillo Vial Fray Junípero Serra s/n, Santiago de Querétaro, Qro. México. C.P. 76140.

FOLIO: 1753

Figura 25. Constancia de cumplimiento del requisito de ingles del Comprensión de Textos en Ingles emitida por la Facultad de Lengua y Letras de la Universidad Autónoma de Querétaro.

Anexo C

A Comparative Analysis of Object Detection Models for Drowsiness Detection: Evaluating Raw, Processed, and Hybrid Image Datasets With Faster R-CNN

Maria Laura Miranda-Leon
Faculty of Engineering
(Autonomous University of Queretaro)
Santiago de Queretaro, Queretaro
miranda.maria2017@gmail.com

Jesus Carlos Pedraza-Ortega
Faculty of Engineering
(Autonomous University of Queretaro)
Santiago de Queretaro, Queretaro
caryoko@yahoo.com

Marco Antonio Aceves-Fernandez
Faculty of Engineering
(Autonomous University of Queretaro)
Santiago de Queretaro, Queretaro
marco.aceves@gmail.com

Juan Manuel Ramos-Arreguin
Faculty of Engineering
(Autonomous University of Queretaro)
Santiago de Queretaro, Queretaro
jsistdig@yahoo.com.mx

***Abstract*— In Mexico, last year, 5.10% of accidents attributed to human factors were related to drowsiness. Consequently, this study investigates the detection of objects to identify visible signs of drowsiness, such as the eyes and mouth, employing a Faster R-CNN model with ResNet50 and FPN. A comparative analysis uses three datasets: raw images, processed images, and a hybrid set comprising both. The results indicate that although the model trained on raw images exhibited the lowest loss and superior performance for certain classes, the model trained on processed images demonstrated the highest efficiency in training time. Meanwhile, the model trained on the hybrid set achieved a balanced performance, integrating the strengths of both preceding models.**

Keywords*— *Image Processing, Convolutional Neural Networks, Faster R-CNN, Datasets, Raw Data.

I. Introduction

Road safety is a critical global concern, with driver drowsiness significantly contributing to traffic accidents. According to the 2024 Federal Highway Crash Statistical Yearbook in 2023, 12,099 incidents contributed to vehicle accidents, of which 4,599 were due to human factors and 235 were because the driver fell asleep [1]. This means that 5.10% of the accidents caused by human factors were due to drowsiness.

From 2022 to 2024 [2][3] Various reviews identified various methods for detecting drowsiness, including visual, and biological approaches, as well as techniques based on vehicle behavior or hybrid systems that combine multiple detection methods. A common technique in artificial intelligence, Convolutional Neural Networks (CNNs) are widely used to solve these problems. Previous research has applied CNNs to driver drowsiness detection. For example, a study by Kepesiova and colleagues in 2020 collected data from 20 subjects, capturing eight behavioral patterns in 5-second video sequences, categorized as either "aware" or "sleepy." Each frame was processed to a resolution of 224x224 pixels in greyscale. The final model, utilizing the VGG Face architecture, a CNN specifically trained on human facial features, attained an accuracy of 90.77% on the training dataset and 98.02% on the validation dataset [4]. However, the model faced challenges in generalizing to new individuals, highlighting the need for further refinement.

Today, the evolution of image processing techniques has played a crucial role in detection methods such as drowsiness detection. Though seemingly modern, the concept of "image processing" has historical roots extending back to the creation of physical images such as paintings, which occasionally required retouching—early forms of aesthetic enhancement. The advent of photography in 1826 by Joseph Nicéphore Niépce marked a significant development [5]. Photographers subsequently adapted their methods to meet aesthetic and technical needs, including collaboration with miniature painters to augment daguerreotypes and calotypes [6] . The evolution continued with the emergence of digital imagery in the late 1950s, which led to digital image processing. This technology was initially employed for electronic manipulation techniques to enhance images captured by space probes.

Despite significant technological advancements and the availability of high-resolution cameras, image processing remains a critical field of study. For instance, the 2019 image of a black hole exemplifies the importance of image processing. The reconstruction of this image employed sophisticated image processing techniques and artificial intelligence, notably Principal Component Interferometric Modeling [7]. A prominent application of image processing is in the medical domain, particularly within medical imaging. Techniques such as Computed Tomography (CT), Magnetic Resonance Imaging (MRI), ultrasound imaging, and mammography are essential for diagnostic purposes. Furthermore, Computer-assisted surgery leverages processed medical images to assist surgeons in planning and performing procedures with millimeter-level accuracy. This includes applications within Advanced Driver Assistance Systems (ADAS), such as driver drowsiness detection, which is the focus of this study.

979-8-3503-7754-5/24/$31.00 ©2024 IEEE

Figura 26. Primera página del artículo publicado por parte del 2024 21st International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE).

Anexo D

Figura 27. Constancia de presentación de articulo en el 2024 21st International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE).

Anexo E

FACULTAD DE INGENIERÍA
DIVISIÓN DE INVESTIGACIÓN Y POSGRADO

Santiago de Querétaro a 20 de julio de 2025

A quien corresponda
P r e s e n t e.

En cumplimiento a lo establecido en el *Artículo 20, Capítulo VII. De la Conclusión de la Beca o Apoyo*, del *Reglamento de Becas de la Secretaría de Ciencia, Humanidades Tecnología e Innovación*, y en el marco de la Convocatoria **Becas nacionales para estudios de posgrado 2023-2**, hago constar que **la C. Maria Laura Miranda Leon** con número de **CVU 1313219** beneficiada con una beca para obtener el grado de **Maestría** en el programa **005351 - Maestría en Ciencias en Inteligencia Artificial**, que se imparte en **Universidad Autónoma de Querétaro – Campus Aeropuerto**, realizó las actividades de retribución social durante el periodo de vigencia de la beca tiempo en el que fue **alumna** regular de esta Institución.

Asimismo, hago constar que, conforme a lo establecido en la Ley General de Archivos, la coordinación del posgrado organiza y conserva la evidencia documental de dichas actividades en caso de que la SECIHTI o cualquier otra instancia la requiera.

Sin más por el momento, le envío un cordial saludo.

Dr. Saúl Tovar Arriaga
Coordinador del programa Maestría en Ciencias en Inteligencia Artificial

UNIVERSIDAD AUTÓNOMA DE QUERÉTARO
CERRO DE LAS CAMPANAS S/N, COL. LAS CAMPANAS
FACULTAD DE INGENIERÍA, EDIFICIO G.
C.P. 76010 QUERÉTARO, QRO. TEL. 192 12 00 EXT. 6023

Figura 28. Constancia de realización de retribución social.