UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS

FACULTAD DE CONTADURÍA, ADMINISTRACIÓN E INFORMÁTICA
MAESTRÍA EN OPTIMIZACIÓN Y CÓMPUTO APLICADO

# CONTEO DE ESTUDIANTES EN AMBIENTES EDUCATIVOS USANDO CNNs PRE ENTRENADAS

**T E S I S**

QUE PARA OBTENER EL GRADO DE

MAESTRÍA EN OPTIMIZACIÓN Y CÓMPUTO APLICADO

PRESENTA

**LIZETH MEZA ALEGRIA**

DIRECTOR DE TESIS

DR. JOSÉ ALBERTO HERNÁNDEZ AGUILAR

CO-DIRECTOR

DRA. MARÍA YASMÍN HERNÁNDEZ PÉREZ

REVISORES:

DR. JOSÉ ALBERTO HERNÁNDEZ AGUILAR
DRA. MARÍA YASMÍN HERNÁNDEZ PÉREZ
DR. PEDRO MORENO BERNAL
DR. VÍCTOR HUGO PACHECO VALENCIA
DR. OUTMANE OUBRAM

CUERNAVACA, MORELOS. JUNIO, 2026

Facultad de Contaduría, Administración e Informática

**COMITÉ REVISOR**

**Dr. José Alberto Hernández Aguilar, FCAeI**

**Dra. María Yasmín Hernández Pérez, CENIDET**

**Dr. Outmane Oubram, FCQeI**

**Dr. Pedro Moreno Bernal, FCAeI**

**Dr. Víctor Hugo Pacheco Valencia, CIICAp**

# Lista de publicaciones relacionadas con la tesis

1. Alegría, L. M., Hernandez-Aguilar, J. A., Pérez, M. y. H., Bernal, P. M., Oubram, O., Pacheco-Valencia, V., & Gallegos, J. C. P. (2026). Counting of Students in Educational Environments Using Pretrained Convolutional Neural Networks. En *Lecture notes in computer science* (pp. 70-81). https://doi.org/10.1007/978-3-032-17933-3_8
2. Meza-Alegría, L (2026), School Environments Mexico Dataset. GitHub. https://github.com/LixxMeza/School-environments-M-xico

# Resumen

La presente investigación aborda el desafío de detectar y contar personas en espacios educativos cerrados como herramienta que ayuda a la toma de decisiones institucionales tras el retorno a las actividades presenciales en la Facultad de Contaduría, Administración e Informática (FCAeI) de la Universidad Autónoma del Estado de Morelos (UAEM). El objetivo principal consiste en implementar un sistema basado en VC (visión por computadora) utilizando Redes Neuronales Convolucionales (CNN) pre-entrenadas para que este realice el conteo de estudiantes en espacios educativos cerrados.

La metodología consiste en el entrenamiento y validación de la arquitectura YOLOv8 (You Only Look Once, versión 8) en sus variantes *nano* (pequeña) y *large* (grande), utilizando un conjunto de datos híbrido que integró el "classroom-monitoring-dataset" con imágenes reales capturadas en salones, laboratorios y el auditorio de la facultad. Para robustecer la detección ante oclusiones, condiciones de luz,lucondicion ángulo y otras condiciones de reto,con complejos, se adoptó la estrategia de etiquetar únicamente las cabezas de los estudiantes. El procesamiento de datos se realizó mediante Roboflow, aplicando técnicas de aumentación como rotación y filtros de escala de grises.

Los resultados demostraron que el modelo YOLOv8l entrenado con 100 épocas obtuvo el mejor rendimiento, alcanzando una precisión del 90.2% y un mAP50 (Precisión Media Promedio con un umbral de IoU del 50 %) del 92.1%, superando el objetivo inicial de efectividad del 90%. Con esto, se validó la hipótesis de que es posible realizar un conteo eficiente en entornos reales, proporcionando una herramienta tecnológica avanzada para la planificación educativa y la seguridad en el campus.

# Abstract

This research addresses the challenge of people counting within enclosed educational spaces to optimize occupancy management and institutional decision-making following the resumption of face-to-face activities at the Faculty of Accounting, Administration, and Informatics (FCAeI) of the UAEM. The main objective was to implement a computer vision system based on pre-trained Convolutional Neural Networks (CNN).

The methodology involved training and validating the YOLOv8 architecture in its *nano* and *large* variants, utilizing a hybrid dataset that integrated the "classroom-monitoring-dataset" with real-world images captured in classrooms, laboratories, and the faculty's auditorium. To enhance detection robustness against occlusions and complex camera angles, the strategy of labeling only the students' heads was adopted. Data processing was conducted via Roboflow, applying augmentation techniques such as rotation and grayscale filters.

The results demonstrated that the YOLOv8l model trained for 100 epochs achieved the highest performance, reaching a precision of 90.2% and an mAP50 of 92.1%, thereby exceeding the initial effectiveness target of 90%. These findings validated the hypothesis that efficient counting is achievable in real-world environments, providing an advanced technological tool for educational planning and campus safety.

# Agradecimientos

Al Consejo Nacional de Humanidades, Ciencias y Tecnologías (CONAHCYT), por el apoyo financiero otorgado, el cual fue vital para poder dedicarme a mis estudios e investigación.

A mis directores de tesis, Dr. José Alberto Hernández Aguilar y Dra. Yasmín Hernández Pérez, por su invaluable guía, paciencia, sabiduría y apoyo a lo largo de esta investigación. Sus conocimientos y experiencia fueron pieza clave para la culminación de esta investigación.

A mi familia y amigos, por los consejos, pláticas, ánimo y apoyo moral.

A mi hijo Gabriel, que es la luz de mi camino y mi mayor motivación para seguir adelante.

# Contenido

## Lista de figuras

**Lista de tablas**

# CAPÍTULO I INTRODUCCIÓN

La Inteligencia ArtificialArtifiIntelI (IA) es una disciplina en las Ciencias de la Computación que se utiliza para resolver una amplia gama de problemas incluyendo la automatización de tareas repetitivas, el análisis de datos avanzados, y la mejora de atención al cliente (Goodfellow et al., 2016). Una ventaja fundamental de la IA es que no hay necesidad de programar un algoritmo específico para cada escenario imaginable, se le puede "enseñar" a través de datos (Aprendizaje Automático o Machine Learning) o permitir que aprenda por sí misma a partir de grandes volúmenes de información (Deep Learning). Existen variantes de cada uno, pero se puede definir de manera general (Alonso, 2023).

Dentro de las múltiples ramas de la IA, la VC (V ha emergido como un campo de alto impacto, permitiendo a las máquinas "ver" e interpretar el mundo visual. Esta disciplina se enfoca en el procesamiento de imágenes mediante algoritmos estratégicos diseñados para cumplir determinados propósitos. Algunas de las técnicas de la visión por computadora son: clasificación de imágenes, detección de objetos en una imagen y seguimiento de objetos (Nalbant, 2021).

La detección de objetos es una herramienta particularmente útil y versátil dentro de la visión por computadora, sirviendo como base para tareas más complejas como la segmentación de instancias (delinear cada objeto individualmente) y la generación de descripciones textuales de imágenes (subtítulos de imágenes) (Sandoval, 2021). Un área específica de la detección de objetos es la detección de rostros, una tecnología comúnmente empleada en cámaras digitales para mejorar el enfoque automático o facilitar la búsqueda de personas en colecciones de imágenes (Briega, 2018).

Si bien para un ser humano con capacidad visual la tarea de distinguir un objeto en una imagen es, en la mayoría de los casos trivial, para una computadora representa un desafío considerable. Cada objeto por detectar requiere considerar múltiples aspectos y, a menudo, recurrir a diferentes modelos computacionales o enfoques. En este trabajo de tesis se centra

específicamente en la detección de cabezas como un paso fundamental hacia el conteo de personas en un contexto particular: los ambientes educativos cerrados.

El conteo de personas adquiere una importancia significativa cuando se busca tener certeza sobre la permanencia de individuos en un lugar, la frecuencia con la que salen o entran a un lugar, los días de mayor o menor afluencia y proporcionar información sobre los patrones de tráfico. Todos estos datos, a su vez, ofrecen información valiosa sobre el comportamiento de las personas en dichos espacios (Getin, 2023).

## 1.1 Problema de investigación

El retorno gradual a las actividades presenciales tras la pandemia iniciada en 2020, incluyendo la vuelta a las aulas, ha puesto de relieve la importancia de gestionar el aforo en espacios cerrados y de contar con mecanismos eficientes para el conteo de personas. El conteo de estudiantes no solo nos sirve para automatizar y digitalizar el proceso, sino también para detectar cualquier variación importante en el número de los asistentes a clases, ya sea por una contingencia social o ambiental, como una marcha, cierre de vialidades, un brote de alguna enfermedad o bien un desastre natural.

Este trabajo de investigación aborda el problema de detectar y contar estudiantes en ambientes educativos presenciales utilizando técnicas de VC, específicamente Redes Neuronales Convolucionales (CNNs).

## 1.2 Justificación

La implementación de un modelo para el conteo de estudiantes en entornos educativos mediante VC y CNNs se justifica por múltiples razones que abarcan beneficios institucionales, contribuciones científicas y relevancia social.

Desde una perspectiva institucional, para la Facultad de Contaduría, Administración e Informática de la UAEM, este modelo permite obtener datos cuantitativos fiables sobre la asistencia y la ocupación de sus espacios. Esta información es crucial para una toma de decisiones basada en evidencia, lo que permite optimizar la asignación de recursos como aulas, laboratorios y personal docente. Además, el análisis de patrones de asistencia puede

coadyuvar al diseño e implementación de estrategias pedagógicas más efectivas, identificar posibles problemas de ausentismo y, en un sentido más amplio, mejorar la planificación educativa.

En el ámbito científico, esta investigación contribuye a la aplicación y validación de arquitecturas de CNNs, como YOLOv8, en el contexto específico del conteo de estudiantes. Aborda desafíos inherentes a los entornos educativos reales, tales como la variabilidad en las condiciones de iluminación, la presencia de oclusiones parciales y las diferentes densidades de ocupación. Los resultados y la metodología pueden servir como referencia para futuros trabajos en áreas similares, aportando al conocimiento sobre la adaptabilidad y el rendimiento de estos modelos en escenarios prácticos.

## 1.3 Objetivos

Para abordar el problema de investigación planteado, se definieron un objetivo general y varios objetivos específicos que guiaron el desarrollo de este trabajo.

**Objetivo General**

Implementar un algoritmo con el propósito de hacer un conteo de estudiantes en espacios educativos presenciales mediante VC y redes neuronales convolucionales (CNNs).

**Objetivos específicos**

- Obtener diferentes conjuntos de datos (datasets) usados en la literatura para el conteo de personas en espacios cerrados, para entrenar y evaluar la arquitectura de CNN propuesta.
- Implementar una arquitectura basada en un modelo de CNN y adaptarla específicamente para la tarea de conteo de estudiantes en imágenes de video.
- Desarrollar un modelo que permita detectar y contar a los estudiantes, y proceder a su validación y evaluación utilizando métricas estándar.
- Adaptar la arquitectura propuesta para su funcionamiento en entornos reales, considerando las posibles restricciones de hardware.

- 

## 1.4 Hipótesis

Para guiar la investigación y permitir una evaluación formal de los resultados, se plantearon las siguientes hipótesis:

Hipótesis nula (H0): Se puede detectar y contar de manera eficiente el número de estudiantes en un aula, utilizando CNNs preentrenadas en imágenes captadas por cámaras de video.

Hipótesis alternativa (H1): No se puede detectar y contar de manera eficiente el número de estudiantes en un aula, utilizando CNNs preentrenadas en imágenes captadas por cámaras de video.

La verificación de estas hipótesis se realizará en el capítulo de resultados, basándose en el rendimiento del modelo desarrollado.

## 1.5 Alcances y limitaciones

Es importante definir con claridad lo que se espera lograr con esta investigación, así como reconocer las restricciones inherentes al proyecto.

Alcances:

1. Detección de estudiantes en diversos espacios educativos de la FCAeI, incluyendo.
    - Salones de clase.
    - Auditorio.
    - Laboratorios de Cómputo
2. Lograr métricas de detección y conteo con una efectividad de al menos un 90% en las métricas de evaluación seleccionadas.
3. Generar una conjunto de datos con imágenes de estudiantes en espacios educativos cerrados para su análisis posterior.

Limitaciones:

1. La precisión en la detección de estudiantes puede verse afectada en condiciones de reto tales como:
    - Salones con iluminación insuficiente o con variaciones drásticas de la iluminación.

- Imágenes de baja resolución provenientes de las cámaras.
- Oclusiones parciales o totales de los estudiantes (uso de gorras, lentes oscuros, sombreros, o estudiantes parcialmente ocultos por otros compañeros u objetos).

2. Existen restricciones de infraestructura que pueden impactar al modelo:

- Dependencia de la disponibilidad y calidad de las cámaras utilizadas en los espacios educativos.
- Limitaciones del hardware disponible para el procesamiento y análisis de imágenes.

Estas limitaciones son importantes consideraciones para la implementación práctica y sugieren áreas en las que futuras investigaciones podrían enfocarse para mejorar la robustez y la aplicabilidad del sistema.

## 1.6 Estructura de la Tesis.

Este documento de tesis está compuesto de cuatro capítulos principales:

CAPÍTULO I: La introducción presenta el contexto general de la investigación, define el problema que se va a resolver, justifica la importancia del estudio, establece los objetivos tanto generales como específicos y la hipótesis, finalmente, explica tanto los alcances como las limitaciones del trabajo.

CAPÍTULO II: Empieza con el marco teórico; este revisa los conceptos fundamentales relacionados con la VC, las CNNs. Se profundiza en el estado del arte y en trabajos relacionados, analizando arquitecturas clave como VGGNet, GoogLeNet, ResNet y MobileNet, y finalizando con la descripción detallada de la arquitectura YOLO y su versión v8, así como las bases teóricas de la investigación.

CAPÍTULO III**:** Describe la metodología desarrollada para la solución del problema de investigación. Se detallan el proceso de obtención y construcción de la base de datos, las técnicas de preprocesamiento de imágenes aplicadas, y las librerías y frameworks utilizados. Asimismo, se explican las etapas de entrenamiento, prueba de los modelos y las métricas de evaluación seleccionadas para validar el conteo.

CAPÍTULO IV: Se presentan los resultados obtenidos tras el entrenamiento y las pruebas de los modelos. Se incluyen una discusión y análisis comparativo del rendimiento de las diferentes arquitecturas evaluadas, mostrando la efectividad de la detección y el conteo de estudiantes.

CAPÍTULO V: Se exponen las conclusiones derivadas del estudio, validando el cumplimiento de los objetivos e hipótesis planteadas. Finalmente, se sugieren líneas de trabajo futuro para dar continuidad y mejora a la investigación.

# CAPÍTULO II MARCO TEÓRICO

Para entender cómo se logró que una computadora cuente estudiantes en espacios educativos, primero se debe explorar el mundo de la VC. En este capítulo, exploraremos los conceptos, las herramientas y las ideas que hicieron posible nuestro proyecto. No se empieza desde cero; hay una base de investigación y descubrimientos de muchos otros científicos. Esta revisión permitirá situar la presente tesis en el panorama científico actual, identificar avances relevantes y fundamentar las decisiones metodológicas tomadas.

## 2.1 Estado del arte

La detección de personas es una de las principales tareas en la detección de objetos, y para resolver este problema en la actualidad existen modelos con diferentes métodos, entre ellos encontramos las redes convolucionales y el método Transformer (Arkin, et al., 2022). Existen métodos para resolver la detección de personas, por ejemplo: métodos mecánicos (Alfaro Hernández, E., et al. 2024), electrónicos (Instituto Nacional de Seguridad y Salud en el Trabajo (INSST) 2018), radiofrecuencia (Yang, C. et al.,2023), e IA (Nguyen. et al., 2025). Este trabajo de investigación se enfoca solo en el uso de las redes convolucionales pre-entrenadas, que combinan la aplicación de VC con el uso de redes neuronales.

La detección de objetos en una imagen puede realizarse por clases; es decir, el modelo debe identificar los objetos incluidos en la imagen. En esta investigación, la clase objetivo son los estudiantes, quienes primero son detectados y posteriormente contabilizados dentro del ambiente educativo presente en la imagen. Para esta tarea se hará un enfoque en los métodos y herramientas que permitan detectar a una persona en un ambiente controlado como condiciones de luz y espacios cerrados. A continuación, se discuten algunas de las redes neuronales convolucionales utilizadas para la detección de personas y estudiantes.

El conteo de estudiantes en entornos educativos mediante VC ha sido un área de investigación activa, con enfoques que han evolucionado para abordar desafíos como la detección de rostros pequeños, la oclusión parcial y la eficiencia computacional. Los trabajos de investigación revisados muestran una progresión desde detectores para un objetivo en

específico basados en contexto hasta la implementación de modelos de CNNs de la familia YOLO.

Una de las investigaciones en la literatura sobre cómo abordar la complejidad de las aulas físicas fue presentada por Chen et al. (2020). En lugar de depender únicamente de la detección de rostros, su enfoque se basó en un detector personalizado que utilizaba información contextual del entorno, como hombros, cuerpos y escritorios, para inferir la presencia de estudiantes incluso cuando los rostros estaban en situación de oclusión o en ángulos que dificultan la detección. Su metodología, basada en una arquitectura de Feature Pyramid Networks (FPN), incluía un Módulo de Mejora de Características (FEM) para hacer las detecciones más robustas. Para mejorar la fiabilidad del modelo, implementaron un "Módulo de Validación por Video", el cual analizaba múltiples fotogramas de un corto período para confirmar las detecciones hechas, aprovechando los momentos de inmovilidad de los alumnos. Este método combinado les permitió alcanzar una precisión en el conteo de estudiantes del 96.5% y un mAP (mean Average Precision) de 91.1%. Es importante mencionar que la base de datos usada fue de imágenes extraídas de videos de una cámara de vigilancia de un salón de clases.

Con la popularización de los modelos YOLO, la investigación se orientó hacia soluciones más directas y eficientes y exploraron la implementación de modelos como YOLOv3, no solo para el conteo, sino también para el análisis de sentimientos en alumnos en aulas virtuales. Su metodología se caracteriza por una etapa de preprocesamiento en la que transformaron las imágenes a escala de grises y aplicaron técnicas de normalización para manejar las diversas condiciones de iluminación de los entornos. En sus resultados de conteo, lograron una precisión promedio del 96.77% usando un umbral de confianza superior a 0.50. En este trabajo se demuestra la efectividad de los modelos YOLO en escenarios no controlados como las clases a distancia.

Continuando con la evolución de YOLO, la investigación más reciente del mismo equipo se centró en el desafío de la oclusión en aulas físicas, utilizando YOLOv5 para comparar sus versiones YOLOv5n y YOLOv5l. El objetivo era desarrollar un sistema para tolerar oclusiones, para lo cual se sintonizaron los hiperparámetros del modelo como un nivel de confianza de 0.35. Los resultados fueron reveladores: mientras ambos modelos alcanzaron

una Precisión de 1.0 (sin falsos positivos), el modelo YOLOv5l demostró una robustez significativamente mayor con un Recall de 0.99, en comparación con el 0.88 de la versión YOLOv5n. Además, el estudio reportó el beneficio del hardware utilizado, mostrando que la implementación en GPU de YOLOv5l era 14.04 veces más rápida en inferencia que en CPU.

## 2.3 Visión por computadora para la detección de personas

La VC es una disciplina que se encuentra en la intersección de la IA, la informática, el procesamiento de imágenes y la ciencia cognitiva. Su tarea fundamental no es principalmente procesar píxeles, sino dar a las máquinas una capacidad análoga a la visión humana: la habilidad de "ver", interpretar y comprender el mundo a través de datos visuales.

### 2.3.1 Definiendo la VC

La VC forma parte de la IA como un campo que entrena a las computadoras para interpretar y comprender el mundo visual, utilizando imágenes digitales provenientes de cámaras, videos y otras tecnologías de sensores, y aprovechando modelos de aprendizaje profundo, las máquinas pueden identificar y clasificar objetos con precisión, y subsecuentemente, reaccionar a lo que "ven"(*SAS, 2025*). El objetivo último es automatizar y, en muchos casos, superar las capacidades del sistema visual humano en una amplia gama de tareas (Kotwel & Kotwel, 2024).

El proceso operativo de un sistema de VC puede descomponerse en tres pasos fundamentales y secuenciales:

1. **Adquisición de la Imagen:** El proceso comienza con la captura de datos visuales. Estos pueden ser imágenes estáticas, secuencias de video en tiempo real o incluso datos tridimensionales obtenidos a través de tecnologías como LiDAR (Light Detection and Ranging) (*SAS, 2025*).
2. **Procesamiento de la Imagen:** Una vez adquirida, la imagen es procesada mediante algoritmos. En la era moderna, este paso es dominado por modelos de aprendizaje profundo, específicamente las CNNs. Estos modelos, a menudo pre-entrenados con miles o millones de imágenes etiquetadas, automatizan la extracción de características

relevantes de los datos de píxeles brutos (*SAS, 2025*).

3. **Comprensión de la Imagen:** Este es el paso interpretativo final, en el que el sistema realiza una tarea específica. Basándose en las características extraídas, un objeto puede ser clasificado, localizado dentro de la imagen, o sus límites pueden ser segmentados a nivel de píxel (*SAS, 2025*).

El objetivo final consiste en imitar la visión humana en un espectro de tareas que incluyen el reconocimiento óptico de caracteres (OCR), la detección de objetos, la clasificación de imágenes y la segmentación de escenas entre otras (Muiruri, 2025). A pesar de los avances exponenciales, la replicación completa de la complejidad, la robustez y la generalización de la visión biológica sigue siendo uno de los desafíos en el campo de la informática (Muiruri, 2025).

## 2.4 CNNs

Las CNNs son una clase de redes neuronales diseñadas para procesar datos con una estructura de cuadrícula, como lo son las imágenes. Introducidas por LeCun et al. (1998), las CNNs han revolucionado el campo de la VC, proporcionando resultados de vanguardia en tareas de clasificación y detección de objetos.

La estructura típica de una CNN se compone de varias capas fundamentales que trabajan en conjunto para transformar una imagen de entrada en una salida deseada, como la clasificación de un objeto o la predicción de su ubicación (Hernández et. al. 2024). Las capas principales incluyen:

**Capas convolucionales**: son el núcleo de la CNN. Realizan una operación de convolución entre un conjunto de filtros (o kernels) aprendibles y la imagen de entrada (o el mapa de características de la capa anterior). Cada filtro está diseñado para detectar patrones en la red. El resultado de esta operación es un mapa de características que resalta la presencia de esos patrones en distintas ubicaciones espaciales de la entrada.

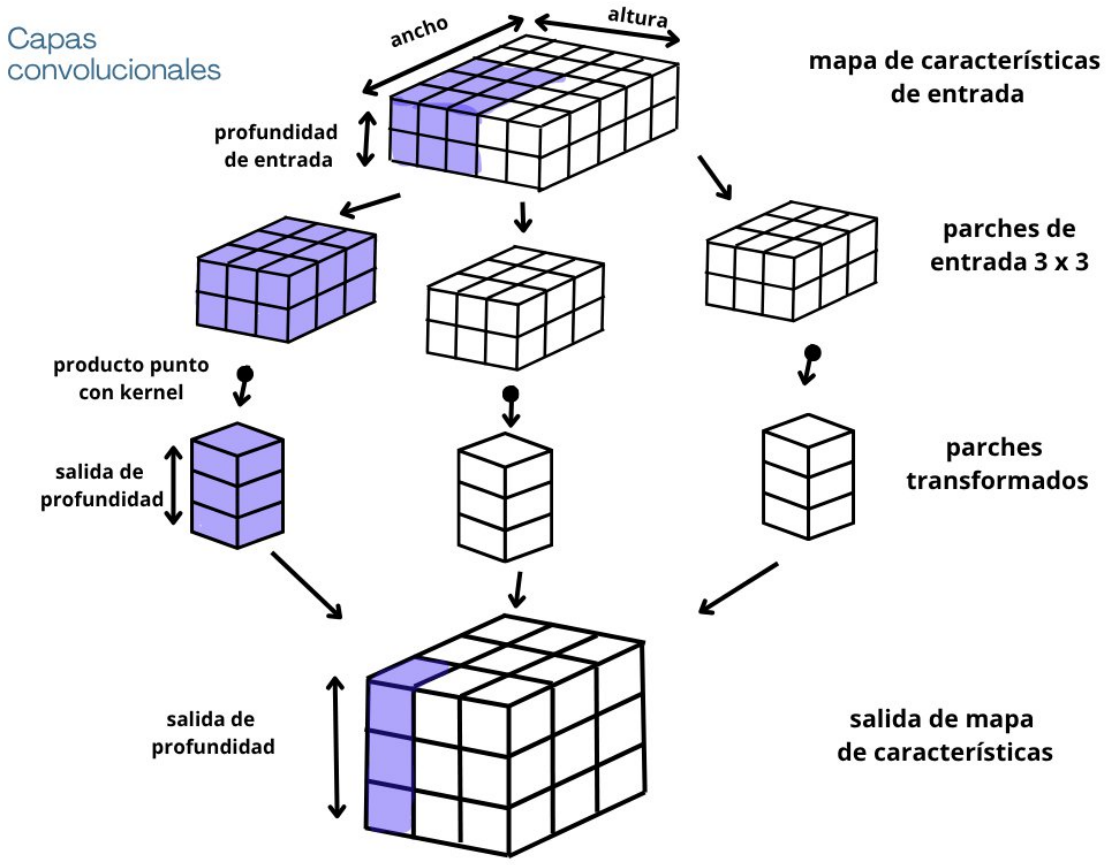



Figura 2.1. Operación básica de redes neuronales convolucionales (Chollet, 2018)

**Funciones de activación:** Después de cada capa convolucional, se aplica una función de activación no lineal (como ReLU, Sigmoide o Tanh) a los mapas de características. Estas funciones introducen no linealidades en el modelo, lo que le permite aprender relaciones más complejas a partir de los datos. ReLU(max(0,x)) es una de las más utilizadas debido a su eficiencia computacional y su capacidad para mitigar el problema del desvanecimiento del gradiente.

**Capas de agrupación (Pooling):** Estas capas tienen como objetivo reducir la dimensionalidad espacial de los mapas de características, lo que disminuye la cantidad de parámetros y la carga computacional, a la vez que proporciona invarianza frente a pequeñas traslaciones. Las operaciones de pooling más comunes son Max Pooling (selecciona el valor máximo dentro de una ventana) y Average Pooling (calcula el promedio de los valores dentro de una ventana).

**Capas totalmente conectadas (Fully Connected Layers):** Después de varias etapas de convolución y pooling, los mapas de características de alto nivel, que representan la información semántica extraída de la imagen, suelen aplanarse y conectarse a una o más capas totalmente conectadas. En estas capas, cada neurona está conectada a todas las neuronas de la capa anterior. Son responsables de realizar la tarea final, como la clasificación (asignando probabilidades a diferentes clases de objetos) o la regresión (prediciendo las coordenadas de las cajas delimitadoras).

El proceso general de identificación de objetos mediante una CNN puede conceptualizarse como una transformación progresiva de la información: la imagen de entrada se descompone inicialmente en valores de píxeles; las primeras capas convolucionales aprenden a identificar características de bajo nivel como bordes y texturas; las capas subsiguientes combinan estas características para detectar formas y partes de objetos más complejas; y finalmente, las capas más profundas (ver figura 2.1).

### 2.4.1 Métodos Modernos Basados en CNNs.

Los métodos modernos utilizan CNNs para aprender automáticamente características relevantes directamente a partir de los datos. Existen diversas aproximaciones para el conteo de personas utilizando CNNs:

**Detección de Objetos:** Utiliza CNNs para detectar y localizar a cada individuo en una imagen. Ejemplos de algoritmos populares incluyen YOLO (You Only Look Once) y Faster R-CNN (Ren, et al., 2015).

**Estimación de Densidad:** En lugar de detectar individuos, algunos métodos generan mapas de densidad que indican la probabilidad de la presencia de personas en diferentes regiones de la imagen. Este enfoque resulta útil en escenas densamente pobladas. Un ejemplo es la arquitectura CSRNet (Congested Scene Recognition Network), la cual es una arquitectura de red neuronal convolucional diseñada específicamente para el conteo de personas en imágenes de escenas con multitudes densas. Su principal innovación es el uso de convoluciones dilatadas para generar mapas de densidad de alta resolución sin perder

información espacial crucial, lo que permite una estimación más precisa del número de individuos (Li et al., 2018).

### 2.4.2 Aplicaciones en Ambientes Educativos

El conteo de estudiantes en entornos educativos se podría aplicar a lo siguiente:

**Gestión de Recursos:** Ayuda en la planificación y asignación eficiente de recursos educativos, como aulas y personal docente.

**Seguridad:** Facilita la implementación de medidas de seguridad al monitorear la ocupación de los espacios y detectar situaciones anómalas.

**Análisis de Asistencia:** Automatiza el proceso de toma de asistencia, reduciendo la carga administrativa y aumentando la precisión.

Las primeras redes convolucionales como Overfeat, CascadeCNN y MTCNN (Multi-task Cascaded Convolutional Neural Network) usaban una especie de ventana deslizante para extraer características de una imagen; sin embargo, esto resultaba costoso para los recursos de memoria:

Debido a este problema de memoria, surgieron nuevos modelos llamados de dos etapas "two stage", como R-CNN, Faster R-CNN (Regions with Convolutional Neural Networks) y R-FCN (Region-based Fully Convolutional Network) (Arkin et al., 2022).

El funcionamiento de las CNNs depende de cada tipo; en general, la figura (2.2) describe su funcionamiento:

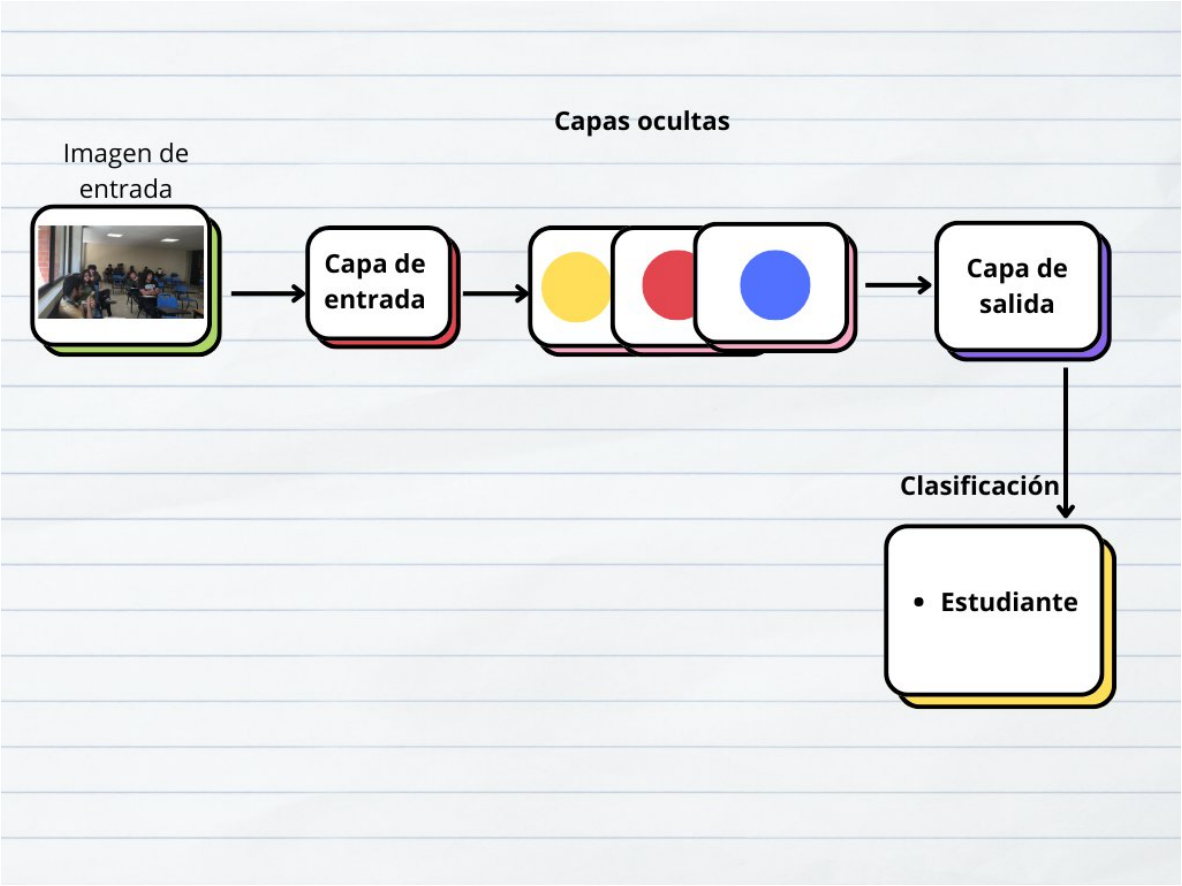



Figura. 2.2 Esquema simplificado de una red CNN (Krichen, 2023).

Una CNN generalmente tiene varias capas, cada una con una función específica. La capa convolucional es la base de una CNN y realiza una convolución en la imagen de entrada con filtros que aprenden. Los filtros son matrices pequeñas que se mueven sobre la imagen, calculando un producto punto en cada paso y generando mapas de características que representan diferentes aspectos de la imagen. La capa de pooling reduce las dimensiones espaciales de estos mapas, tomando el valor máximo o el promedio de pequeñas regiones, lo que hace que la red sea más eficiente y resista a pequeñas variaciones en la imagen de entrada. La capa de activación aplica una función de activación no lineal a la salida de la capa anterior, lo que permite a la red aprender características más complejas. La capa de normalización por lotes normaliza la salida de la capa anterior restando la media y dividiendo por la desviación estándar del lote, lo que ayuda a la red a entrenarse más rápido y de manera más estable. La capa de dropout desactiva aleatoriamente un porcentaje de las neuronas de la capa anterior

durante el entrenamiento para evitar el sobreajuste, lo que hace que la red aprenda características más robustas. Finalmente, la capa completamente conectada (fully connected) conecta todas las neuronas de la capa anterior con todas las neuronas de la capa actual, y se usa al final de la red para producir la salida final (Krichen, 2023).

## 2.5 Definición y Fundamentos de las CNNs Preentrenadas

Una CNN pre-entrenada es un modelo de red neuronal que ha sido entrenado previamente en un conjunto de datos a gran escala, típicamente para una tarea de reconocimiento de imágenes de propósito general (Masood, 2024). El ejemplo canónico y más influyente es el de un modelo entrenado en el conjunto de datos.

**ImageNet**, una vasta colección que contiene más de 14 millones de imágenes anotadas en más de 1000 categorías de objetos cotidianos (Thakur, R. 2024). El proceso de entrenamiento en un conjunto de datos de esta magnitud y diversidad permite que el modelo aprenda a reconocer una jerarquía robusta y generalizable de características visuales (Masood, 2024).

En las capas iniciales, la red aprende a detectar características de bajo nivel, como bordes, gradientes de color, texturas y patrones simples. A medida que la información fluye a través de las capas más profundas de la red, estas características simples se componen para formar representaciones más complejas y abstractas, como partes de objetos (una rueda, un ojo), formas completas y, finalmente, conceptos semánticos de alto nivel (un automóvil, un rostro humano) (Fagbuyiro, D. 2024).

Los "pesos" y "sesgos" de la red, que son los parámetros numéricos ajustados durante el proceso de entrenamiento, encapsulan este conocimiento jerárquico. Una vez que el entrenamiento ha concluido y el modelo ha alcanzado un alto nivel de rendimiento en la tarea original (por ejemplo, la clasificación en ImageNet), estos pesos se guardan y se ponen a disposición de la comunidad investigadora (Masood, 2024). El resultado es un modelo "preentrenado": un sistema que ya posee un conocimiento fundamental y sofisticado sobre el mundo visual, listo para ser adaptado a nuevas tareas (Masood, D. 2024). Este

conocimiento preadquirido constituye una base de partida sumamente poderosa para una amplia variedad de problemas de VC, que van desde la clasificación de imágenes y la detección de objetos hasta la segmentación semántica y el análisis de imágenes médicas (Masood, 2024).

### 2.5.1 Arquitecturas fundacionales y su evolución.

El desarrollo de las redes neuronales convolucionales no ha sido una progresión lineal, sino una evolución de ideas y filosofías de diseño. Cada arquitectura influyente no solo ha mejorado las métricas de rendimiento, sino que también ha introducido un nuevo enfoque para resolver un problema fundamental. Esta sección analiza tres arquitecturas seminales, VGGNet, GoogLeNet (Inception) y Xception, que ilustran una clara línea de pensamiento: desde la búsqueda de la profundidad mediante la fuerza bruta y la simplicidad, hasta la búsqueda de la eficiencia computacional a través de la complejidad paralela y, finalmente, la purificación de esa idea hasta su extremo lógico. Este recorrido traza la transición de "¿qué tan profundo podemos llegar?" a "¿qué tan inteligentemente podemos diseñar la profundidad?".

### 2.5.2 VGGNet.

La familia de modelos VGGNet, propuesta por Karen Simonyan y Andrew Zisserman del Visual Geometry Group de la Universidad de Oxford, representa un hito en la historia de las CNN. Su contribución clave, presentada en el desafío ILSVRC 2014, fue una investigación sistemática y exhaustiva sobre cómo la profundidad de la red afecta su precisión en tareas de reconocimiento de imágenes a gran escala (Simonyan et al., 2014).

### 2.5.3 El Kernel Uniforme de 3x3

La filosofía de diseño de VGGNet fue una de notable simplicidad y uniformidad, un marcado contraste con arquitecturas anteriores como AlexNet, que utilizaban una variedad de tamaños de filtros grandes en sus primeras capas (por ejemplo, 11x11) (Arkin et al., 2022). VGG adoptó un enfoque radicalmente diferente: utilizó exclusivamente filtros convolucionales muy pequeños de tamaño 3×3 en toda la red, con un paso (*stride*) de 1 (Krichen, 2023).

La justificación de esta elección de diseño es doble. Primero, una pila de capas convolucionales de 3×3 puede emular el campo receptivo efectivo de filtros más grandes. Por ejemplo, dos capas de 3×3 consecutivas tienen un campo receptivo efectivo de 5×5 y tres capas de 3×3 tienen un campo de 7×7. Sin embargo, la pila de filtros más pequeños tiene dos ventajas cruciales:

1. **Mayor no linealidad:** La pila de capas incluye más funciones de activación no lineales (como ReLU), lo que hace que la función de decisión sea más discriminativa y potente (Goodfellow et al., 2016).
2. **Menos parámetros:** Una pila de tres capas de 3×3 con C canales requiere 3×(32C2) = 27C2 pesos, mientras que una sola capa de 7×7 requeriría 72C2=49C2 pesos. Esto permite construir redes más profundas con menos parámetros (Simonyan & Zisserman, 2014).

Esta uniformidad y simplicidad hicieron que la arquitectura VGG fuera elegante, fácil de entender y de implementar (Krichen, 2023).

### 2.5.4 Arquitectura de VGG16

De las diversas configuraciones propuestas por los autores, VGG16 se convirtió en la más popular y una de las de mejor rendimiento. Como su nombre indica, consta de 16 capas con pesos aprendibles: 13 convolucionales y 3 totalmente conectadas (Krichen, 2023).

La arquitectura se estructura de la siguiente manera:

- **Entrada:** Acepta una imagen RGB de tamaño fijo de 224×224 píxeles (Arkin et al., 2022).
- **Bloques Convolucionales:** Las 13 capas convolucionales se organizan en cinco bloques. Todos los bloques utilizan filtros de 3×3 con un paso de 1 y un relleno (*padding*) "same" para preservar las dimensiones espaciales dentro del bloque. El número de filtros comienza en 64 en el primer bloque y se duplica tras cada operación de pooling, alcanzando 128, 256 y, finalmente, 512 en los dos últimos bloques. Este aumento de la profundidad del canal compensa la reducción de las dimensiones espaciales (Zafar, S., 2023).

- **Capas de Agrupación (*Pooling*):** Después de cada bloque convolucional, se aplica una capa de max-pooling de 2×2 con un paso de 2. Esta operación reduce a la mitad la altura y el ancho de los mapas de características, lo que disminuye la carga computacional y ayuda a controlar el sobreajuste (Krichen, 2023).
- **Capas Totalmente Conectadas:** La arquitectura culmina en tres capas totalmente conectadas. Las dos primeras tienen 4096 neuronas cada una y la capa final es un clasificador softmax de 1000 vías, correspondiente a las 1000 clases del conjunto de datos ImageNet (Masood, 2024).

### 2.5.5 GoogLeNet (Inception)

Como respuesta directa a los crecientes costos de computación de las redes neuronales cada vez más profundas, como VGGNet, un equipo de investigadores de Google desarrolló la arquitectura Inception. Su encarnación original, denominada GoogLeNet, ganó el desafío ILSVRC 2014, superando a VGGNet (Szegedy et al., 2014). La filosofía de diseño de Inception fue fundamentalmente diferente: en lugar de simplemente apilar más capas del mismo tipo, buscaba aumentar la profundidad y la anchura de la red de manera inteligente, manteniendo constante el presupuesto de computación (Szegedy et al., 2014).

### 2.5.6 El módulo Inception

La innovación central de GoogLeNet es el módulo Inception. La idea principal detrás de este módulo es aproximar una estructura local óptima y dispersa en una red convolucional mediante componentes densos y fácilmente computables (Szegedy et al., 2014). En lugar de obligar a la red a elegir un único tamaño de filtro para una capa, el módulo Inception realiza múltiples operaciones en paralelo sobre la misma entrada (Kaur, 2024). Un módulo Inception típico aplica convoluciones de 1×1, 3×3 y 5×5, junto con una operación de max-pooling de 3×3, todo al mismo tiempo. Las salidas de estas ramas paralelas se concatenan posteriormente a lo largo de la dimensión del canal para formar la salida del módulo (Maelfabien, 2019).

Esta estructura de múltiples ramas permite a la red capturar características visuales a diferentes escalas simultáneamente. Los filtros de 1×1 capturan detalles muy finos, los de

3×3 capturan características locales y los de 5×5 capturan características globales en la misma región de la imagen (Arkin et al., 2022).

Sin embargo, realizar convoluciones de 3×3 y 5×5 en mapas de características con una gran profundidad de canal (muchos canales) resulta computacionalmente muy caro. Para resolver este problema, Inception introdujo un uso ingenioso de las convoluciones de 1×1 como capas de "cuello de botella" (*bottleneck*) (Szegedy et al., 2014). Antes de las convoluciones de 3×3 y 5×5, se aplica una convolución de 1×1 para reducir la profundidad del canal de entrada. Esta reducción de la dimensionalidad disminuye drásticamente el número de operaciones en las convoluciones más grandes que le siguen, con una pérdida mínima de información. Después de la convolución principal, otra convolución de 1×1 puede restaurar la profundidad del canal, si es necesario. Este diseño de "reducir, convolucionar, restaurar" es la clave de la eficiencia computacional de Inception. Por ejemplo, un módulo que de otro modo requeriría 120 millones de cómputos podría reducirse a sólo 12.4 millones con la adición de capas de cuello de botella (Terven et al., 2023).

### 2.5.7 Inception v1 a Inception-ResNet

La arquitectura Inception no fue estática, sino que evolucionó a través de varias versiones, cada una introduciendo mejoras clave:

- **Inception v1 (GoogLeNet):** La red original de 22 capas que ganó el ILSVRC 2014. Una característica notable fue el uso de **clasificadores auxiliares** en las capas intermedias de la red (Szegedy et al., 2014). Estos eran pequeños clasificadores que se añadían durante el entrenamiento para proporcionar gradientes adicionales a las capas inferiores, combatiendo así el problema del gradiente desvanecido (*vanishing gradient*) en una red tan profunda. Estos clasificadores se descartaban durante la inferencia (Goodfellow et al., 2016).
- **Inception v2 y v3:** Estas versiones posteriores introdujeron mejoras significativas. La más importante fue la incorporación de la **Normalización por Lotes (*Batch Normalization*)**, propuesta en el mismo artículo que el de Inception v2 (Szegedy et al., 2016). La normalización por lotes estabilizó el entrenamiento, permitió tasas de aprendizaje más altas y actuó como regularizador, lo que hizo que otras técnicas, como

el dropout, fueran menos necesarias (Krichen, 2023). Inception v3 refinó aún más la arquitectura al factorizar convoluciones más grandes en convoluciones más pequeñas. Por ejemplo, una convolución de 5×5 se reemplazó por dos convoluciones de 3×3 apiladas, lo que redujo el número de parámetros y el costo computacional manteniendo el mismo campo receptivo (Arkin et al., 2022). También introdujo la regularización por suavizado de etiquetas ( *label smoothing*) (Szegedy et al., 2016).

- **Inception v4 e Inception-ResNet:** En 2016, los investigadores de Google presentaron dos nuevas arquitecturas (Szegedy et al., 2016). **Inception-v4** fue una versión más pura y uniforme de la arquitectura Inception, con un "tallo" (*stem*) más estandarizado y más módulos Inception (GeeksforGeeks, 2022). Al mismo tiempo, inspirados por el éxito de ResNet, introdujeron **Inception-ResNet-v1** e **Inception-ResNet-v2**. Estos modelos híbridos modificaron el módulo Inception para incorporar conexiones residuales (o de salto). En lugar de concatenar las salidas de las diferentes ramas, estas se sumaban a la entrada del módulo, de forma similar a un bloque residual de ResNet. Se demostró empíricamente que la adición de conexiones residuales aceleraba significativamente el entrenamiento de las redes Inception y mejoraba ligeramente el rendimiento (Szegedy et al., 2016).

## 2.6 Arquitecturas de Alta Profundidad.

Tras establecer los fundamentos con arquitecturas que exploraban la profundidad y la eficiencia computacional, el campo de la VC se enfrentó a dos nuevos y distintos desafíos. El primero era un límite práctico para la profundidad: ¿cómo entrenar redes que fueran no solo profundas, sino también *ultraprofundas* sin que su rendimiento se degradara? El segundo era un límite de despliegue: ¿cómo llevar el poder de las redes profundas a dispositivos con recursos limitados, como los teléfonos móviles? Esta sección examina dos familias de arquitecturas que proporcionaron respuestas definitivas a estas preguntas: **ResNet**, que resolvió el problema del entrenamiento de redes de profundidad extrema, y **MobileNet**, que resolvió el problema de la eficiencia en el borde (*edge*).

### 2.6.1 Resnet

A mediados de la década de 2010, los investigadores se toparon con una paradoja. Si bien la profundidad parecía ser la clave para un mejor rendimiento, simplemente apilar más y más capas en una red neuronal profunda conducía a un problema conocido como degradación. En este fenómeno, la precisión de la red se saturaba y luego disminuía rápidamente a medida que se añadían más capas (Kurama, 2025). Es importante destacar que este problema no se debía al sobreajuste, ya que el error aumentaba tanto en el conjunto de entrenamiento como en el de prueba. Experimentos mostraron que una red "plana" de 56 capas tenía un error de entrenamiento más alto que una de 20 capas (He et al., 2016). El problema radicaba en la optimización: las redes muy profundas eran inherentemente más difíciles de entrenar.

### 3.6.2 MobileNet

Mientras ResNet resolvía el problema de la profundidad, surgió otro reto igualmente importante, impulsado por el auge de la computación móvil y del Internet de las Cosas (IoT). Las arquitecturas de vanguardia como VGG y ResNet, aunque precisas, eran demasiado grandes, lentas y consumían demasiada energía como para ser desplegadas en dispositivos con recursos limitados, como teléfonos inteligentes o sistemas embebidos (Howard et al., 2017). Una red como VGG16, con sus más de 500 MB de tamaño de modelo, era simplemente impráctico para una aplicación móvil (Rosebrock, 2017).

La familia de arquitecturas MobileNet, desarrollada por Google, fue diseñada desde cero para abordar este problema. Su filosofía de diseño no era maximizar la precisión a toda costa, sino lograr el mejor equilibrio posible entre precisión, latencia, consumo de energía y tamaño del modelo (*Kaggle*, 2025).

La principal estructura de MobileNet es el uso extensivo de convoluciones separables en profundidad (*depthwise separable convolutions*), la misma operación eficiente que se encuentra en Xception (Howard et al., 2017). Esta técnica factoriza una convolución estándar en dos pasos:

1. Una convolución en profundidad que aplica un único filtro a cada canal de entrada por separado.

2. Una convolución puntual (una convolución de 1×1) que combina las salidas de la convolución en profundidad.

Esta factorización reduce drásticamente la carga computacional y el número de parámetros, típicamente entre 8 y 9 veces en comparación con una convolución estándar, con solo una pequeña pérdida de precisión (Howard et al., 2017).

## 2.7 YOLO

YOLO, introducido por Joseph Redmon et al. (2015), propusieron una solución a la lentitud y complejidad de sistemas de detección anteriores: tratar la detección de objetos como un único problema de regresión (Redmon et al., 2015). En lugar de un proceso de dos etapas, una única red neuronal convolucional procesa la imagen completa una sola vez y predice directamente las coordenadas de los cuadros delimitadores y las probabilidades de clase asociadas en una sola evaluación. Esta arquitectura unificada es entrenable de extremo a extremo y es extremadamente rápida (Redmon et al., 2015).

### 2.7.1 YOLO v8

YOLOv8 fue desarrollado por Ultralytics, y representa la versión más reciente estable de la familia de algoritmos de detección de objetos en tiempo real YOLO (Ultralytics 2018). Este modelo ha ganado una considerable atención en la comunidad de VC debido a sus notables mejoras en precisión y eficiencia en comparación con sus predecesores y otros modelos de detección de objetos (Solawetz, 2023). YOLOv8 no solo sobresale en la detección de objetos, sino que también ofrece capacidades para la segmentación de instancias y la clasificación de imágenes, lo que lo convierte en una herramienta versátil para una amplia gama de aplicaciones (Wang et al. 2020).

La arquitectura de YOLOv8 introduce varias innovaciones significativas que contribuyen a su rendimiento superior. Si bien Ultralytics no ha publicado un *paper* formal detallando exhaustivamente cada aspecto arquitectónico como en versiones anteriores, la comunidad y la documentación oficial destacan los siguientes componentes y mejoras:

1. Nuevo Backbone: YOLOv8 incorpora un nuevo diseño de red troncal (backbone) que es más eficiente y potente. Aunque los detalles específicos pueden variar entre las diferentes sub-versiones (e.g., YOLOv8n(nano), YOLOv8s (small), YOLOv8m (medium), YOLOv8l (large), YOLOv8x (extra large), generalmente se basa en los principios de las redes CSP (Cross Stage Partial) como Darknet, optimizadas para un mejor flujo de gradientes y una reducción en la carga computacional (Wang et al. 2020,). Esta nueva estructura permite una extracción de características más robusta.
2. Neck sin Anclas (Anchor-Free Head): Una de las modificaciones más sustanciales en YOLOv8 es la adopción de un cabezal de detección sin anclas (anchor-free), (Solawetz 2023). Las versiones anteriores de YOLO dependían de cajas de anclaje predefinidas para predecir las cajas delimitadoras. El enfoque sin anclas simplifica el proceso de entrenamiento y posprocesamiento, ya que elimina la necesidad de ajustar estos hiperparámetros, lo que a menudo era específico del conjunto de datos. Esto permite que el modelo sea más generalizable y reduzca el número de predicciones por imagen, lo que puede llevar a una mayor velocidad (Geet al. 2021).

3. Nueva Función de Pérdida: YOLOv8 introduce modificaciones en la función de pérdida para mejorar la precisión de la clasificación y la localización de las cajas delimitadoras. Aunque los detalles exactos pueden ser propiedad de Ultralytics, se sabe que se enfoca en optimizar la asignación de etiquetas y la calidad de las predicciones de las cajas (Terven et al., 2023). Algunas fuentes sugieren el uso de la función de pérdida de Focalización de Distribución (Distribution Focal Loss) y la pérdida CIoU (Complete Intersection over Union) o variantes similares para la regresión de las cajas (Li et al. 2020).
4. PAN (Path Aggregation Network) en el Neck: Al igual que versiones recientes de YOLO, YOLOv8 utiliza una estructura de cuello (neck) como PANet para combinar características de distintos niveles del backbone (Wang et al. 2020). Esto permite combinar información semántica rica de las capas más profundas con información espacial detallada de las más superficiales, mejorando la detección de objetos en diferentes escalas.
5. Optimización para la Exportación y el Despliegue: Ultralytics ha puesto un fuerte énfasis en la facilidad de uso y de despliegue de YOLOv8. El modelo es compatible

con diversos formatos de exportación, como ONNX, TensorRT y CoreML, entre otros, lo que facilita su implementación en una variedad de plataformas y dispositivos, desde servidores hasta dispositivos móviles y embebidos (Terven et al., 2023).

YOLOv8 ha demostrado un equilibrio sobresaliente entre precisión (medida comúnmente por el mAP, mean Average Precision) y velocidad (FPS, Frames Per Second) en benchmarks estándar como COCO (Common Objects in Context) (Lin et al. 2014). Las diferentes variantes del modelo (nano, small, medium, large, xlarge) ofrecen un *equilibrio que permite a los usuarios seleccionar la* que mejor se adapte a sus necesidades específicas de rendimiento y de recursos computacionales (Terven et al., 2023). Generalmente, las versiones más grandes ofrecen mayor precisión a costa de una menor velocidad y de un mayor consumo de recursos, mientras que las versiones más pequeñas son más rápidas y ligeras, ideales para aplicaciones en tiempo real con hardware limitado.

Gracias a su versatilidad y rendimiento, YOLOv8 es adecuado para una amplia gama de tareas de VC, incluyendo:

- Detección de objetos en tiempo real: para aplicaciones como la videovigilancia, los vehículos autónomos y la robótica (YOLOv8 en GitHub).
- Segmentación de instancias: Permite no solo la detección sino también la delimitación a nivel de píxel de cada objeto (Terven et al., 2023).
- Clasificación de imágenes: Aunque su fortaleza radica en la detección, también puede ser entrenado para tareas de clasificación (Terven et al., 2023).
- Seguimiento de objetos (Tracking): Al combinarse con algoritmos de seguimiento, YOLOv8 puede utilizarse para rastrear objetos a lo largo de secuencias de vídeo (Wojke et al. 2017).

# CAPÍTULO III METODOLOGÍA

Para la implementación del modelo de detección y conteo de estudiantes mediante VC y redes neuronales convolucionales preentrenadas, se definió una metodología compuesta por siete etapas (ver Figura 3.1).

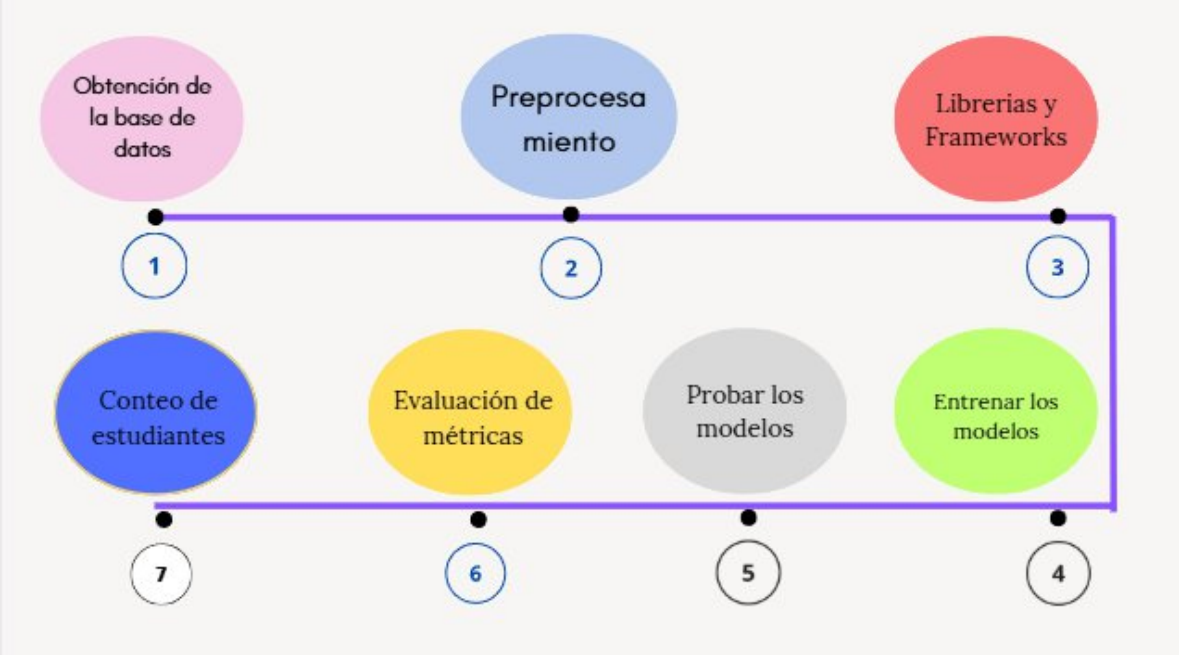



Figura. 3.1 Metodología establecida para la implementación del proyecto.

Para la obtención del conjunto de datos (dataset) se buscó en una variedad de páginas y blogs en internet donde la comunidad de investigadores proporciona datasets de uso libre como Roboflow, GitHub, Kaggle entre otras. La base de datos buscada tenía que cumplir al menos con ciertas características, ya que este trabajo de investigación se sitúa en un contexto escolar, es decir, que las imágenes se sitúen en ambientes cerrados como salones, laboratorios de cómputo e incluso auditorios; además, se buscaba que estos espacios tuvieran diferentes condiciones de iluminación, oclusiones, diferentes ángulos y diferente grado de ocupación. Después de esta búsqueda, el conjunto de imágenes elegido para este estudio resultó ser el

dataset “classroom-monitoring-dataset” (Classroom Monitoring Dataset, 2021), el cual fue descargado del sitio kaggle.com, un sitio web que reúne contenido publicado por la comunidad de científicos dedicados al machine learning, donde se comparte todo tipo de datasets.

De todas las características buscadas, cumple con la oclusión, diferentes condiciones de luz y niveles de ocupación, y ángulos en espacios cerrados como salones y auditorios, a excepción de los laboratorios de cómputo.

El dataset consta de 4342 imágenes en formato JPG y 4 videos en formato mp4. La mayoría de las imágenes tienen una dimensión diferente con respecto a la otra, las cuales pueden ir desde 1261px x 710 px a 974px x 545 px, razón por la cual la tarea de redimensión de las imágenes se convierte en un paso muy importante para el preprocesamiento de estas. El dataset está dividido en 2 partes:

Parte A.- Contiene 2000 imágenes las cuales fueron extraídas de una cámara de vigilancia de un salón de clases tomadas desde un mismo ángulo, la serie de imágenes en esta parte “A” muestra un salón de clases con diferentes niveles de ocupación y variabilidad de iluminación, así como casos de oclusión y estudiantes que se encuentran a una distancia considerable de la cámara, lo cual resulta en un caso de reto para la detección de estudiantes. Esta parte fue descartada debido a que cuenta con un solo ángulo, las caras de los estudiantes son indistinguibles y están tomadas en un solo espacio.


Figura.3.2.1 Imagen de ejemplo del dataset “classroom-monitoring-dataset” (*Classroom Monitoring Dataset*, 2021) parte A


Figura.3.2.2 Segunda imagen de ejemplo del dataset “classroom-monitoring-dataset” (*Classroom Monitoring Dataset*, 2021) parte A

Parte B.- Esta parte está compuesta de 2342 imágenes en formato JPG en la que se muestran diferentes escenarios de espacios educativos. Las imágenes fueron tomadas desde diferentes ángulos, cuenta con diferente grado de ocupación, incluye personas de diferente edad, aspecto, color de piel y uso de accesorios como lentes gorras y sombreros, así como diferentes condiciones de iluminación.

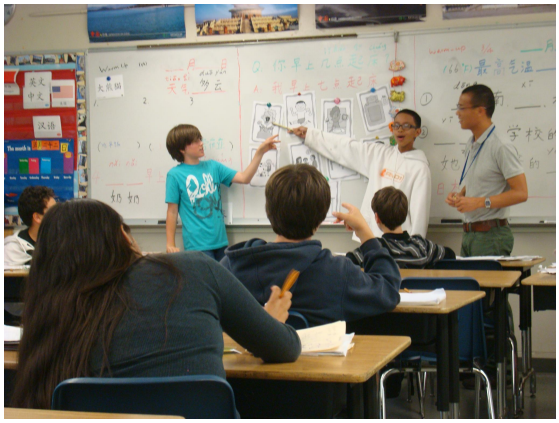


Figura 3.2.3 Imagen de ejemplo del dataset “classroom-monitoring-dataset” (*Classroom Monitoring Dataset*, 2021) parte B

Figura 3.2.4 Segunda imagen de ejemplo del dataset “classroom-monitoring-dataset” (*Classroom Monitoring Dataset*, 2021) parte B

Tomando en cuenta las características antes mencionadas se decidió utilizar el dataset “classroom-monitoring-dataset” parte B, el cual cumple con las características buscadas para las etapas de entrenamiento, validación y prueba del modelo.

Como parte de esta investigación, se agregaron imágenes recopiladas de frames de videos cortos tomados en un salón, en un laboratorio de cómputo y en el auditorio de la Facultad de Contaduría, Administración e Informática, con el consentimiento informado de los alumnos que aparecen en las imágenes, o bien que se encuentran disponibles de manera abierta en las redes sociales de la FCAeI.

Fueron grabados, en total, 5 videos; 4 de ellos en el laboratorio de cómputo, desde 4 ángulos diferentes: dos frontales y dos traseros. Un video fue tomado en un salón desde la parte frontal, haciendo un paneo completo del salón; en el caso del auditorio. Fueron tomadas dos imágenes desde redes sociales y 5 más tomadas en el mismo auditorio desde diferentes ángulos; estas últimas imágenes cuentan con diferente grado de ocupación.

De los videos divididos en frames resultaron:

- De los 4 videos del laboratorio se obtuvieron:
    - 150 frames de cada video tomada de la parte de enfrente dando un total de 300
    - 506 frames del video tomado de la parte de atrás derecha

    - 199 frames del video tomado de la parte de atrás izquierda
- 100 frames del video tomado en el salón de clases

El número de frames es diferente dado que los videos tomados tienen diferente duración

El total de imágenes derivadas de los videos y fotos tomados son 1105.

Figura 3.3 Imagen extraída de video tomado en un laboratorio de cómputo.

Del total de imágenes obtenidas de los frames del laboratorio y del salón, se tomaron 10 imágenes de cada ángulo, dando un total de 50 imágenes más las 7 del auditorio.

Se toma esa cantidad debido a la limitación del tiempo del programa de posgrado, tomar más imágenes significa disponer de más tiempo para el proceso de etiquetado, que es una tarea que lleva mucho tiempo.

Es importante mencionar que cada alumno que aparece en las imágenes descritas anteriormente accedió a participar en el estudio y accedió a firmar una carta de consentimiento informado (véase apéndice 1). En cuanto a las imágenes del auditorio, fueron recopiladas de la red social de la Facultad de Contaduría, Administración e Informática en Facebook, las cuales se encuentran posteadas de forma pública (*http://facebook.com/FCAeI/photos*, s. f.).

Estas imágenes que se añadieron al dataset de Kaggle son de ambientes reales y con personas de México, lo que aportará al modelo mayor variedad de escenarios para el procesamiento en las etapas de entrenamiento y prueba, el dataset completo se encuentra disponible en el siguiente repositorio de Github:

https://github.com/LixxMeza/School-environments-M-xico

## 3.2 Preprocesamiento de las imágenes

Para buscar que el modelo obtenga métricas satisfactorias y evitar el alto costo computacional, fue necesario que las imágenes fueran llevadas a una etapa de preprocesamiento. Como primer paso para la preparación de las imágenes, se procedió a etiquetar las imágenes con la clase objetivo que en este caso es "estudiante", para "enseñarle" al modelo qué queremos detectar. Para esta tarea se utilizó el sitio en línea Roboflow (*Roboflow: Computer Vision Tools For Developers And Enterprises*, s. f.), el cual es un sitio en internet que permite la recopilación de datos (mediante Roboflow Collect y Universe), la gestión de conjuntos de datos, la conversión y el etiquetado asistido por IA (utilizando Roboflow Annotate y herramientas como Smart Polygon para detección de objetos, segmentación y clasificación), el preprocesamiento y aumento de datos para robustecer el modelo (mediante técnicas como rotación en grados, volteo, recorte aleatorio entre otras), la exportación de datos a múltiples formatos (como YOLO), el entrenamiento y la evaluación de modelos personalizados.

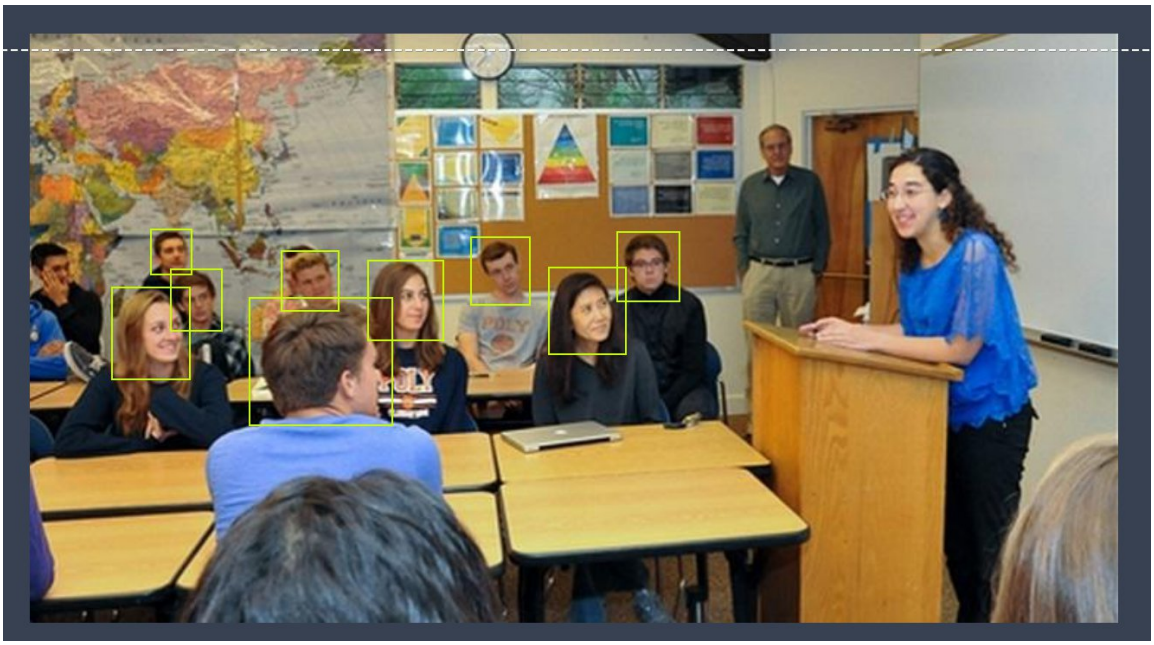


Figura 3.4 Proceso de etiquetado en Roboflow, en la figura se muestran estudiantes con cajas delimitadoras asignadas por la autora, faltando algunas personas por etiquetar.

Esta plataforma nos permite subir la base de datos y crear proyectos en los cuales podemos definir la clase objetivo y hacer el etiquetado de las imágenes por medio de cajas delimitadoras. La estrategia usada para la delimitación de las cajas de etiqueta fue marcar como estudiantes únicamente las cabezas de las personas en las imágenes. Esta forma de etiquetar las imágenes nos proporciona claridad visual al momento de observarlas y hacer el seguimiento del etiquetado, además se reduce el ruido al no delimitar más información como cuerpo completo o medio cuerpo. (véase figura 3.7)

En esta etapa, 977 imágenes del dataset fueron utilizadas para el etiquetado y resultaron 14645 etiquetas.

Roboflow también permite que una vez que se ha culminado con el proceso de etiquetado, se le apliquen filtros a las imágenes. Para esta investigación se decidió utilizar el filtro de escala de grises para que el modelo procese valores en un canal entre 0 - 255, en lugar de 3 canales como en las imágenes a color (véase la figura 3.8). Esto se realizó con el propósito de reducir el costo computacional tanto en la etapa de preprocesamiento, entrenamiento y validación.

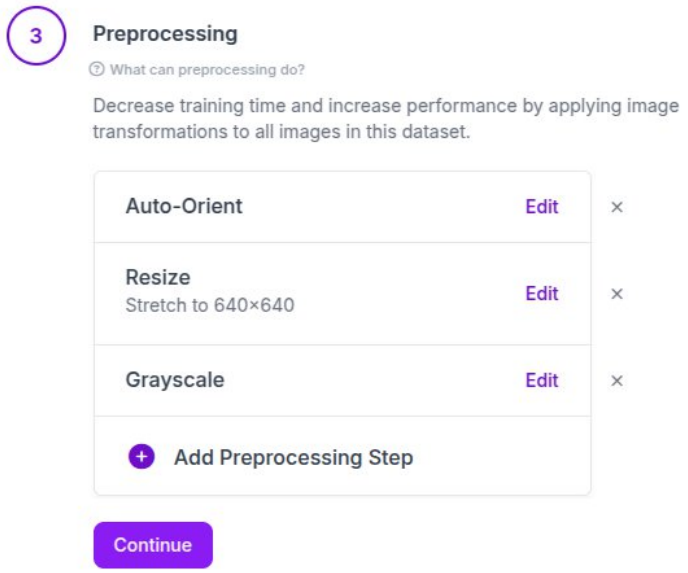



Figura 3.5. Interfaz de aplicación de orientación, redimensión y filtros, como escala de grises, a las imágenes como parte del preprocesamiento.

De las imágenes que fueron etiquetadas en su totalidad, se les aplicó una de las técnicas de aumentación disponibles en Roboflow (véase figura 3.9), esta técnica permite la rotación de las imágenes a 15 grados, lo cual ayudará al modelo a detectar la clase objetivo incluso en situaciones donde no se presente una alineación perfecta o natural de la cabeza.

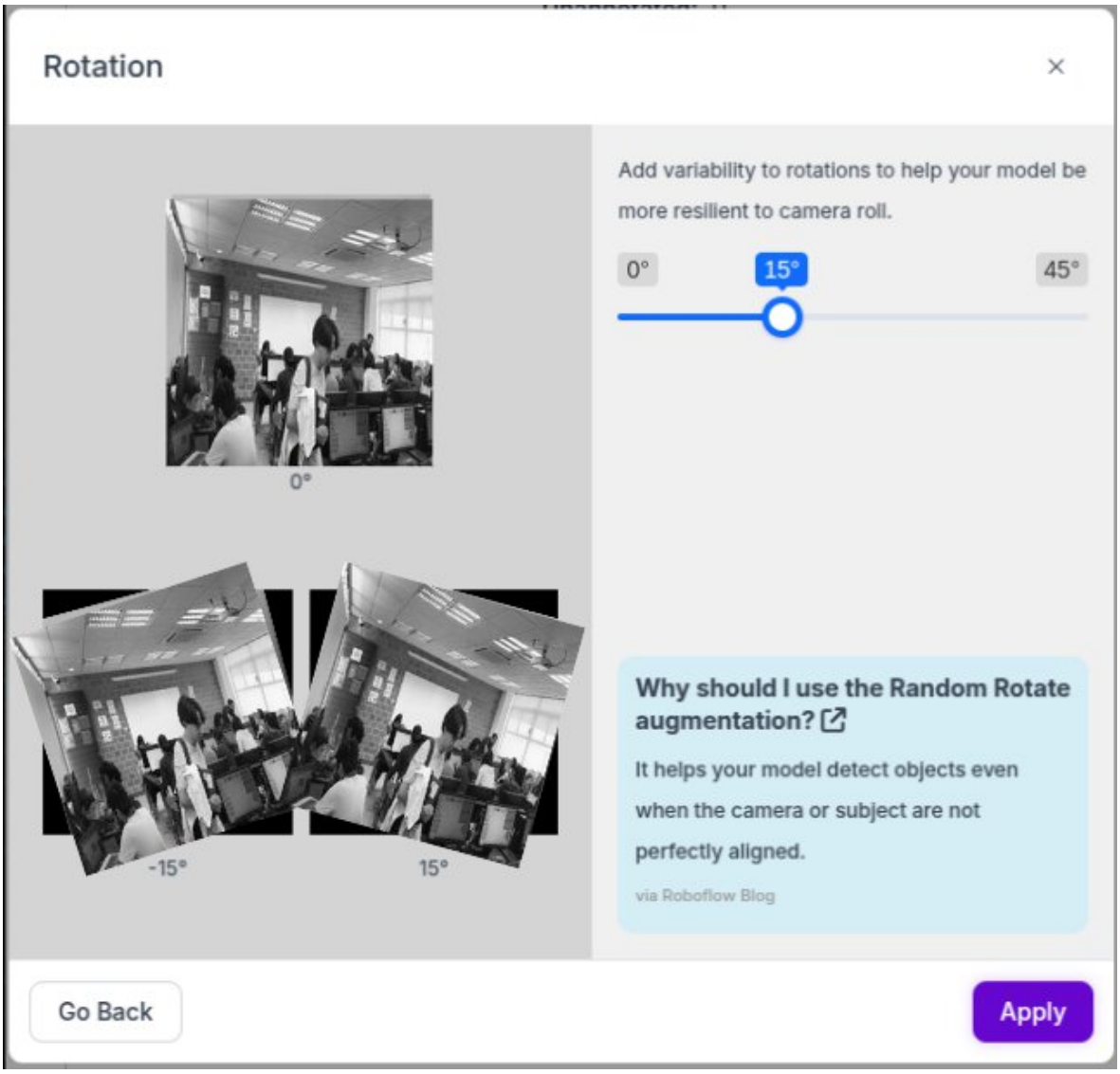



Figura 3.6 Aplicación de proceso de aumentación del dataset, en la imagen se muestra la técnica de rotación a las imágenes.

Después de haber aplicado esta técnica, se obtiene un resultado final de un dataset de 2,539 imágenes con 37,743 etiquetas.

## 3.3 Librerías y frameworks.

La siguiente etapa consiste en identificar las herramientas que ayuden a implementar el modelo y permitan la detección y conteo de estudiantes. Existen librerías y frameworks disponibles para su uso ya probados en diferentes trabajos de investigación e implementaciones (Chen et al., 2020), por lo que pueden ser utilizados con confianza en plataformas como YOLOv8, Keras, Pytorch y Roboflow.

Las librerías y frameworks utilizados para la implementación de este proyecto de investigación, no tienen problema de compatibilidad entre ellas y se instalaron directamente desde la terminal del ambiente creado en Visual Studio Code, la plataforma que se usó para realizar el entrenamiento, validación y pruebas utilizando cuadernos de Jupyter Notebook en un equipo de la marca Dell con las siguientes características:

- GPU NVIDIA 4060
- RAM 16GB
- 20 procesadores lógicos 2.60 GHz

En estos recursos de hardware, fue cambiado el sistema operativo Windows 11 por Ubuntu 24.04. Este cambio se realizó ya que se presentaron problemas de compatibilidad con la tarjeta gráfica NVIDIA y Windows constantemente mostraba la pantalla azul haciendo difícil avanzar en el proyecto, este problema se resolvió gracias al cambio de sistema operativo y la instalación correcta de drivers.

Se utilizó YOLOv8 ya que es el modelo de detección de objetos central en este trabajo de investigación, es un modelo estable, probado, y facilita el entrenamiento, la validación y la inferencia. Para su instalación se utilizó la implementación proporcionada por Ultralytics (Ultralytics, s. f.) https://docs.ultralytics.com/es/models/yolov8/.

## 3.4 Entrenamiento.

Después de obtener el dataset final, se dividieron los datos de la siguiente manera: 80% de entrenamiento, 10% de validación y 10% de prueba.

Posteriormente se ajustaron los hiperparámetros del modelo, como la tasa de aprendizaje, el tamaño del lote, y el número de épocas con el propósito de optimizar el modelo.

Se seleccionaron dos modelos de YOLOv8 con el propósito de evaluar su desempeño y el compromiso entre tamaño/velocidad y precisión; los modelos elegidos fueron los siguientes:

YOLOv8n (nano) que es el más pequeño disponible de esta versión, se entrenó con 10, 30, 50 y 100 épocas, respectivamente, todas ellas con el mismo conjunto de imágenes y en el mismo hardware; se ajustó el número de lote a 16, el número de workers en 8 y la tasa de aprendizaje a 0.01.

YOLOv8l (large) es uno de los modelos más grandes de YOLOv8, no se usó el modelo YOLOv8x (extra large) dada las limitaciones de memoria del hardware utilizado. Este modelo fue entrenado de igual manera con 10, 30, 50 y 100 épocas. Se usó el mismo conjunto de imágenes de entrenamiento que en el modelo nano, así como el mismo hardware. Para la sintonización de hiperparámetros y debido a las limitaciones de memoria se tuvo que ajustar el número de lotes a 4 y el número de workers a 4 con una tasa se aprendizaje de 0.01.

Las métricas de evaluación que se utilizaron para estos modelos fueron las siguientes:

- (mAP)

El cálculo del mAP se resume en varios pasos:

1. Para cada clase de objeto, se genera una curva de precisión-recall al variar el umbral de confianza de las detecciones.
2. La Precisión Promedio (Average Precision, AP) se calcula para cada clase como el área bajo la curva de precisión-recall interpolada. Hay diferentes formas de interpolar esta curva (por ejemplo: interpolación de todos los puntos o interpolación de 11 puntos como en PASCAL VOC).
3. El mAP es simplemente la media de los valores de AP calculados para todas las clases de objetos. Si solo hay una clase (como "estudiante" en este caso), el mAP es igual al AP de esa clase.

- (F1-score)

El F1-Score es la media armónica de la precisión y el recall. Proporciona una única medida que equilibra ambas métricas (Liu et al., 2016). Su fórmula es:

$$F1 = 2 \times \text{Precision} + \text{Recall} \text{ Precision} \times \text{Recall} = 2TP + FP + FN \ 2 \ TP$$

El F1-Score alcanza su valor máximo en 1 (precisión y recall perfectos) y su valor mínimo

en 0. Es particularmente útil cuando se desea saber el equilibrio entre precisión y recall, o cuando se trata de clases desbalanceadas (aunque el mAP ya considera esto al promediar APs por clase). En el contexto de esta tesis, donde solo hay una clase ("estudiante"), el F1-Score ofrece una visión consolidada del rendimiento del modelo en esa clase específica, como complemento del mAP.

Dicho esto, al usar el mAP y el F1 score se logra una evaluación completa de los modelos. El mAP valida la calidad de la localización de las cajas delimitadoras y el rendimiento general de la detección a distintos umbrales, y el F1 score valida el rendimiento de la clasificación y del conteo al evaluar el equilibrio entre precisión y recall. Estas dos métricas garantizan que el modelo evaluado no solo identifique correctamente a los estudiantes, sino que lo haga de manera precisa, completa y robusta.

En la tabla 1.1 podemos notar que las mejores métricas después del entrenamiento las tiene el modelo YOLOv8l con 100 épocas, por lo tanto, es el modelo que vamos a utilizar para realizar nuestras pruebas en ambientes reales.

Tabla 3.1 Métricas de evaluación de los modelos durante el entrenamiento.

| **Modelo** | **Épocas** | **mAP50** | **mAP50-95** | **Precisión** | **Recall** | **F1 score** |
|---|---|---|---|---|---|---|
| YOLOv8n | 10 | 0.88983 | 0.52775 | 0.85825 | 0.83545 | 0.84669 |
| YOLOv8n | 30 | 0.90663 | 0.55572 | 0.87796 | 0.85508 | 0.86636 |
| YOLOv8n | 50 | 0.91308 | 0.55929 | 0.88935 | 0.86649 | 0.87777 |
| YOLOv8n | 100 | 0.91132 | 0.56191 | 0.89248 | 0.86218 | 0.87706 |

| | | | | | | |
|---|---|---|---|---|---|---|
| YOLOv8l | 10 | 0.90357 | 0.54427 | 0.87180 | 0.83238 | 0.85163 |
| YOLOv8l | 30 | 0.91347 | 0.56910 | 0.89787 | 0.85912 | 0.87806 |
| YOLOv8l | 50 | 0.91805 | 0.57689 | 0.90096 | 0.87067 | 0.88555 |
| **YOLOv8l** | **100** | **0.92179** | **0.57412** | **0.90237** | **0.88124** | **0.89168** |

## 3.5 Prueba de los modelos

Para esta etapa de pruebas, se recolectaron 30 imágenes del dataset que se obtuvieron de los videos tomados en entornos reales. De igual manera que el dataset para el entrenamiento de los modelos, se llevaron estas imágenes a una etapa de preprocesamiento. Se aplicaron las mismas técnicas de redimensión dejándolas en una resolución de 640X640 en escala de grises, se consideraron imágenes de auditorio, salones y laboratorio de cómputo. Se llevaron a cabo las pruebas en todos los modelos entrenados de YOLOv8 nano y large con 10, 30, 50 y 100 épocas, utilizando en cada uno las 30 imágenes mencionadas anteriormente, se utilizó un nivel de confianza de 0.3 y se recolectaron las métricas resultantes por modelo; posteriormente por ambiente educativo, dicha información se encuentra reportada en el apartado de resultados y discusión.

## 3.6 Evaluación

Para el proceso de evaluación se recopilaron las métricas resultantes de las pruebas realizadas de cada uno de los modelos con 30 imágenes tomadas de videos de ambientes educativos reales, se hizo la comparación por modelo nano y large así como de las épocas en las que fueron entrenadas, se realizó una tabla comparativa para cada modelo, se realizó también una tabla de métricas por cada imagen, además se calculó el promedio de las métricas por cada ambiente (salón, auditorio y laboratorio). Finalmente, después del análisis minucioso de las métricas, se redactaron las conclusiones acerca de los modelos.

## 3.7 Conteo

Para el proceso de conteo en las detecciones hechas en las imágenes, se llevó a cabo, en primer lugar, la clasificación de estas, es decir, se separaron por espacio educativo: auditorio, laboratorio y salón de clases. En segundo lugar, se realizó el conteo manual en todas las imágenes para poder obtener el Ground_truth (Conteo real), posteriormente para obtener el número de predicciones se corrió el modelo de YOLOv8 large entrenado en 100 épocas y éste proporcionó el número de estudiantes detectados por imagen, las métricas restantes son TP,TN, FP y FN, y con ellas se calculó el Precision, Recall y F1-score.

# CAPÍTULO IV RESULTADOS Y DISCUSIÓN

## 4.1 Resultados

En este capítulo se presentan los resultados obtenidos en esta investigación; para ello, se realizó un análisis basado en las métricas de detección de objetos reportados en la literatura. La discusión de estos resultados se centrará en interpretar su significado, comparar con el estado del arte, evaluar el impacto del trabajo metodológico propuesto para verificar la hipótesis de investigación.

Después del entrenamiento y la evaluación en los modelos nano y large con 10, 30, 50 y 100 épocas, se compararon las métricas de precisión, recall y F1 score obtenidas para cada uno de ellos, las cuales se muestran en la figura 4.1.

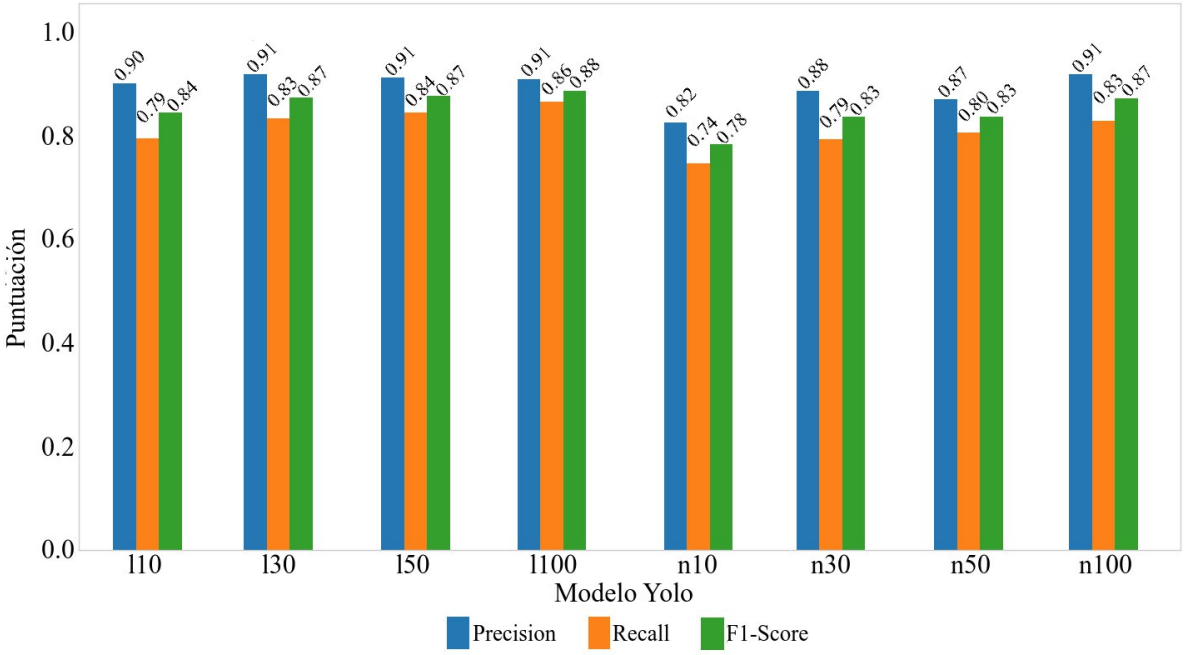
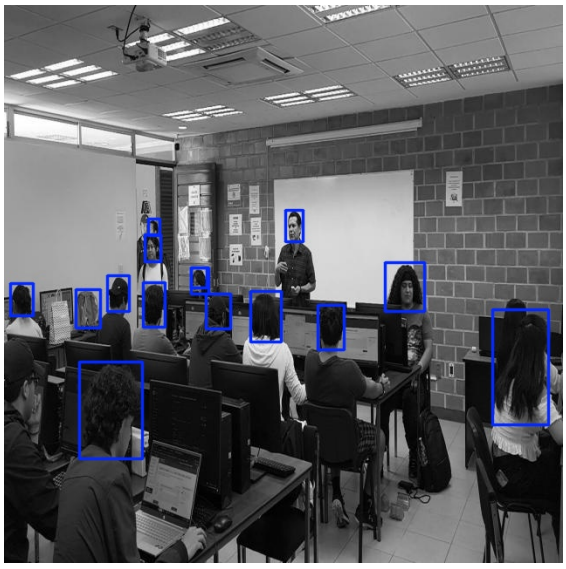



Figura 4.1 Comparación de las métricas en el periodo de prueba de todos los modelos entrenados.

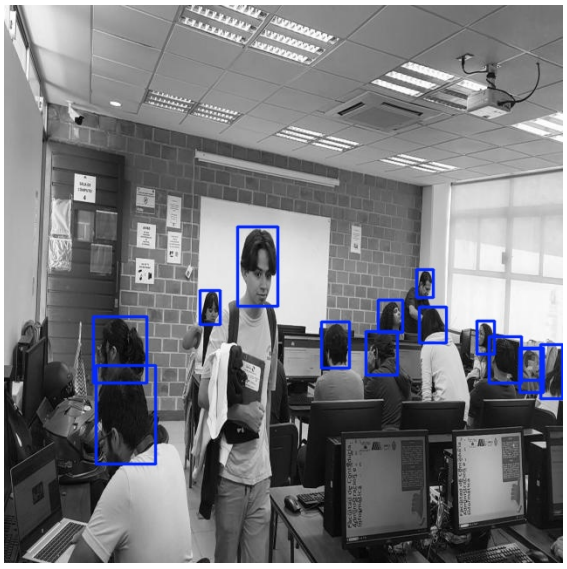
Los resultados de la figura 4.1 muestran que la arquitectura YOLOv8 Large es superior a la YOLOv8 Nano. El modelo con el mejor rendimiento es el YOLOv8 Large entrenado con 100 épocas (l100), ya que la Precision lograda fue la más alta (0.91) y el F1-Score más alto

(0.88); sin embargo, el modelo large con 50 épocas (l50) casi iguala en la métrica de F1-Score (0.87) al modelo large (l100) con un número menor de iteraciones de entrenamiento. El modelo Large con 30 épocas (l30) presenta el valor de recall más alto después de (l100), con 0.87, lo que es relevante en aplicaciones donde minimizar los falsos negativos es una prioridad. En cuanto al modelo de tamaño nano, el mejor fue el entrenado con 100 épocas (n100), con 0.91 de Precision, 0.83 de Recall y 0.87 de F1 Score.

Es fundamental observar el comportamiento de los modelos en imágenes reales para comprender sus fortalezas y debilidades. A continuación, se muestran ejemplos visuales en diferentes escenarios del conjunto de prueba:

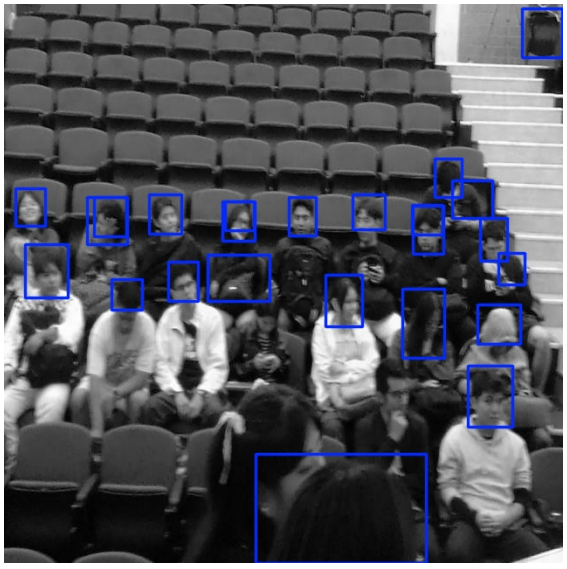


(a) Laboratorio parte trasera derecha 1

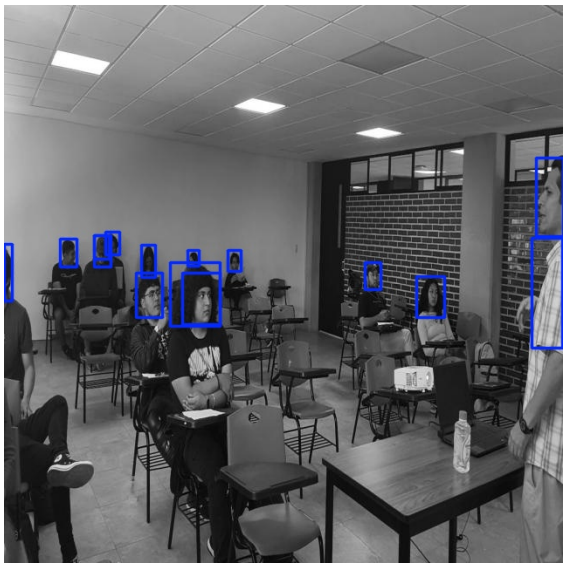


(b) Laboratorio parte trasera izquierda 1

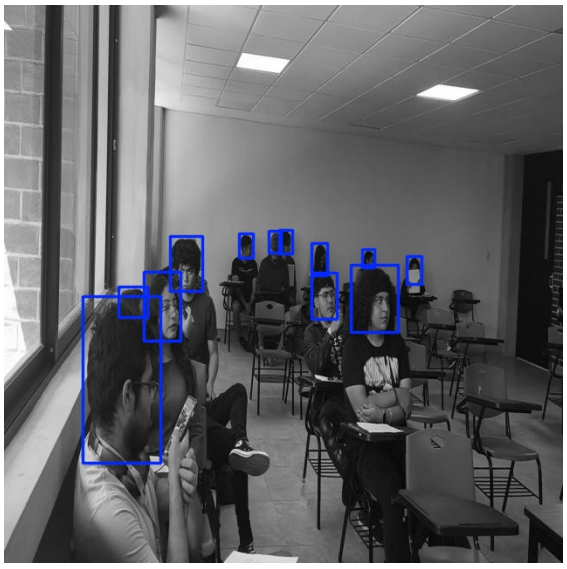
(c) Auditorio frente 1



(d) Auditorio costado izquierdo 1

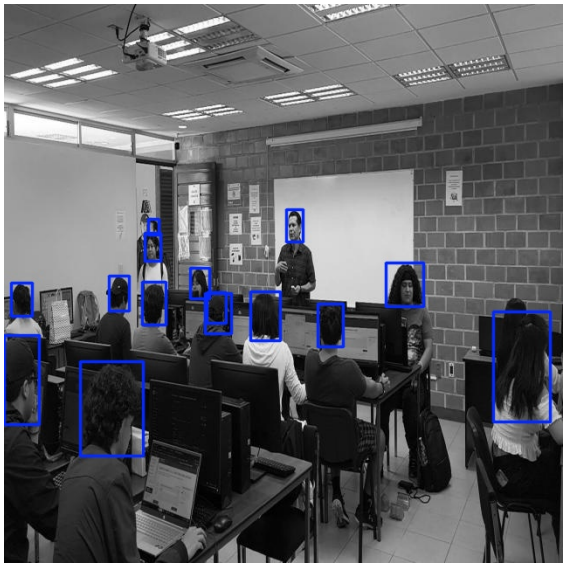


(e) Salón frente izquierdo 1

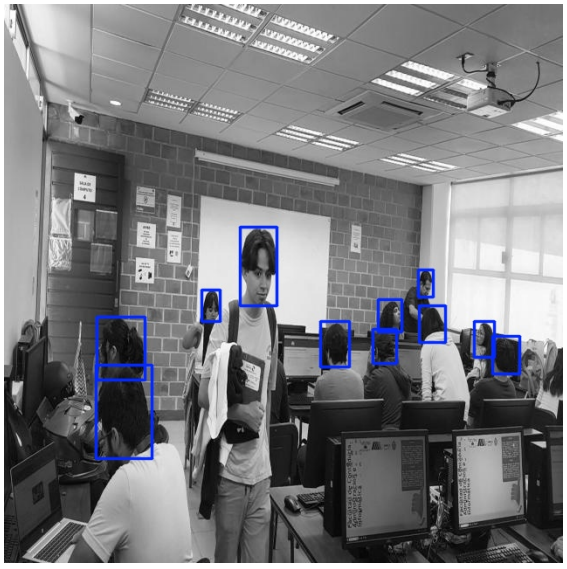


(f) Salón detalle lado izquierdo 1

Figura 4.2 Imágenes de predicciones del modelo YOLOv8 nano con 10 épocas de entrenamiento

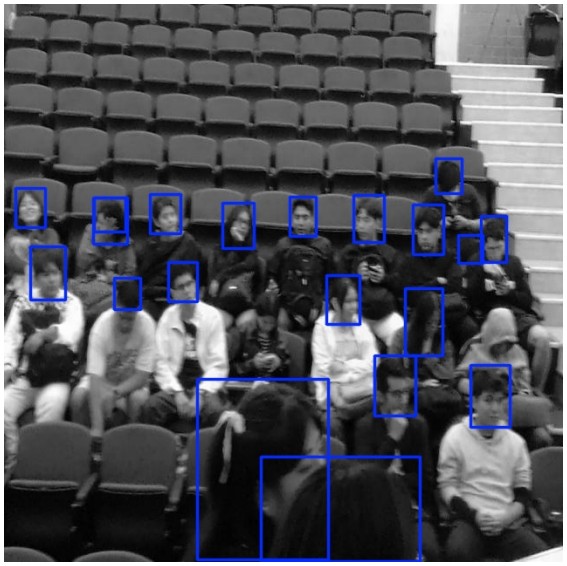


(a) Laboratorio parte trasera derecha 2

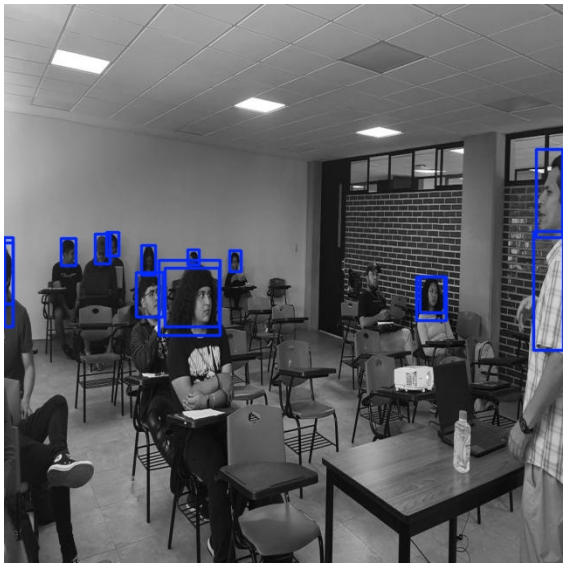


(b) Laboratorio parte trasera izquierda 2

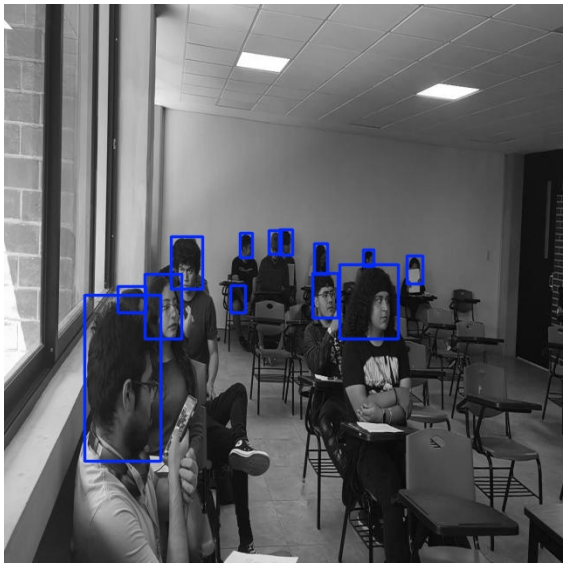
(c) Auditorio frente 2



(d) Auditorio costado izquierdo 2

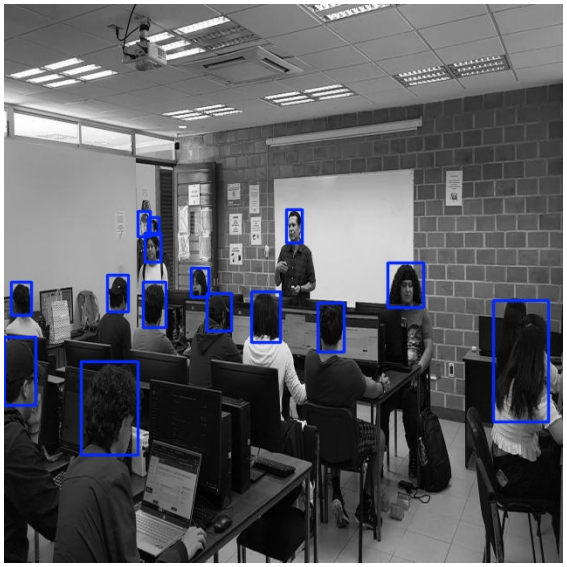


(e) Salón frente izquierdo 2

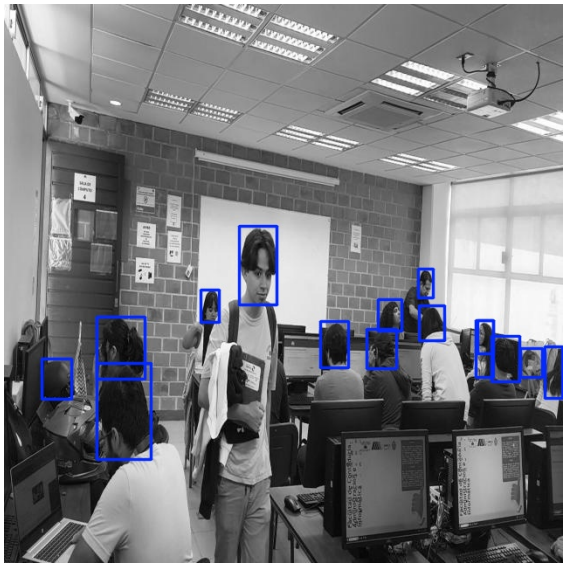


(f) Salón detalle lado izquierdo 2

4.3 Imágenes de predicciones del modelo YOLOv8 large con 10 épocas de entrenamiento.

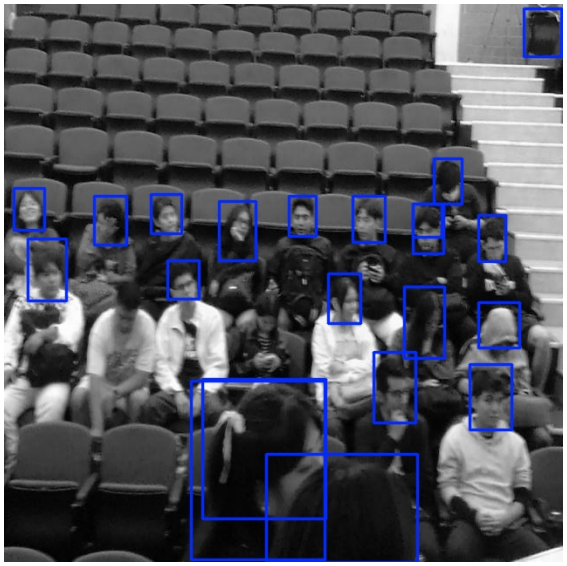


(a) Laboratorio parte trasera derecha 3

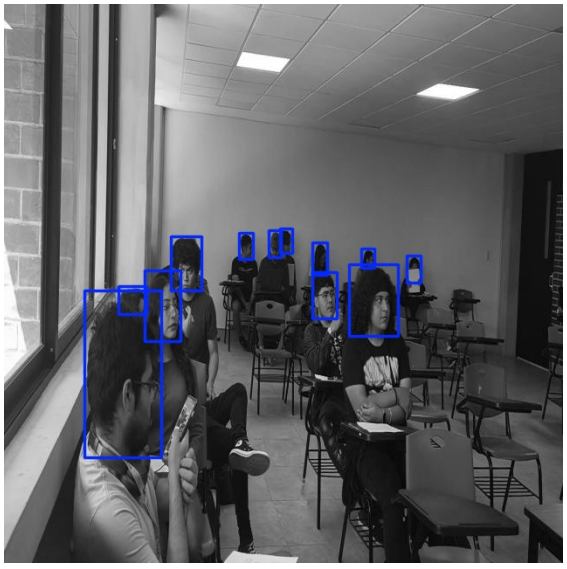


(b) Laboratorio parte trasera izquierda 3

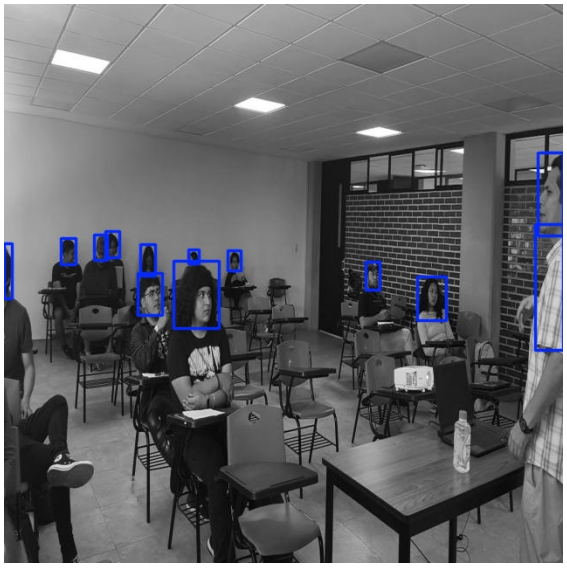
(c) Auditorio frente 3



(d) Auditorio costado izquierdo 3

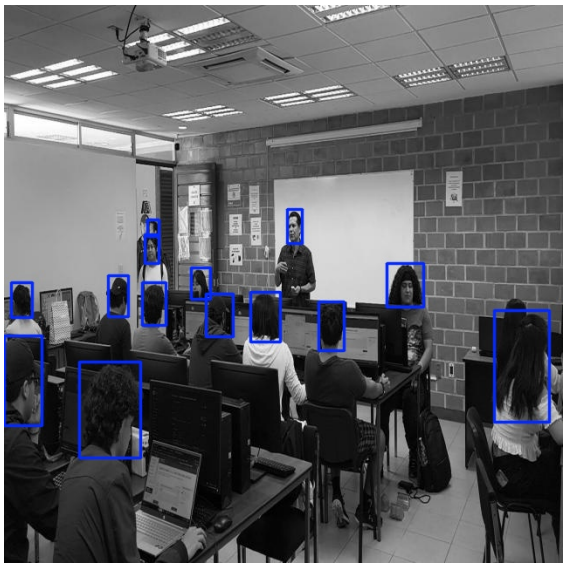


(e) Salón frente izquierdo 3

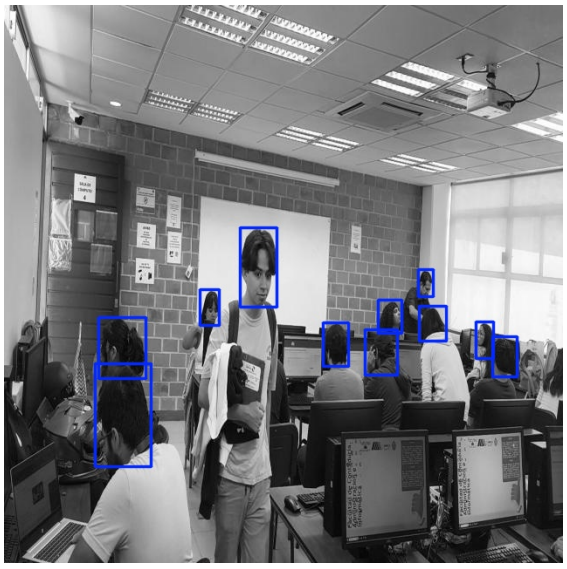


(f) Salón detalle lado izquierdo 3

4.4 Imágenes de predicción del modelo YOLOv8 nano con 30 épocas de entrenamiento.

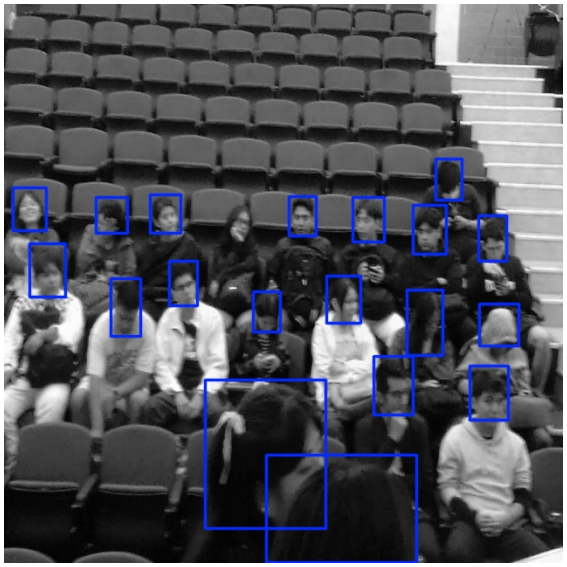


(a) Laboratorio parte trasera derecha 4

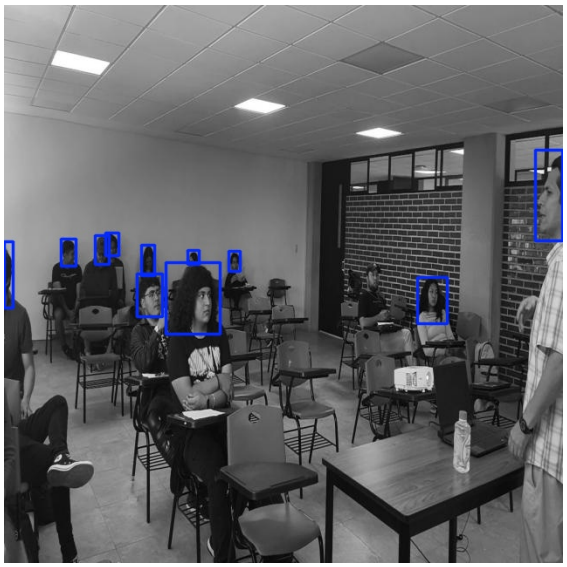


(b) Laboratorio parte trasera izquierda 4

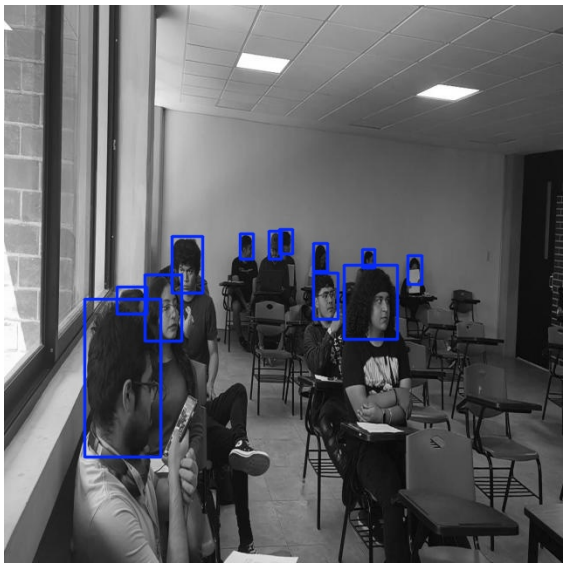
(c) Auditorio frente 4



(d) Auditorio frente costado izquierdo 4

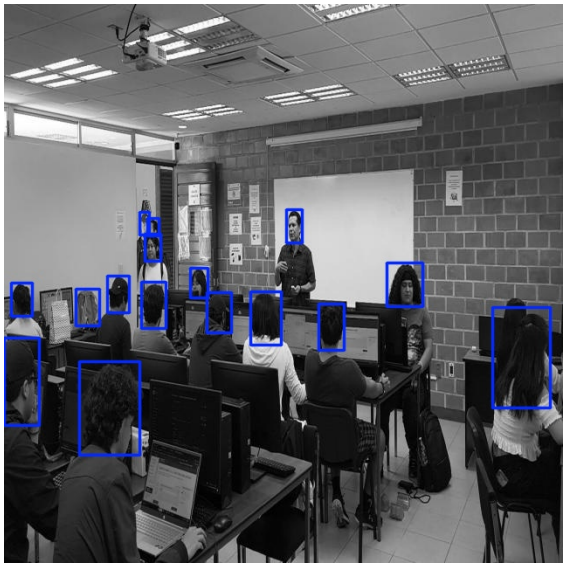


(e) Salón frente lado izquierdo 4

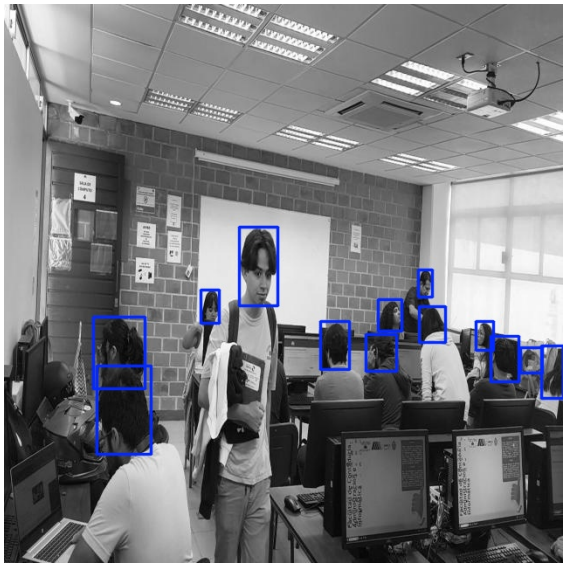


(f) Salón detalle lado izquierdo 4

4.5 Imágenes de predicción del modelo YOLOv8 large con 30 épocas de entrenamiento.

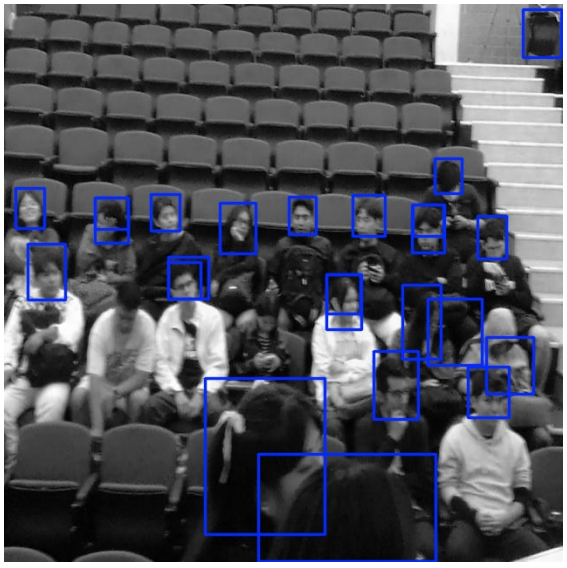


(a) Laboratorio parte trasera derecha 5

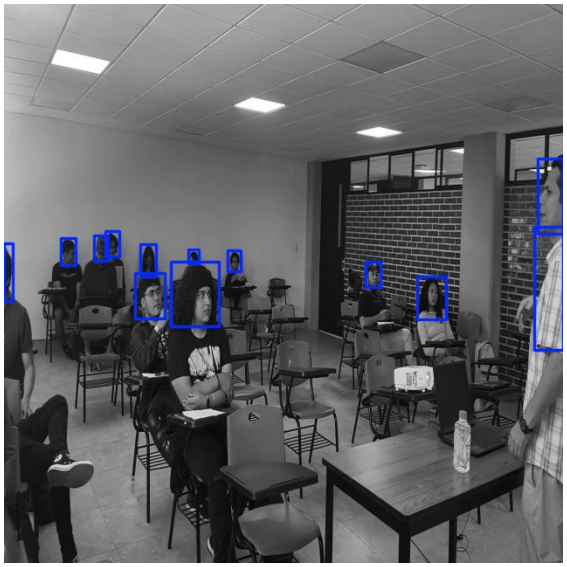


(b) Laboratorio parte trasera izquierda 5

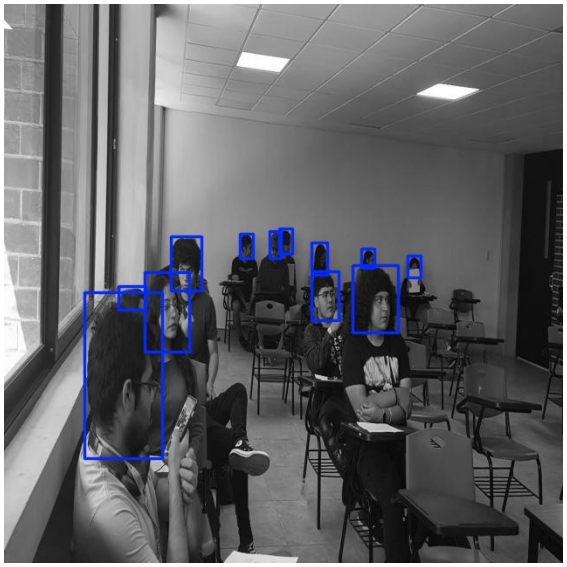
(c) Auditorio frente 5



(d) Auditorio frente costado izquierdo 5

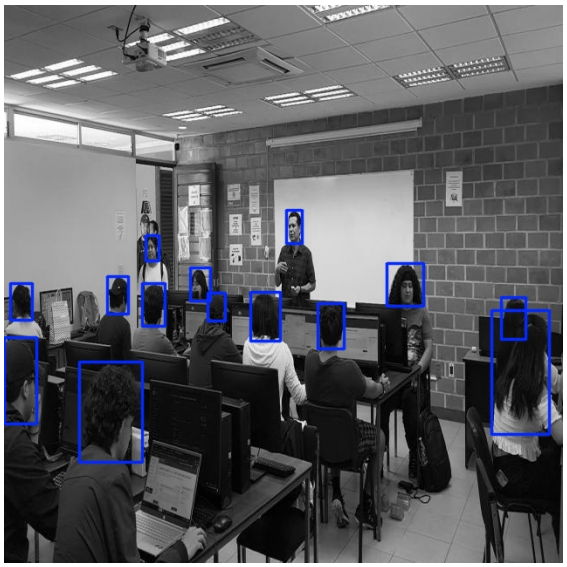


(e) Salón frente lado izquierdo 5

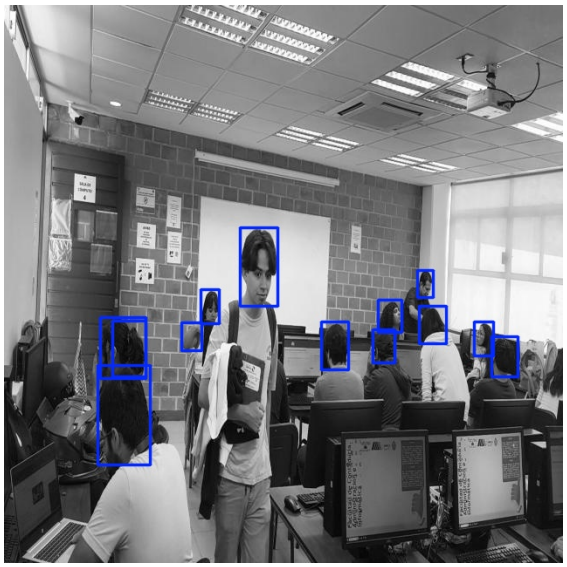


(f) Salón detalle lado izquierdo 5

4.6 Imágenes de predicciones del modelo YOLOv8 nano con 50 épocas de entrenamiento.

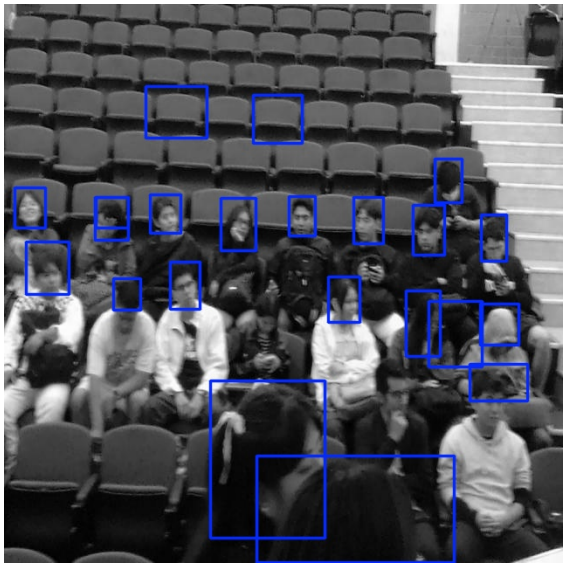


(a) Laboratorio parte trasera derecha 6

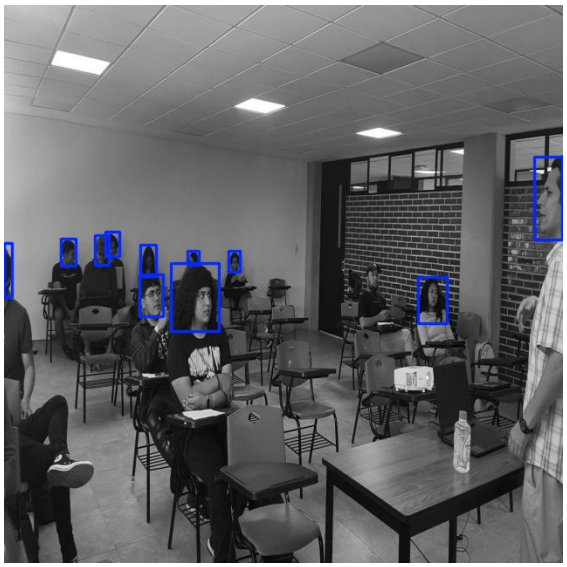


(b) Laboratorio parte trasera izquierda 6

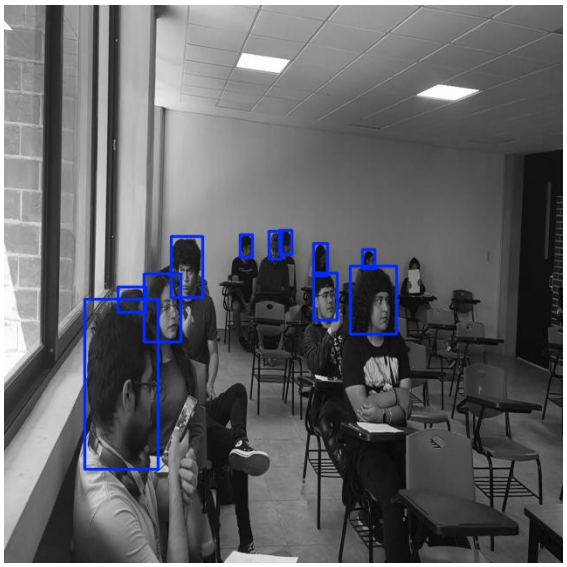
(c) Auditorio frente 6



(d) Auditorio costado derecho izquierdo 6

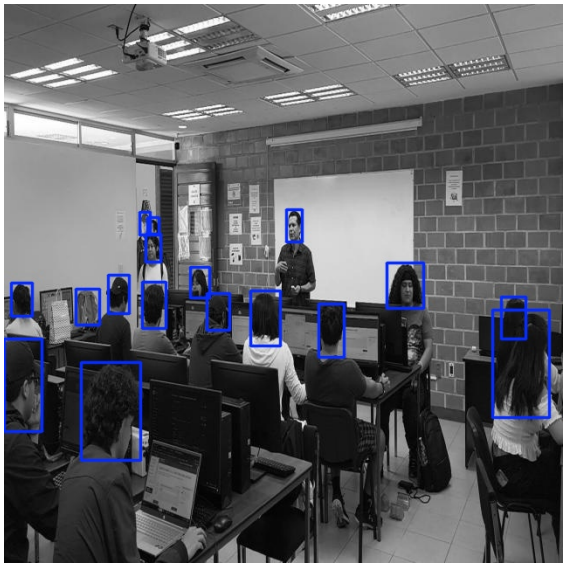


(e) Salón frente lado izquierdo 6

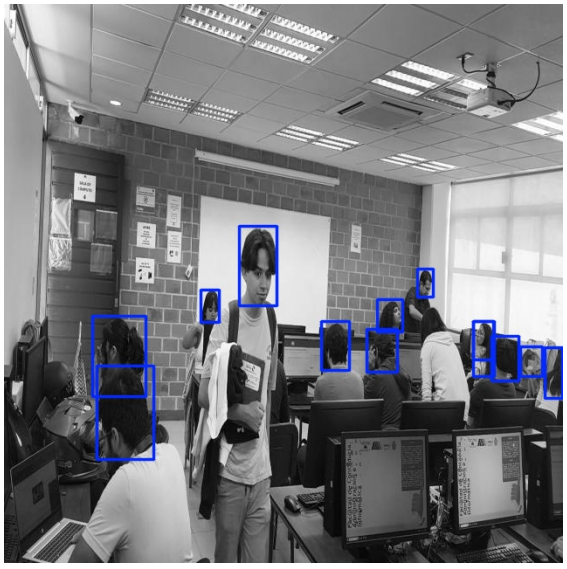


(f) Salón detalle lado izquierdo 6

4.7 Imágenes de predicción del modelo YOLOv8 large con 50 épocas de entrenamiento.

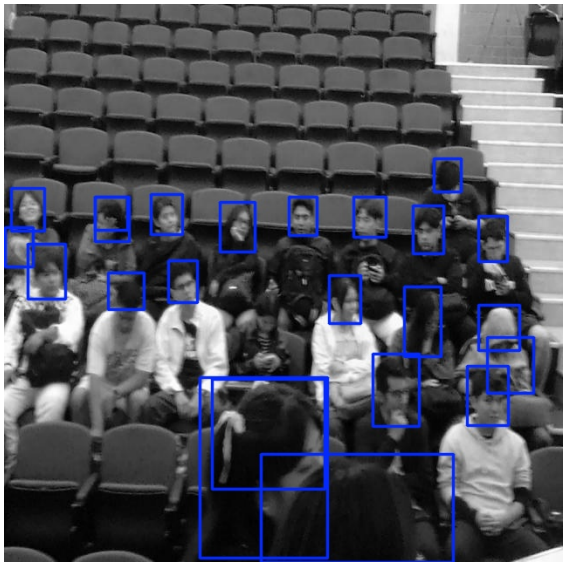


(a) Laboratorio parte trasera derecha 7

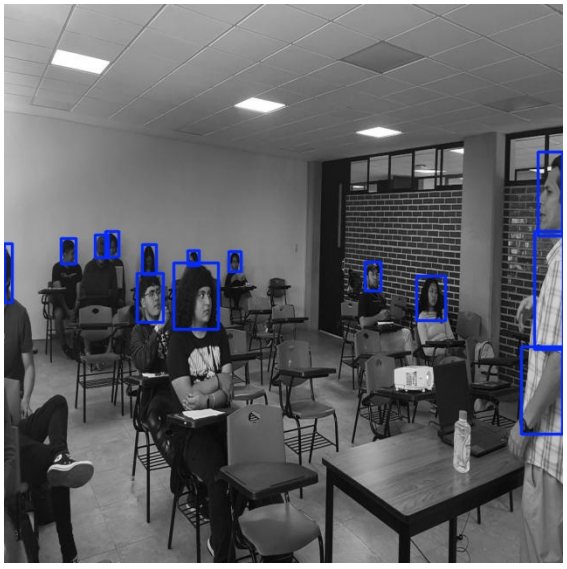


(b) Laboratorio parte trasera izquierda 7

(c) Auditorio frente 7



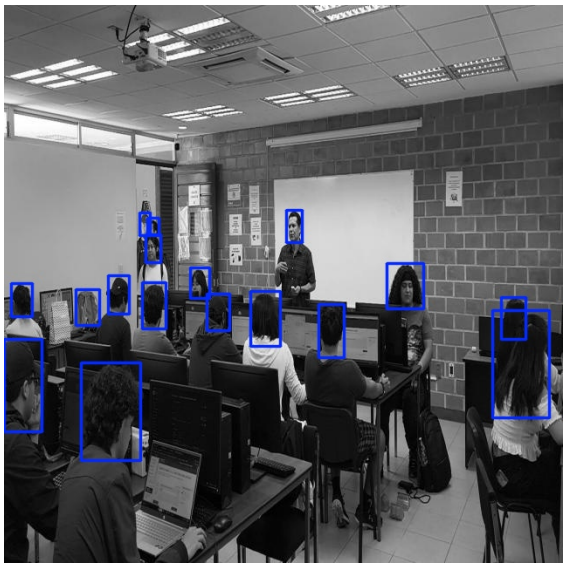
(d) Auditorio frente costado izquierdo 7



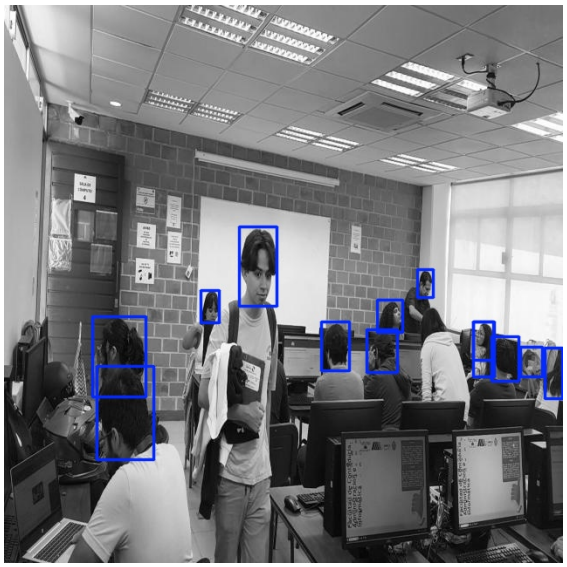
(e) Salón frente lado izquierdo 7

(f) Salón detalle lado izquierdo 7

4.8 Imágenes de predicción del modelo YOLOv8 nano con 100 épocas de entrenamiento.

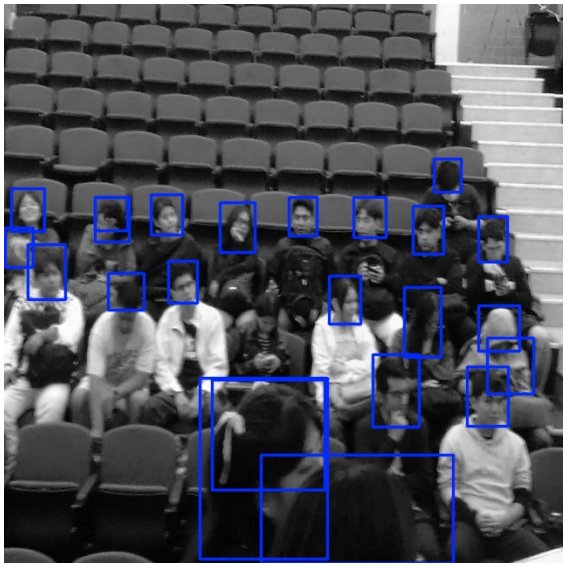


(a) Laboratorio parte trasera derecha 8

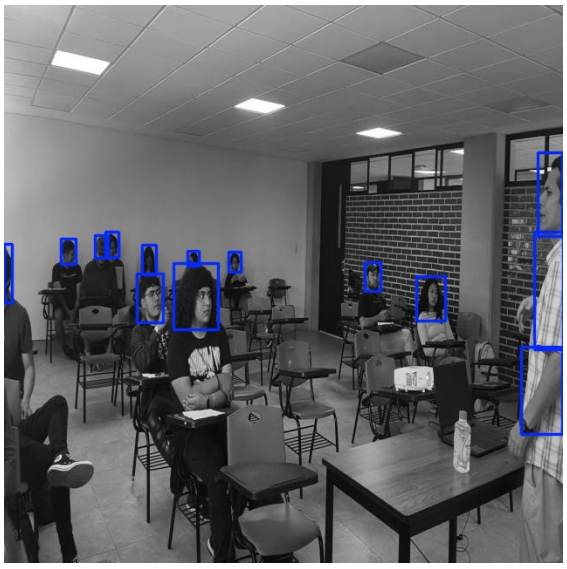


(b) Laboratorio parte trasera izquierda 8

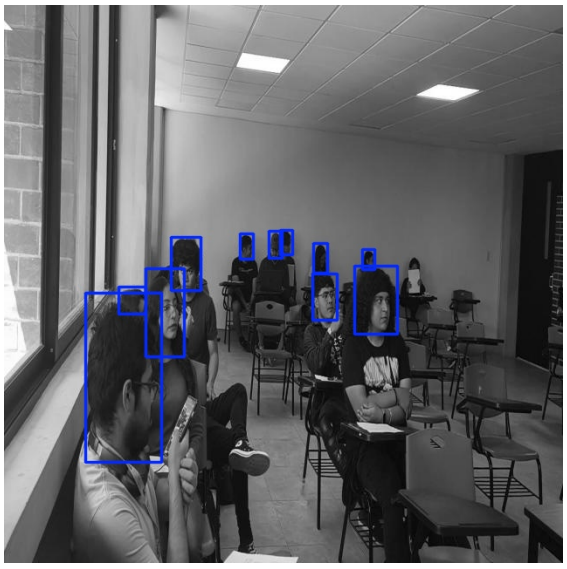
(c) Auditorio frente 8



(d) Auditorio costado izquierdo 8



(e) Salón frente lado izquierdo 8



(f) Salón detalle lado izquierdo 8

4.9 Imágenes de predicción del modelo YOLOv8 large con 100 épocas de entrenamiento.

En estas imágenes de ejemplo de detección de las pruebas de cada modelo, en sus dos tamaños (nano y large), se presentan casos de oclusión en las figuras (a) Laboratorio parte trasera derecha, (b) Laboratorio parte trasera izquierda, (d) auditorio costado izquierdo y (f) salón detalle lado izquierdo, en las cuales los alumnos fueron detectados adecuadamente en todas las pruebas arriba realizadas.

En la figura 4.10 se observa que para ambas versiones utilizadas (YOLOv8n y YOLOv8l), hay una tendencia de mejora en las métricas; a medida que aumenta el número de épocas de entrenamiento, mejora la Precision en la detección. Por ejemplo, para YOLOv8n, el mAP50 aumenta de 0.88983 (10 épocas) a 0.91308 (50 épocas). Se puede apreciar también que en el caso de YOLOv8n, el rendimiento en mAP50 y F1-Score parecen estancarse e incluso disminuir ligeramente al pasar de 50 a 100 épocas, esto evidencia un sobreajuste o bien que el modelo ya alcanzó su capacidad máxima de aprendizaje con este dataset (Terven et al., 2023), por otro lado, el modelo YOLOv8l continúa mejorando hasta las 100 épocas en la mayoría de las métricas.

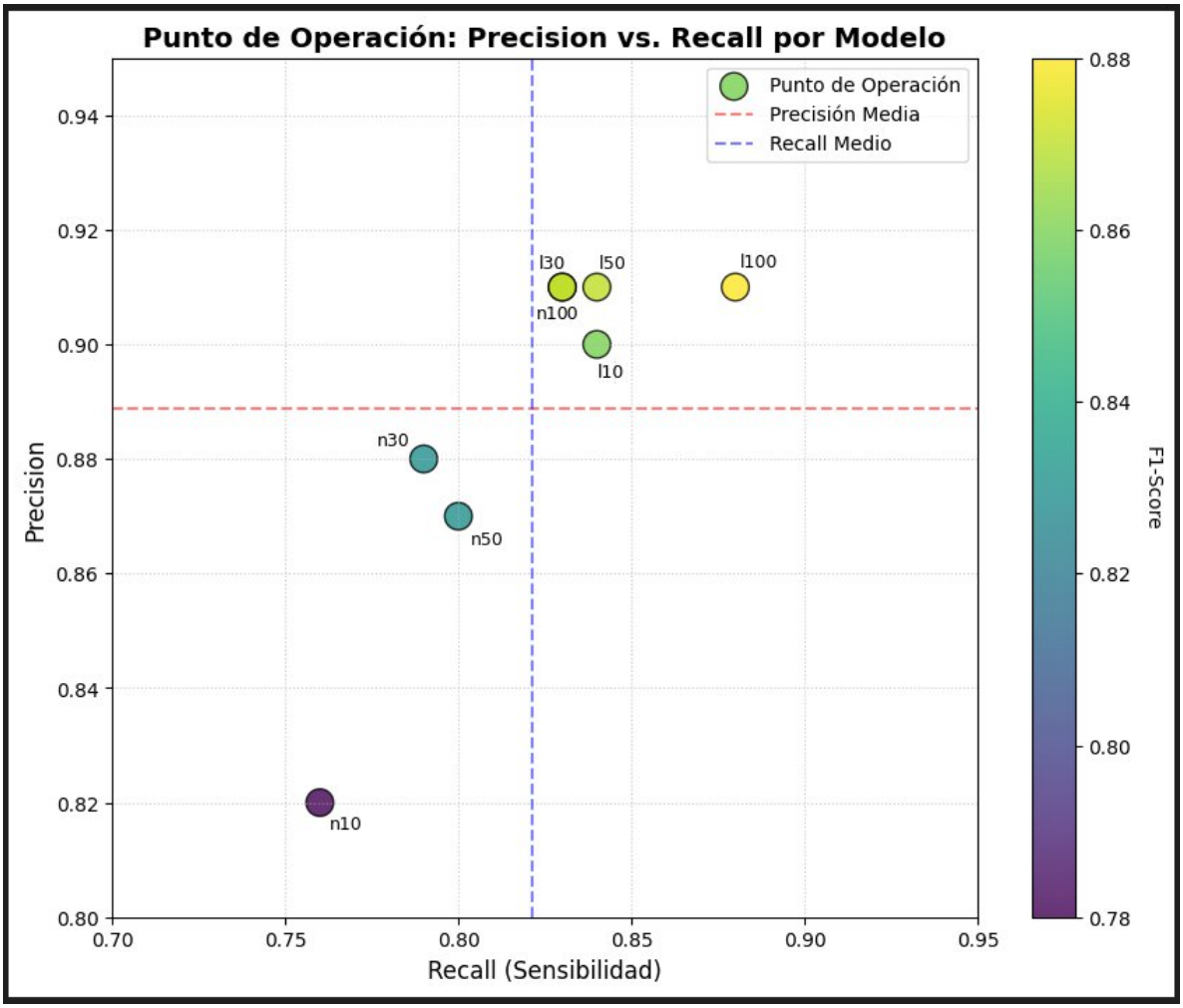



Figura 4.10 Se muestra el rendimiento de todos los modelos mencionados anteriormente; en esta gráfica se nota claramente que el modelo YOLOv8l de 100 épocas es el de mayor

rendimiento, seguido de cerca de modelos como el “l30” y “l50” (véase cuadrante superior derecho).

La versión del modelo large supera a la variante nano en todas las métricas evaluadas para un mismo número de épocas; esto es de esperar ya que los modelos más grandes tienen mayor capacidad para aprender representaciones de características más complejas (Terven et al., 2023) aunque a costa de una mayor carga computacional y tiempos potencialmente más largos.

En cuanto al conteo, la tabla 4.1 muestra que el modelo YOLOv8 Large, entrenado durante 100 épocas, ofrece un rendimiento general muy sólido y preciso, destacando especialmente en los entornos de "salón" con 0.9616 de Precision, 0.9603 de Recall, 0.9605 de F1-score y "laboratorio" con 0.9640 de Precision, 0.9491 de Recall y 0.8588 de F1-Score. En estas categorías, el modelo logra F1-Scores perfectos (1.0000) en 13 de 30 imágenes, lo que se traduce en una detección adecuada, sin omitir personas ni generar falsas alarmas. En cuanto en las imágenes tomadas en el auditorio disminuyen estas métricas debido a la cantidad de personas presentes ya que representa un ambiente complejo resultando en métricas como 0.8429 de Precision, 0.8768 de Recall y 0.8638 de F1-Score.

Tabla 4.1 Métricas de conteo de estudiantes por imagen.

| Imagen | Ground_truth | Predicciones | TP | FP | FN | Precision | Recall | F1- Score |
|---|---|---|---|---|---|---|---|---|
| auditorio 1.jpg | 158 | 128 | 99 | 29 | 59 | 0.7734 | 0.6266 | 0.6923 |
| auditorio2.jpg | 151 | 143 | 138 | 5 | 13 | 0.9650 | 0.9139 | 0.9388 |
| auditorio 3.jpg | 6 | 6 | 6 | 0 | 0 | 1.0000 | 1.0000 | 1.0000 |
| auditorio 4.jpg | 4 | 5 | 4 | 1 | 0 | 0.8000 | 1.0000 | 0.8889 |
| auditorio 5.jpg | 39 | 42 | 39 | 3 | 0 | 0.9286 | 1.0000 | 0.9630 |
| auditorio 6.jpg | 39 | 42 | 35 | 7 | 4 | 0.8333 | 0.8974 | 0.8642 |
| auditorio 7.jpg | 20 | 20 | 14 | 6 | 6 | 0.7000 | 0.7000 | 0.7000 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Promedio auditorio** | | | | | | **0.8429** | **0.8768** | **0.8638** |
| laboratorio1.jpg | 16 | 15 | 15 | 0 | 1 | 1.0000 | 0.9375 | 0.9677 |
| laboratorio2.jpg | 12 | 11 | 11 | 0 | 1 | 1.0000 | 0.9167 | 0.9565 |
| laboratorio3.jpg | 14 | 15 | 14 | 1 | 0 | 0.9333 | 1.0000 | 0.9655 |
| laboratorio4.jpg | 16 | 15 | 15 | 0 | 1 | 1.000 | 0.9375 | 0.9677 |
| laboratorio5.jpg | 10 | 11 | 9 | 2 | 1 | 0.8182 | 0.9000 | 0.8571 |
| laboratorio6.jpg | 12 | 12 | 12 | 0 | 0 | 1.0000 | 1.0000 | 1.0000 |
| laboratorio7.jpg | 16 | 16 | 16 | 0 | 0 | 1.0000 | 1.0000 | 1.0000 |
| laboratorio8.jpg | 10 | 9 | 8 | 1 | 2 | 0.8888 | 0.8000 | 0.8421 |
| laboratorio9.jpg | 16 | 16 | 16 | 0 | 0 | 1.0000 | 1.0000 | 1.0000 |
| laboratorio10.jpg | 13 | 13 | 13 | 0 | 0 | 1.0000 | 1.0000 | 1.0000 |
| **Promedio Laboratorio** | | | | | | **0.9640** | **0.9491** | **0.8588** |
| salón 1.jpg | 10 | 11 | 10 | 1 | 0 | 0.9091 | 1.0000 | 0.9524 |
| salón 2.jpg | 14 | 14 | 14 | 0 | 0 | 1.0000 | 1.0000 | 1.0000 |
| salón 3.jpg | 14 | 14 | 14 | 0 | 0 | 1.0000 | 1.0000 | 1.0000 |
| salón 4.jpg | 12 | 12 | 11 | 1 | 1 | 0.9167 | 0.9167 | 0.9167 |
| salón 5.jpg | 11 | 11 | 10 | 1 | 1 | 0.9091 | 0.9091 | 0.9091 |
| salón 6.jpg | 11 | 11 | 10 | 1 | 1 | 0.9091 | 0.9091 | 0.9091 |
| salón 7.jpg | 8 | 8 | 8 | 0 | 0 | 1.0000 | 1.0000 | 1.0000 |
| salón 8.jpg | 8 | 7 | 6 | 1 | 2 | 0.8571 | 0.7500 | 0.8000 |
| salón 9.jpg | 6 | 6 | 6 | 0 | 0 | 1.0000 | 1.0000 | 1.0000 |
| salón 10.jpg | 6 | 6 | 6 | 0 | 0 | 1.0000 | 1.0000 | 1.0000 |
| salón 11.jpg | 6 | 6 | 6 | 0 | 0 | 1.0000 | 1.0000 | 1.0000 |
| salón 12.jpg | 6 | 6 | 6 | 0 | 0 | 1.0000 | 1.0000 | 1.0000 |
| salón 13.jpg | 6 | 6 | 6 | 0 | 0 | 1.0000 | 1.0000 | 1.0000 |
| **Promedio salón** | | | | | | **0.9616** | **0.9603** | **0.9605** |

Podemos mencionar que la principal contribución de este trabajo se encuentra en la adaptación, entrenamiento y validación del modelo YOLOv8, para las tareas de detectar y contar estudiantes en diferentes ambientes educativos, mediante la detección de cabezas en

escenarios reales incluyendo salones, laboratorio de cómputo y un auditorio (FCAeI de la UAEM). La metodología y el desarrollo de un dataset propio, forman parte de esta contribución ya que demostraron ser efectivos para resolver retos como la oclusión y las diferentes condiciones de iluminación.

Si bien en este trabajo se demostró un rendimiento excelente en varios escenarios de prueba, las limitaciones se presentaron principalmente en las imágenes con niveles de ocupación altos (multitudes), por ejemplo en las imágenes auditorio1.jpg y auditorio7.jpg, en ambos casos se observa una disminución del recall lo que significa que el modelo falla en la detección de los estudiantes que ahí se muestran, esto es de esperarse ya que YOLOv8 no es una arquitectura especializada en multitudes como sí lo es CSRNet (Li et al., 2018), que utiliza mapas de densidad.

Las limitaciones mencionadas en el Capítulo I, como la iluminación baja y la escasa resolución de las imágenes, siguen siendo factores clave que contribuyen a la reducción de la precisión del modelo.

## 4.2 Comparación con el estado del arte.

El rendimiento del modelo desarrollado en este trabajo de tesis se sitúa de manera competitiva con otros trabajos de investigación en el estado del arte:

Con respecto del sistema basado en un detector personalizado de Chen et al. (2020), que alcanzó una precisión en el conteo de 89.47%, 88.35% de Recall y un F1-Score de 88.91%, el modelo de este trabajo de tesis (YOLOv8l con 100 épocas) superó las métricas de precisión con 91% y el F1-Score esta cerca con un 88%.

Tabla 4.2 Tabla comparativa con el estado del arte.

| Autores | Título | Base de datos | Modelo | Ambiente | Resultados |
|---|---|---|---|---|---|
| **Meza,Hernández 2025** | **Conteo de estudiantes en ambiente educativos usando CNNs preentrenadas** | **Classroom monitoring dataset. Base de datos propia** | **YOLOv8** | **Salón, Laboratorio y Auditorio** | **YOLOv8l: Precision = 91% Recall = 86% F1-Score = 88%** |
| Chen, Jin, Xu 2020 | A Classroom student counting system based on improved context-based face detector | Imágenes de videos de vigilancia | YOLOv3 y Deepface | Salón de clases | Precision= 89.47% Recall = 88.35% F1-Score= 88.91% |
| Hernández, et. al. 2024 | Intelligent Student counting system tolerant to occlussions based on convolutional neural networks | Base de datos obtenida en Roboflow | YOLOv5n y YOLOv5l | Laboratorio | YOLOv5n: Recall = 88% Precision=100% YOLOv5l: Recall = 99% Precision = 100% |
| Saporana, Elhanashi, Gagliardi 2021 | Implementing a realtime, AI-based, people detection and social distancing measuring system for COVID-19 | Dataset I/ Dataset II | YOLOv2, MATLAB | Escenarios exteriores e interiores | Accuracy = 95.6% Precision = 95% Recall = 96% |
| Hernández, et. al. 2023 | A new approach for counting and identification of students' sentiments in online virtual environments using convolutional neural networks | 50 imágenes obtenidas de clases en línea | YOLOv3 y Deepface | Clase en línea | Precision = 96.7% Average Precision = 96.7% |

La Tabla 4.2 compara las métricas de la presente investigación con el estado del arte reciente (2020-2024), destacando la implementación de la arquitectura YOLOv8 como una actualización tecnológica significativa respecto de modelos anteriores (YOLOv2, v3 y v5) predominantes en la literatura revisada. Haciendo el comparativo, podemos notar una diferencia fundamental en el alcance de la evaluación: mientras que trabajos previos como los de Chen et al. (2020) Hernández et al. (2024) limitan su validación a un único entorno controlado (salón o laboratorio, respectivamente), esta propuesta de investigación demuestra una capacidad de generalización superior al ser puesta a prueba en tres escenarios con diferentes casos de reto: salón, laboratorio y auditorio. En este sentido, el F1-Score global

del 88% obtenido no solo es competitivo frente a las métricas de entornos aislados, sino que también respalda la robustez y versatilidad del modelo para operar eficazmente en una diversidad de infraestructuras educativas reales, superando la limitación del sobreajuste a un solo tipo de escenario que presentan otras soluciones.

# CONCLUSIONES Y TRABAJO FUTURO

## Conclusiones

En este trabajo de investigación se propuso implementar modelo CNN preentrenado para la detección y conteo de estudiantes en ambientes educativos, para la FCAeI de la UAEM. Con los resultados obtenidos y discutidos anteriormente, podemos llegar a las siguientes conclusiones:

Se logró implementar un modelo para la detección y conteo de estudiantes utilizando YOLOv8, esto cumple con el objetivo general de la tesis. La metodología fue desarrollada desde la adquisición y el preprocesamiento del dataset hasta el entrenamiento y la evaluación del modelo, el cual demostró ser funcional.

En este trabajo de investigación se planteó la siguiente hipótesis nula:

H0: Se puede detectar y contar de manera eficiente el número de estudiantes en un espacio educativo cerrado, utilizando CNNs preentrenadas en imágenes captadas por cámaras de video.

Los resultados obtenidos con el modelo YOLOv8 large entrenado con 100 épocas demuestra una capacidad notable para detectar estudiantes en diferentes tipos de espacios educativos, lo cual permite su conteo de manera eficiente y precisa dentro del contexto del dataset.

Si tomamos en cuenta que el objetivo de alcanzar métricas de precisión de detección de al menos 90% fue superado en términos de mAP50 (92.179%) y un Precision de (90.237%), se concluye que los resultados obtenidos respaldan la hipótesis nula (H0). Específicamente en la tarea de conteo, el modelo demostró una precisión perfecta en más del 44% de las imágenes de prueba de salón y de laboratorio (tabla 3.2), lo que indica una capacidad de conteo eficiente en escenarios que cuentan con baja o mediana densidad.

## Trabajos futuros

Como trabajo futuro, es deseable mejorar la robustez del modelo para extender la aplicabilidad del sistema; para esto, se proponen los siguientes trabajos de investigación:

Modelos de densidad para abordar la limitación observada en escenarios como los auditorios, se podría implementar un modelo como CSRNet combinado con algún otra herramienta de VC para abordar este problema, ya que son más robustas y adecuadas para las imágenes con alta densidad de personas.

Implementar un modelo de conteo en tiempo real en dispositivos de bajo consumo como Raspberry Pi o NVIDIA Jetson para su prueba y validación.

La información proporcionada por el modelo puede ser usada para ser analizada por medio de estadísticas y ayudar en la toma de decisiones informadas y en la planificación estratégica de recursos en la FCAeI de la UAEM.

# Referencias


- **A guide to Inception Model in Keras.** (2019, 5 marzo). https://maelfabien.github.io/deeplearning/inception/
- **Alonso, R.** (2023, 15 septiembre). IA, Machine learning y Deep Learning, ¿cuál es la diferencia? HardZone. https://hardzone.es/tutoriales/rendimiento/diferencias-ia-deep-machine-learning/
- **Arkin, E., Yadikar, N., Xu, X., Aysa, A., & Ubul, K.** (2022). A survey: object detection methods from CNN to transformer. Multimedia Tools And Applications, 82(14), 21353-21383. https://doi.org/10.1007/s11042-022-13801-3
- **Briega, R. E. L.** (s. f.-b). Visión por computadora - libro online de IAAR. https://iaarbook.github.io/vision-por-computadora/
- **Chen, R., Jin, Y., & Xu, L.** (2020). A classroom student counting system based on improved context-based face detector. In Web Information Systems and Applications: 17th International Conference, WISA 2020, Guangzhou, China, September 23–25, 2020, Proceedings 17 (pp. 326-332). Springer International Publishing.
- **Chollet, F.** (2018). Deep learning with Python. https://bvbr.bib-bvb.de:443/F?func=service&doc_library=BVB01&local_base=BVB01&doc_number=029957611&sequence=000001&line_number=0001&func_code=DB_RECORDS&service_type=MEDIA
- **Classroom Monitoring Dataset.** (2021, 15 junio). Kaggle. https://www.kaggle.com/datasets/lunarwhite/classroom-monitoring-dataset
- **Das, A.** (2025, 26 febrero). VGG Architecture Deep layers revolutionize image recognition. Arunangshu Das. https://arunangshudas.com/blog/vgg-architecture-deep-layers-revolutionize/
- **Fagbuyiro, D.** (2024, 28 noviembre). Guide To Transfer Learning in Deep Learning - David Fagbuyiro - Medium. Medium. https://medium.com/@davidfagb/guide-to-transfer-learning-in-deep-learning-1f685db1fc94
- **G. Howard, A., Zhu, M., & Chen, B.** (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. Google Inc.
- **Ge, Z., Liu, S., Wang, F., Li, Z., & Sun, J.** (2021). YOLOX: Exceeding YOLO Series in 2021. arXiv preprint arXiv:2107.08430.
- **Getin.** (2023, 28 septiembre). La importancia del contador científico de personas en la era del comercio omnicanal - Getin. https://getin.mx/blog/contador-cientifico-personas-en-comercio-omnicanal/
- **Goodfellow, I., Bengio, Y., & Courville, A.** (2016). Deep learning. En MIT Press eBooks. https://dl.acm.org/citation.cfm?id=3086952
- **Great Learning.** (2025, 11 febrero). Introduction to VGG16 – What is VGG16? Great Learning Blog. https://www.mygreatlearning.com/blog/introduction-to-vgg16/
- **He, K., Zhang, X., Ren, S., & Sun, J.** (2016). Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770-778. https://doi.org/10.1109/CVPR.2016.90
- **Hermanto, F., & Chandra, W.** (2023). Face Recognition Implementation as an Attendance Feature on Web-Based Video Conference Application. Brilliance: Research of Artificial Intelligence, 3(2), 282-289.
- **Hernández, E. A., Sánchez, A. A., Flores, M. U. I., Chernyshov, P. E. L., Gutiérrez, J. J. R., Bermúdez, E. A. Z., Manzano, L. A. I., & Manzano, M. A. I.** (2024). Algoritmo para el procesamiento de sensores de presión utilizados en la detección de la postura humana en reposo. JÓVENES EN LA CIENCIA, 28, 1-8. https://doi.org/10.15174/jc.2024.4509

- **Kaur, J.** (2024, 29 agosto). Inception Architecture for Computer Vision and its Future. XenonStack. https://www.xenonstack.com/blog/inception-architecture-computer-vision
- **Krichen, M.** (2023). Convolutional Neural Networks: a survey. Computers, 12(8), 151. https://doi.org/10.3390/computers12080151
- **Kumar, A., Kaur, A., & Kumar, M.** (2018). Face Detection Techniques: a review. Artificial Intelligence Review, 52(2), 927-948. https://doi.org/10.1007/s10462-018-9650-2
- **Kurama, V.** (2025, 20 marzo). Deep learning architectures explained: ResNet, InceptionV3, SqueezeNet. DigitalOcean. https://www.digitalocean.com/community/tutorials/popular-deep-learning-architectures-resnet-inceptionv3-squeezenet
- **Li, X., Wang, W., Wu, L., Chen, S., Hu, X., Li, J., Tang, J., & Yang, J.** (2020). Generalized Focal Loss: Learning Qualified and Distributed Bounding Boxes for Dense Object Detection. Advances in Neural Information Processing Systems, 33, 1-12.
- **Li, Y., Xu, H., Bian, M., & Xiao, J.** (2020). Attention Based CNN-ConvLSTM for Pedestrian Attribute Recognition. Sensors, 20(3), 811. https://doi.org/10.3390/s20030811
- **Li, Y., Zhang, X., & Chen, D.** (2018). CSRNet: Dilated Convolutional Neural Networks for Understanding Highly Congested Scenes. En Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 1091-1100). https://doi.org/10.1109/cvpr.2018.00120
- **Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L.** (2014). Microsoft coco: Common objects in context. European conference on computer vision (pp. 740-755). Springer, Cham.
- **Liu, S., Qi, L., Qin, H., Shi, J., & Jia, J.** (2018). Path aggregation network for instance segmentation. Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 8759-8768).
- **Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C., & Berg, A. C.** (2016). SSD: Single Shot MultiBox Detector. En Lecture notes in computer science (pp. 21-37). https://doi.org/10.1007/978-3-319-46448-0_2
- **Masood, D.** (2024b, marzo 5). Pre-trained CNN architectures designs, performance analysis and comparison. Medium. https://medium.com/@daniyalmasoodai/pre-train-cnn-architectures-designs-performance-analysis-and-comparison-802228a5ce92
- **Mokayed, H., Quan, T. Z., Alkhaled, L., & Sivakumar, V.** (2022). Real-Time Human Detection and Counting System Using Deep Learning Computer Vision Techniques. Artificial Intelligence And Applications, 1(4), 221-229. https://doi.org/10.47852/bonviewaia2202391
- **Nguyen, T. A., & Nguyen, T. H.** (2025). Artificial Intelligence for Human Detection, Identification and Tracking: Methods and Applications. Journal of Future Artificial Intelligence and Technologies, 2(1), 1-16.
- **NTP 1102: Equipos de detección de presencia de personas (II).** (s. f.). Portal INSST. https://www.insst.es/documentacion/colecciones-tecnicas/ntp-notas-tecnicas-de-prevencion/32-serie-ntp-numeros-1101-a-1135-ano-2018/ntp-1.102-equipos-de-deteccion-de-presencia-de-personas-ii-posicionamiento-de-cortinas-fotoelectricas
- **Redmon, J., Divvala, S., Girshick, R., & Farhadi, A.** (2015, 8 junio). You only look once: Unified, Real-Time Object Detection. arXiv.org. https://arxiv.org/abs/1506.02640
- **Ren, S., He, K., Girshick, R., & Sun, J.** (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. Advances in Neural Information Processing Systems, 28.
- **Roboflow: Computer vision tools for developers and enterprises.** (s. f.). Roboflow. https://roboflow.com/
- **Rosebrock, A.** (2017, marzo 20). ImageNet: VGGNet, ResNet, Inception, and xception with Keras. PyImageSearch. https://pyimagesearch.com/2017/03/20/imagenet-vggnet-resnet-inception-xcept

- **Sandoval, C.** (2023, 4 abril). Hablemos de la detección de objetos. LISA Insurtech. https://lisainsurtech.com/hablemos-de-la-deteccion-de-objetos/
- **Simonyan, K., & Zisserman, A.** (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv preprint arXiv:1409.1556.
- **Solawetz, J.** (2023). What is YOLOv8? The Ultimate Guide. Roboflow Blog. https://blog.roboflow.com/whats-new-in-yolov8/
- **Stewart, R., Andriluka, M., & Ng, A. Y.** (2016). End-to-end people detection in crowded scenes. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2325-2333).
- **Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A.** (2016, 23 febrero). Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. arXiv.org. https://arxiv.org/abs/1602.07261
- **Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A.** (2014, 17 septiembre). Going Deeper with Convolutions. arXiv.org. https://arxiv.org/abs/1409.4842
- **Terven, J., Córdova-Esparza, D., & Romero-González, J.** (2023). A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS. Machine Learning And Knowledge Extraction, 5(4), 1680-1716. https://doi.org/10.3390/make5040083
- **Thakur, R.** (2024, 12 marzo). Beginner's Guide to VGG16 Implementation in Keras. Built In. https://builtin.com/machine-learning/vgg16
- **Ultralytics.** (s.f.). YOLOv8 by Ultralytics. Recuperado de https://ultralytics.com/yolo
- **VGG vs ResNet vs Inception vs MobileNet | Kaggle.** (s. f.). https://www.kaggle.com/discussions/getting-started/433540
- **Wang, C. Y., Liao, H. Y. M., Wu, Y. H., Chen, P. Y., Hsieh, J. W., & Yeh, I. H.** (2020). CSPNet: A new backbone that can enhance learning capability of CNN. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops (pp. 390-391).
- **Wojke, N., Bewley, A., & Paulus, D.** (2017). Simple online and realtime tracking with a deep association metric. 2017 IEEE international conference on image processing (ICIP) (pp. 3645-3649). IEEE.
- **Yang, C., Wang, X., & Mao, S.** (2023). TARF: Technology-Agnostic RF Sensing for Human Activity Recognition. IEEE Journal of Biomedical and Health Informatics, 27(2), 636-647.
- **YOLOv8.** (s.f.). GitHub - ultralytics/ultralytics. Recuperado de https://github.com/ultralytics/ultralytics
- **YOLOV8: State-of-the-Art Computer Vision Model.** (s. f.). https://yolov8.com/
- **Zafar, S.** (2023, 12 noviembre). Choosing the Right Pre-Trained Model: A Guide to VGGNet, ResNet, GoogleNet, AlexNet, and Inception. Medium. https://medium.com/@sohaib.zafar522/choosing-the-right-pre-trained-model-a-guide-to-vggnet-resnet-googlenet-alexnet-and-inception-db7a8c918510

# Apéndice 1: Carta de Consentimiento para Participación en el Estudio

**FACULTAD DE CONTADURÍA, ADMINISTRACIÓN E INFORMÁTICA**

**UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS**

**CARTA DE CONSENTIMIENTO INFORMADO**

Yo, **(nombre completo del participante)**, mayor de edad, por medio de la presente, otorgo mi consentimiento libre, voluntario e informado para la captura y uso de mi imagen y video en el marco del proyecto de investigación sobre **identificación de personas mediante modelos de VC**, llevado a cabo en la Facultad de Contaduría, Administración e Informática de la Universidad Autónoma del Estado de Morelos.

Declaro que:

1. Se me ha informado que las imágenes y videos capturados serán utilizados exclusivamente para fines de investigación y desarrollo del modelo de identificación de personas.

2. Comprendo que mi imagen y video podrán ser incluidas en publicaciones académicas, incluyendo la tesis del investigador responsable y otros documentos relacionados con la investigación.

3. Se me ha asegurado que los datos recopilados serán tratados con estricta confidencialidad y bajo las normativas de protección de datos personales.

4. Entiendo que mi imagen no será utilizada con fines comerciales sin una autorización adicional.

5. Tengo el derecho de revocar este consentimiento en cualquier momento, notificando por escrito a los responsables del proyecto, salvo en los casos en los que la información ya haya sido publicada.

Firm del Participante: _____

Nombre: _____

Fecha: _____

Cuernavaca, Morelos a 25 de Mayo del 2026.

**MTRO. ELMER IVAN SANCHEZ RABADAN**
**ENCARGADO DE DESPACHO DE LA DIRECCION DE LA FACULTAD**
**DE CONTADURIA, ADMINISTRACION E INFORMATICA DE LA UAEM.**
**PRESENTE.**

En mi carácter de revisor de tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Cómputo Aplicado, de la estudiante Lizeth Meza Alegria, con matrícula 10072384, con el título **CONTEO DE ESTUDIANTES EN AMBIENTES EDUCATIVOS USANDO CNNs PRE ENTRENADAS**, por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que la Universidad Autónoma del Estado de Morelos tenga establecidos para obtener el grado mencionado.

Atentamente
***Por una humanidad culta***

Dr. José Alberto Hernández Aguilar
**Profesor- investigador**
Facultad de Contaduría, Administración e Informática

UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS

Se expide el presente documento con firma electrónica UAEM, soportada por el certificado vigente a la fecha de su elaboración y con efectos plenos de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS PUBLICADOS en el ÓRGANÓ INFORMATIVO UNIVERSITARIO "ADOLFO MENÉNDEZ SAMARÁ" número 117 de fecha 20 de abril de 2021.

Sello electrónico

**JOSE ALBERTO HERNANDEZ AGUILAR | Fecha:2026-05-27 16:16:14 | FIRMANTE**

SFBFmzIBa0lLWmPm+QAupuffFDwgJ5/Sh5aDBpoOqee0jCZsCJB/9/0FF/dQioX6oektGCPr4B2vtROKZeVFJbHjka1JYjt4b5n6PS1TrjuRSRZUija25nLQgc3PVoPaNugb1RLw52C73bSBOIRFAKzqMdFCugPYDG/5Fxla7cMKAoB4jq8+bD+OeBNAN5kqj9AyHRbVJdMCy9W6OdIOiKJ9Oywg93JwMlWqo8SUPHdrYvaYgvjI1hO3KEZAaCVXESEgeIH3qw8XV33rVLloRqkyVz8WmoS0FKDYJVWNpZ8yLgObcvysI+dW/0kMAJkOgN+WMmIiDX+1S6OYvgufxA==

**Puede verificar la autenticidad del documento en la siguiente dirección electrónica o escaneando el código QR ingresando la siguiente clave:**

**5qenJ0wrd**

https://efirma.uaem.mx/noRepudio/WNhShDUoKvNPwAIosaKom238jC7xwWxo

Cuernavaca, Morelos a 15 de Mayo del 2026.

**MTRO. ELMER IVAN SANCHEZ RABADAN**
**ENCARGADO DE DESPACHO DE LA DIRECCION DE LA FACULTAD**
**DE CONTADURIA, ADMINISTRACION E INFORMATICA DE LA UAEM.**
**PRESENTE.**

En mi carácter de revisor de tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Cómputo Aplicado, de la estudiante Lizeth Meza Alegria, con matrícula 10072384, con el título **CONTEO DE ESTUDIANTES EN AMBIENTES EDUCATIVOS USANDO CNNs PRE ENTRENADAS**, por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que la Universidad Autónoma del Estado de Morelos tenga establecidos para obtener el grado mencionado.

Atentamente
***Por una humanidad culta***

Dr. Outmane Oubram
**Profesor- investigador**
Facultad de Ciencias Químicas e Ingeniería.

UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS

Se expide el presente documento con firma electrónica UAEM, soportada por el certificado vigente a la fecha de su elaboración y con efectos plenos de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS PUBLICADOS en el ÓRGANÓ INFORMATIVO UNIVERSITARIO "ADOLFO MENÉNDEZ SAMARÁ" número 117 de fecha 20 de abril de 2021.

Sello electrónico

**OUTMANE OUBRAM | Fecha:2026-05-24 12:34:14 | FIRMANTE**

icA5m4S169IRDFMArNRSLh9J2vtjmqVcEdmanoL8vlyBPopzwk5MR1gSjtTBK2jw6CfpcX5ej0cBXC7EN1QvJTwIQ5FMgdsy26F80g34cksPt7gUsQ46b0u48fZsYNd2UsRygmbNNDRYBoNtUt84oQXSHdMLO7ymMHL5VtjqqowD96gI8eysNduZ4xAg2cvKskcrxpWwyuIYq/wP+Iz8DwJB0baTSiO8220Lu/keRUlzclJUhu3YyzyOchAUcJKyvcXQ0KU39UOUjJsMvlScQhXqW32+uGUjYlrAF3pLCjteBmE2c19Bb6Pp0kSu+E2ylGjlynDM53t4/TXo+id9/w==

**Puede verificar la autenticidad del documento en la siguiente dirección electrónica o escaneando el código QR ingresando la siguiente clave:**

**ELmG5rYkZ**

https://efirma.uaem.mx/noRepudio/YUoruS5KMGaGPYOTCMP7r8Dx7mh3qz2E

Cuernavaca, Morelos a 15 de Mayo del 2026.

**MTRO. ELMER IVAN SANCHEZ RABADAN**
**ENCARGADO DE DESPACHO DE LA DIRECCION DE LA FACULTAD**
**DE CONTADURIA, ADMINISTRACION E INFORMATICA DE LA UAEM.**
**PRESENTE.**

En mi carácter de revisor de tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Cómputo Aplicado, de la estudiante Lizeth Meza Alegria, con matrícula 10072384, con el título **CONTEO DE ESTUDIANTES EN AMBIENTES EDUCATIVOS USANDO CNNs PRE ENTRENADAS**, por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que la Universidad Autónoma del Estado de Morelos tenga establecidos para obtener el grado mencionado.

Atentamente
***Por una humanidad culta***

Dr. Pedro Moreno Bernal
**Profesor- investigador**
Facultad de Contaduría, Administración e Informática

UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS

Se expide el presente documento con firma electrónica UAEM, soportada por el certificado vigente a la fecha de su elaboración y con efectos plenos de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS PUBLICADOS en el ÓRGANÓ INFORMATIVO UNIVERSITARIO "ADOLFO MENÉNDEZ SAMARÁ" número 117 de fecha 20 de abril de 2021.

Sello electrónico

**PEDRO MORENO BERNAL | Fecha:2026-05-21 15:17:38 | FIRMANTE**

qE0Z2TJcLxEkD5sJOOrrHpxSAhlix3Nry8ARsW8PNyyQ4PMhdcShQ7Emjg07y2rmFUC/iCtb7k3UXqL2rv+yCvr0f3lmaNHIXT6kD+3s1mXPnNSxRq1ULboGSy6XlGROOYaPKk0+Jv2Gu44BuugRfR/DjbFslRdfDdvpRJWDU+iP87GEZ561w6dpH/a0JPGYvHoMXtl3fJ+MsvOU7H6kS1HAkPjjl6MhMfa6i4Mq16syN/nBqOeEBr1hrABjjdZhoQrnvYBZDO5S/HkxEdrU31tG87JCFJZQzJpiUnAe8FhzCUFDbHMhR+i5NsMd6zWP0HBh9uLCDyZTOp/0x3U5oQ==

**Puede verificar la autenticidad del documento en la siguiente dirección electrónica o escaneando el código QR ingresando la siguiente clave:**

**VYyqWS4Am**

https://efirma.uaem.mx/noRepudio/VRXjO8Bwzd71TdRjD81Z8rCDaPUP2A20

Cuernavaca, Morelos a 15 de Mayo del 2026.

**MTRO. ELMER IVAN SANCHEZ RABADAN**
**ENCARGADO DE DESPACHO DE LA DIRECCION DE LA FACULTAD**
**DE CONTADURIA, ADMINISTRACION E INFORMATICA DE LA UAEM.**
**PRESENTE.**

En mi carácter de revisor de tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Cómputo Aplicado, de la estudiante Lizeth Meza Alegria, con matrícula 10072384, con el título **CONTEO DE ESTUDIANTES EN AMBIENTES EDUCATIVOS USANDO CNNs PRE ENTRENADAS**, por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que la Universidad Autónoma del Estado de Morelos tenga establecidos para obtener el grado mencionado.

Atentamente
***Por una humanidad culta***

Dr. Víctor Hugo Pacheco Valencia
**Profesor- investigador**
**Centro de Investigación en Ingeniería y Ciencias aplicadas**

UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS

Se expide el presente documento con firma electrónica UAEM, soportada por el certificado vigente a la fecha de su elaboración y con efectos plenos de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS PUBLICADOS en el ÓRGANÓ INFORMATIVO UNIVERSITARIO "ADOLFO MENÉNDEZ SAMARÁ" número 117 de fecha 20 de abril de 2021.

Sello electrónico

**VICTOR HUGO PACHECO VALENCIA | Fecha:2026-05-16 10:31:42 | FIRMANTE**

kTvULIP1btxuySos3+DHZtKz83L9UjtxMy1Cd2MITpkprnOdki9dbNZqhwXlLTPxIQoFPyl2z9uvz/KlrZIMA06wLvGkNQLvvr6wSKLZLY9nye2qLiXOpRpKkKGTcgq8soMtqP7iSfNwgVN5AO1FW2qSZdT9iIuN3wDRRw+Ykjg/bU/sEX6lE2BGjxwGbA4W99dJUJbESAogLvztrEceVGHTFPtAY8TXBNgaQSyvLRs5HhqldQO15OVGl1kDtBaJQkvkfzEg0YdsKXMcrShyvmJ0ABA0vhoDxBS9rweEwYtDEv7WGWb5nabH5Uxk/pa44hoDbu1DoBjddScKbF8Icg==

**Puede verificar la autenticidad del documento en la siguiente dirección electrónica o escaneando el código QR ingresando la siguiente clave:**

**uWlatbDJ0**

https://efirma.uaem.mx/noRepudio/qibW5If7pZpggeojGHjGHcBINiOZuTKG

Cuernavaca, Morelos a 25 de Mayo del 2026.

**MTRO. ELMER IVAN SANCHEZ RABADAN**
**ENCARGADO DE DESPACHO DE LA DIRECCION DE LA FACULTAD**
**DE CONTADURIA, ADMINISTRACION E INFORMATICA DE LA UAEM.**
**PRESENTE.**

En mi carácter de revisor de tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Cómputo Aplicado, de la estudiante Lizeth Meza Alegria, con matrícula 10072384, con el título **CONTEO DE ESTUDIANTES EN AMBIENTES EDUCATIVOS USANDO CNNs PRE ENTRENADAS**, por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que la Universidad Autónoma del Estado de Morelos tenga establecidos para obtener el grado mencionado.

Atentamente
***Por una humanidad culta***

Dra. María Yasmín Hernández Pérez
**Profesor- investigador**
TecNM Centro Nacional de Investigación y Desarrollo Tecnológico

UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS

Se expide el presente documento con firma electrónica UAEM, soportada por el certificado vigente a la fecha de su elaboración y con efectos plenos de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS PUBLICADOS en el ÓRGANÓ INFORMATIVO UNIVERSITARIO "ADOLFO MENÉNDEZ SAMARÁ" número 117 de fecha 20 de abril de 2021.

Sello electrónico

**MARÍA YASMÍN HERNÁNDEZ PÉREZ | Fecha:2026-06-02 13:36:55 | FIRMANTE**

IGVEG+d/vJKGsNTUKtEdW8fnpzhgVi9UriaS/sY7KyWPhg1WM2Tk1j9qazSBLFFejkQk9DIB95REQ175Ar31KnJoxBU6h0+KvgM97ZuKOuNgcjOjawheo2/hkmjh7l55iocwKZtMjdBL46eHn8pi04RYodPFEefG36zkZep7Vz/fBBiss/BVnQIqQSnH1GDkvRhmHhYMUjIY0ptP1jrFxMe61RPuA/rORFO6pzrLqz17ujrPazmrb5X0XsyzDQGNFb+tyHWAXEFlACDwFE9ZZjmOHp09MESi9dN/V0OnEOT9alrI8ljS4UTO6Vo06UlciQMQYPPdw5J8BOONueuxKg==

**Puede verificar la autenticidad del documento en la siguiente dirección electrónica o escaneando el código QR ingresando la siguiente clave:**

**SqAk7gmvW**

https://efirma.uaem.mx/noRepudio/zRxM7IZVm1hJAvayM5lKeIuVkjdCYF7B