



Facultad de Ingeniería  
Departamento de Ingeniería Eléctrica  
**Universidad de Concepción**



# **DETECCIÓN DE ESTACIONAMIENTOS NO AUTORIZADOS Y RESERVADOS, MEDIANTE USO DE TECNICAS DE DEEP LEARNING**

POR

*Sergio Nicolás Opazo Flores*

Memoria de Título presentada a la Facultad de Ingeniería de la Universidad de Concepción para optar al título profesional de Ingeniero Civil en Telecomunicaciones.

**Profesor Guía**  
Francisco Pérez Venegas

Concepción, 11 de agosto de 2025

© 2025 Sergio Nicolás Opazo Flores

© 2025 Sergio Nicolás Opazo Flores

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento.

Perseverar es la clave del éxito,  
nunca te rindas.

## **Agradecimientos**

Agradecer en primera instancia a mi familia, por el apoyo constante en mis proyectos, durante la época universitaria, a Monserrat por el apoyo en el desarrollo de esta memoria de título, por acompañarme a tomar imágenes en diversos horarios, condiciones climáticas y motivarme a no fallar y mejorar. A mis amigos tanto del deporte, que se hicieron parte de mi familia y los amigos de siempre, por distender el ambiente y acompañar en momentos importantes, a las horas de estudio y todo lo que conlleva el proceso universitario.

A los auxiliares, los “tíos” que, en diversos momentos, deportivos y académicos, me acompañaron con una palabra de aliento, momentos de conversación y tomar cafecito, momentos que no serán olvidados.

Agradecer al profesor Francisco Pérez Venegas, por las ganas que puso en el proyecto, la confianza y el acompañamiento constante en la etapa final de la carrera, fue un guía fundamental en la finalización de esta etapa tan importante, sin duda su ejemplo motiva a seguir en el camino de la ciencia y tecnología.

Finalmente, el agradecimiento más importante es a mí, por no decaer, perseverar y trabajar por el cumplimiento de objetivos, tanto en lo académico y deportivo, generando lazos que seguirán para toda la vida, muchos, que en el proceso entendieron que llegábamos con una mochila llena de ropa para cambiar los cuadernos por una pelota, entendiendo que no solo vivimos de los estudios y día a día forjamos el desarrollo profesional con el deportivo, entendiendo que la constancia, trabajo y compromiso son valores que trascienden en la vida.

## **Resumen**

Este proyecto aborda el desarrollo de modelos, utilizando visión por computador, basado en aprendizaje profundo para la detección automática de patentes chilenas y de detección de espacios de estacionamientos. El constante aumento del parque automotriz en nuestro país ha generado desafíos en la gestión del tránsito y el control de zonas de estacionamiento, lo que motivó la creación de una solución capaz de identificar vehículos y su disposición en espacios públicos o privados.

Se implementó un modelo de detección usando la arquitectura de YOLOv8, entrenado y optimizado mediante el uso de técnicas de aumento de datos, preprocesamiento y validación de patentes chilenas. En el caso de las patentes, se integró un sistema de reconocimiento óptico de caracteres, el cual nos ayuda a reconocer formatos válidos para nuestro país, el que consiste en 2 letras y 4 números o 4 letras y 2 números, ignorando caracteres irrelevantes como lo son los reparadores, en el caso de puntos, estrellas y diversos textos institucionales. Para el modelo de espacios de estacionamiento, se entrenó un modelo separado capaz de detectar los espacios libres y ocupados, en ambos casos se consideraron variables como iluminación, clima, altura, ángulos y distancia como factores principales.

El sistema fue evaluado utilizando diversas métricas como Precisión, Recall, F1-score, curvas Precisión vs Recall y curva ROC y también, mediante la evaluación humana, observando el desempeño óptimo en la detección de patentes en condiciones normales alcanzando un área bajo la curva ROC (AUC) de 0.69, lo que indica una buena capacidad del modelo para distinguir, mientras que el modelo de estacionamientos mostró un máximo más bajo (0.53), lo que evidencia un comportamiento más errático, pero con mucho margen de mejora.

## **Abstract**

This project addresses the development of models using computer vision based on deep learning for the automatic detection of Chilean license plates and parking space occupancy. The continuous increase in the number of vehicles in our country has generated challenges in traffic management and parking zone control, which motivated the creation of a solution capable of identifying vehicles and their placement in public or private spaces.

A detection model was implemented using the YOLOv8 architecture, trained and optimized through data augmentation techniques, preprocessing, and validation with Chilean license plates. For license plate recognition, an optical character recognition (OCR) system was integrated to identify valid formats in Chile, consisting of 2 letters and 4 numbers or 4 letters and 2 numbers, while ignoring irrelevant characters such as dots, stars, and institutional texts. For the parking space model, a separate model was trained to detect free and occupied spaces. In both cases, variables such as lighting, weather, height, angles, and distance were considered key factors.

The system was evaluated using various metrics such as Precision, Recall, F1-score, Precision vs Recall curves, and ROC curves. Additionally, a qualitative evaluation was performed through human inspection. The license plate detection model showed optimal performance under normal conditions, achieving an Area Under the ROC Curve (AUC) of 0.69, indicating a good ability to distinguish between classes. In contrast, the parking space detection model reached a lower AUC of 0.53, reflecting more erratic behavior but with significant room for improvement.

# Índice

1. Introducción .....	10
1.1. Trabajos previos .....	12
1.2. Discusión .....	13
1.3. Contexto .....	13
1.4. Definición del problema .....	15
1.5. Objetivos .....	16
1.5.1. General .....	16
1.5.2. Específicos .....	16
1.6. Metodologías .....	17
1.7. Alcances y limitaciones .....	20
1.7.1. Alcances .....	20
1.7.2. Limitaciones .....	21
2. Marco teórico .....	23
2.1. Introducción .....	23
2.2. Conceptos Fundamentales .....	23
2.3. Estado del Arte .....	29
2.4. Justificación del Enfoque .....	30
3. Desarrollo .....	31
3.1. YOLOv8 [32] [33] .....	31
3.2. Modelo de detección de patentes y lectura de caracteres .....	32
3.3. Modelo de detección de espacios de estacionamiento, Libre u Ocupado .....	33
4. Resultados .....	38
4.1. Resultados de detección de Placas Patentes .....	38
4.1.1. Resultados de simulación .....	38
4.1.2. Evaluación humana .....	41
4.2. Resultados de detección de espacios disponibles .....	43
4.2.1. Resultados de simulación .....	43
4.2.2. Evaluación Humana .....	46
4.3. Estimación económica .....	48
4.3.1. Costos de desarrollo de modelos .....	48
4.3.2. Costos de implementación de modelos .....	50
4.3.3. Total general Proyecto .....	51

4.3.4. Observaciones generales.....	52
4.4. Discusión .....	53
5. Conclusiones .....	54
5.1. Sumario .....	54
5.2. Conclusiones .....	54
5.3. Trabajo a futuro.....	56
6. Bibliografía .....	57
Anexos.....	60
Anexo A.....	60
Anexo B.....	60
Anexo C.....	61
Anexo D.....	65

## **Índice de tablas**

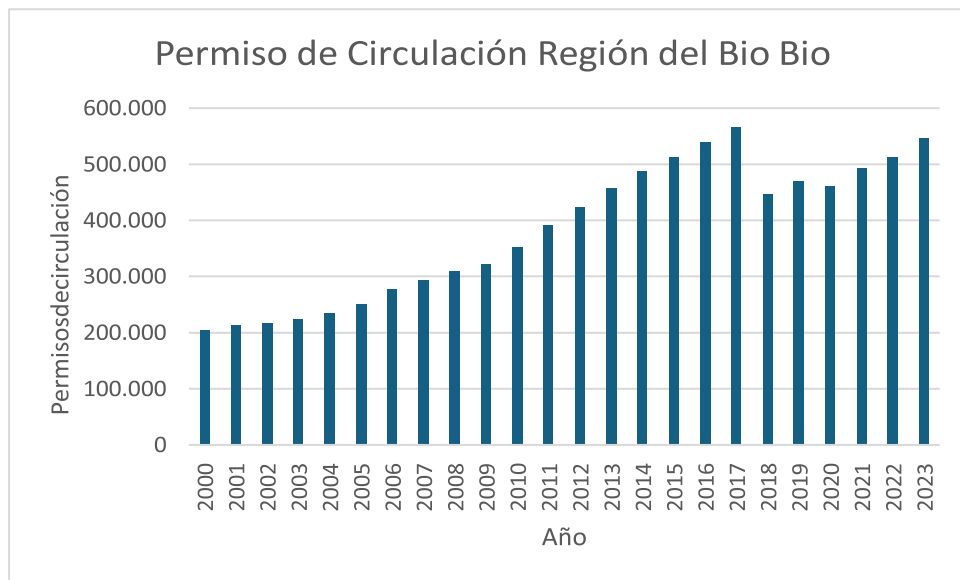
<b>Tabla 1:</b> Espacios disponibles en estacionamiento FI. ....	14
<b>Tabla 2:</b> Creación dataset inicial, modelo detección de placas patentes. ....	17
<b>Tabla 3:</b> Creación de dataset de imágenes, modelo de detección de espacios disponibles. ....	18
<b>Tabla 4:</b> Separación de dataset para modelo de detección de espacios disponibles en FI. ....	18
<b>Tabla 5:</b> Separación de dataset para modelo de detección de placas patentes y lectura de caracteres. ....	19
<b>Tabla 6:</b> Criterios toma de imágenes, estacionamiento FI. ....	37
<b>Tabla 7:</b> Costos desarrollo de modelos. ....	49
<b>Tabla 8:</b> Costo desarrollo de modelos para caso a y b.....	50
<b>Tabla 9:</b> Costo de materiales para implementación de modelos. ....	50
<b>Tabla 10:</b> Costo total de materiales para implementación de modelos. ....	51
<b>Tabla 11:</b> Costo desarrollo de modelo e implementación, para caso a. ....	52
<b>Tabla 12:</b> Costo desarrollo de modelo e implementación, para caso b. ....	52
<b>Tabla 13:</b> Permisos de circulación año 2023, datos de INE. ....	60

## Índice de figuras

<b>Figura 1:</b> Permisos de Circulación, Datos recolectados por INE. ....	10
<b>Figura 2:</b> Precisión vs Recall, imagen obtenida de [28] .....	25
<b>Figura 3:</b> F1- Score vs Confianza, imagen obtenida de [29] .....	26
<b>Figura 4:</b> Precisión vs Confianza, imagen obtenida de [29]. ....	26
<b>Figura 5:</b> Recall vs Confianza, imagen obtenida de [29]. ....	27
<b>Figura 6:</b> Curva ROC, imagen obtenida de [30]. ....	28
<b>Figura 7:</b> Funcionamiento de YOLOv8. ....	31
<b>Figura 8:</b> Patentes chilenas, imagen obtenida de Radio Bio Bio [27]. ....	32
<b>Figura 9:</b> Ejemplo imagen clase PATENTE SIN LECTURA, imagen propia. ....	33
<b>Figura 10:</b> Ejemplo clase PATENTE, imagen propia. ....	33
<b>Figura 11:</b> Imagen desde Edificio de Ingenieria, imagen propia. ....	35
<b>Figura 12:</b> Imagen desde Edificio TM, imagen propia. ....	35
<b>Figura 13:</b> Imagen desde altura mediante dron, imagen propia. ....	35
<b>Figura 14:</b> Imagen desde Edificio Sistemas, imagen propia. ....	36
<b>Figura 15:</b> Imagen desde Edificio ICQ, imagen propia. ....	36
<b>Figura 16:</b> F1 vs Confianza. ....	38
<b>Figura 17:</b> Precisión vs Confianza. ....	39
<b>Figura 18:</b> Recall vs Confianza. ....	39
<b>Figura 19:</b> Precisión vs Recall. ....	40
<b>Figura 20:</b> Curva ROC. ....	40
<b>Figura 21:</b> Detección inicial fallida. ....	41
<b>Figura 22:</b> Detecciones de placas patentes. ....	42
<b>Figura 23:</b> Detecciones de placas patentes. ....	42
<b>Figura 24:</b> Detecciones patentes no elegibles. ....	43
<b>Figura 25:</b> F1- Score vs Confianza. ....	43
<b>Figura 26:</b> Precisión vs Confianza. ....	44
<b>Figura 27:</b> Curva ROC. ....	44
<b>Figura 28:</b> Recall vs Confianza. ....	45
<b>Figura 29:</b> Precisión vs Recall. ....	45
<b>Figura 30:</b> Detección espacios de estacionamiento. ....	46
<b>Figura 31:</b> Detección de espacios de estacionamiento. ....	47
<b>Figura 32:</b> Detección de espacios de estacionamiento. ....	48
<b>Figura 33:</b> Detección de espacios de estacionamiento. ....	60
<b>Figura 34:</b> Detección de estacionamiento. ....	61

# 1. Introducción

Dado el aumento que ha tenido el parque automotriz en nuestro país, en el último reporte del Instituto Nacional de Estadísticas (INE), se ha aumentado en un 4.6% el número de permisos de circulación otorgados a nivel nacional, actualmente circulan 6538727 vehículos motorizados, en donde se observa que el mayor flujo de automóviles se centra en la zona Metropolitana, región de Valparaíso y Bio Bio [1]. Viendo esto, nuestra región muestra un aumento significativo en el tránsito vehicular [2], lo que genera congestión diaria, que afecta a conductores y peatones. Estos datos, fueron verificados, observando los permisos de circulación otorgados anualmente por cada municipalidad, en ella podemos ver un aumento considerable desde los años 2000 hasta el año 2017, con disminución en el año 2018, volviendo a aumentar hasta día de hoy.



**Figura 1:** Permisos de Circulación, Datos recolectados por INE.

Por otra parte, en nuestra región se ven constantes arreglos viales [3], que si bien son necesarias para mejorar calles e infraestructura, aumentan los tiempos de traslado en los distintos trayectos y también, en la búsqueda de espacios disponibles en los diversos puntos de estacionamientos, debido a su alta ocupación, generando un impacto negativo, en la calidad del aire, dado el aumento de gases contaminantes [4], sino que también los conductores se ven afectados en su salud mental [5], a causa del estrés producido por horas de conducción en el trayecto a su hogar o lugares de trabajo, o en la dificultad de encontrar espacios de estacionamiento en los lugares a los cuales se movilizan.

Por esto, es que se ha observado en investigaciones relacionadas, se han utilizado redes neuronales convolucionales, OpenCV y algoritmos de reconocimiento óptico de caracteres, para poder determinar espacios disponibles, en diversos estacionamientos, por lo que la utilización de esta tecnología, ha sido aplicada de manera favorable en diversos contextos a nivel mundial, por lo que podemos entender que el proyecto es escalable, optimizable y genera beneficios sociales y ambientales, dado su uso a futuro en tiempo real.

La importancia de este proyecto va en busca de resolver una problemática, dentro de nuestra universidad, lo cual es la falta de control en el uso de espacios en el estacionamiento institucional, lo que se puede observar mediante la identificación de espacios libres u ocupados de estacionamientos, además de detectar diversas placas patentes, dado esto se puede optimizar el tiempo de búsqueda de espacios disponibles y además de ello, se aportara en la disminución de emisión de gases contaminantes y reducción del estrés de los conductores, por las problemáticas que se generan en tiempo y búsqueda de espacios disponibles, además de la ineficiencia y mala ocupación de diversos espacios disponibles, esto genera dificultades en el acceso y salida del estacionamiento debido a que solo cuenta con una entrada y salida desde el estacionamiento de la FI, por lo que el objetivo es desarrollar un sistema utilizando DL, el cual nos ayudará a identificar diversos patrones, tanto en el uso de estacionamientos disponibles, como además en la detección de placas patentes de las personas que utilizan el espacio dentro de nuestra facultad, para ello se entrenará un modelo de detección de espacios disponibles y otro para detección de placas patentes y sus respectivos caracteres, con imágenes tomadas en diversas condiciones tanto dentro de la FI, para la visualización del espacio de estacionamiento, como imágenes en nuestro país, para la detección de placas patentes y sus respectivos caracteres, con la finalidad de tener un dataset propio de ambos modelos y sus características.

El desarrollo del proyecto de memoria de título, consta en diversas etapas, considerando revisión bibliográfica, recopilación de datos, preprocesamiento de imágenes capturadas para el entrenamiento de ambos modelos, validación y evaluación de los resultados obtenidos para los modelos de detección de espacios de estacionamiento y de placas patentes y lectura de sus caracteres, visualizando diversas métricas a considerar, además de su visualización de la lectura de caracteres y de espacios disponibles.

Todo esto, nace bajo la necesidad de mejora en el sistema de monitoreo del estacionamiento, siendo escalable a más ambientes, por ejemplo, condominios o centros privados de estacionamiento u otros lugares dentro de la UDEC, entre otros ambientes, por lo que se espera que los resultados de este proyecto no solo beneficien a la mayor cantidad de personas dentro de la FI, sino que también dentro del país.

## **1.1. Trabajos previos**

Tanto para el modelo de desarrollo de detección de espacios disponibles dentro de estacionamientos, como para la detección y reconocimiento de patentes, se han utilizado diversas técnicas de DL, además de la utilización de librerías como OpenCV, Tesseract, OpenALPR, en ellas se ha observado un alto nivel de precisión, lo que nos da a entender que estas herramientas son útiles para mejorar el control y acceso de vehículos, además de la optimización de espacios de estacionamientos, que ayudará a disminuir el tiempo de búsqueda de espacios, como la detección y utilización de ellos. Por otra parte, destaca el uso de técnicas de redes neuronales convolucionales, dada su aplicación con éxito en el reconocimiento en tiempo real gracias al uso de YOLOv8, para la automatización de procesos de control automotriz.

Para la detección y reconocimiento automático de placas patentes, se han detectado resultados óptimos, mejorables y escalables a futuro, mediante el uso de redes neuronales, uso de ML o DL, obteniendo mejoras en la precisión y velocidad en el reconocimiento de patrones en tiempo real. [6]

Por ejemplo, en la utilización de OpenCV y Tesseract [7], para el reconocimiento óptico de caracteres se ha podido automatizar de mejor forma la identificación de las placas patentes, las cuales se caracteriza por la utilización de mecanismos OCR [8], el cual funciona, detectando la imagen, la cual se procesa utilizando OCR, detectando los caracteres a leer y devolviendo el texto para su respectiva detección, lo que nos demuestra ser útil en un entorno real.

Por otra parte, el uso de CNN, en tareas como es el reconocimiento de imágenes, ha permitido identificar diversas patentes y caracteres, además de diferenciar entre espacios libres u ocupados de estacionamientos.

## **1.2. Discusión**

Hoy los sistemas buscan detectar y reconocer diversas placas patentes que existen en nuestro país, lo que además permite observar la disponibilidad de espacios disponibles en los diversos lugares aptos para estacionar, por lo que podemos ver que los métodos basados en técnicas de DL han mostrado ventaja, debido a que tienen alta precisión y capacidad para ser utilizados en diferentes entornos, como lo son las diferentes condiciones de iluminación, condiciones climáticas, ángulos de imágenes, diferencias de alturas o imágenes que pueden sufrir obstrucciones, que dificultan la selección de la imagen, además de otros factores que influyen en el entrenamiento, por otra parte, ofrece reconocimiento en tiempo real, lo que hace más atractivo para evaluar visualmente en cada instante.

Además, debemos entender, que existen desventajas, como lo es el costo computacional para su implementación, generar un volumen apropiado de imágenes para el entrenamiento, validación y pruebas, todo ello para validar su correcto funcionamiento, es por esto que es necesario seguir avanzando en el proceso de optimización e investigación de estos sistemas tanto para la detección de espacios disponibles u ocupados y para la detección de placas patentes y el reconocimiento de sus respectivos caracteres, esto además busca una reducción de costos y mejorar el rendimiento en condiciones adversas.

Por lo que, en este trabajo, se busca obtener un rendimiento óptimo, en la precisión y rendimiento, utilizando redes neuronales convolucionales y técnicas de preprocesamiento, lo que permitirá la optimización y eficiencia, con la utilización para la FI y escalabilidad para la Universidad de Concepción, lo que proporciona un menor desgaste para los usuarios, como también para sus trabajadores, dado a que no necesitaran mayor fiscalización de forma presencial y continua.

## **1.3. Contexto**

El aumento del parque automotriz en nuestro país no es ajeno a nuestra realidad regional, esta realidad, también se observa a nivel universitario y nuestra FI, dado que se observa saturación de espacios disponibles dentro del estacionamiento, debido a la

cantidad de espacios disponibles y ocupación diaria, lo transforma en una problemática en horario punta.

Actualmente, el estacionamiento de la FI consta con 57 espacios disponibles, de los cuales 46 son espacios de libre disposición, tanto para funcionarios como académicos, 9 reservados para distintos directores de departamentos, además de los espacios reservados para el Decano y Vicedecano, y otros reservados para personas en situación de discapacidad. Este espacio cuenta con una entrada y una salida, lo que genera que el flujo vehicular genere cuellos de botella en horas punta, esto genera incertidumbre para los usuarios que necesitan acceder al estacionamiento, debido a que no existe información que verifique la disponibilidad de espacios, a excepción si ingresas al lugar. Es por ello que, si no existiera un espacio disponible, deberán estacionar en algún espacio fuera de la FI, lo que genera atrasos de los trabajadores, molestias y pérdida de tiempo, ya que deben regresar caminando hacia las dependencias universitarias.

Esto genera ineficiencia en el espacio disponible para estacionar, debido a la existencia de vehículos mal estacionados, además del uso de espacios reservados, tanto para autoridades como para personas en situación de discapacidad y en otros casos, automóviles estacionados en áreas que no están habilitadas para dicha ocupación, lo que afecta a usuarios que deben salir o entrar en un espacio habilitado, esta situación genera estrés constante, especialmente entre funcionarios y académicos, los cuales deben cumplir horarios, los que se pueden ver retrasados por dichas fallas nombradas anteriormente, además se genera un deterioro en la convivencia como conductores y tensiones por el uso no regulado de un espacio común.

A esto, se suma la sostenibilidad ambiental, la que se ve afectada debido a que aumenta el recorrido de los automóviles, en búsqueda de encontrar lugares disponibles, lo que genera aumento de gases contaminantes y va en contra del cuidado medioambiental, es por ello que se ve la necesidad de aportar en el desarrollo, mediante tecnología para la gestión de estos espacios, por lo que el uso de DL se observa como solución efectiva para estos casos.

**Tabla 1:** Espacios disponibles en estacionamiento FI.

<b>Espacio</b>	<b>Cantidad de espacios disponibles</b>
Libre disposición	46

Reservado para autoridades y personas en situación de discapacidad	11
Total de estacionamiento FI	57

#### **1.4. Definición del problema**

En la actualidad, el espacio disponible de estacionamiento no da abasto, lo que genera saturación del espacio, lo que produce que los conductores que no encuentran lugar deban estacionar fuera de la FI, o hacerlo de forma indebida, usando zonas no habilitadas o permitidas, esto genera incomodidad, molestia y estrés entre los usuarios, que deben cumplir horarios, además de que se muestra la falta de control sobre estos espacios.

Por otra parte, se observa que no existe un sistema que permita visualizar disponibilidad en tiempo real y su respectiva fiscalización, debido a la no existencia de cámaras de vigilancia dentro del espacio de estacionamiento de la FI, lo que obliga a recorrer el estacionamiento sin saber si se encontrara algún espacio disponible. Esto complica a los usuarios, ya que pierden tiempo en su búsqueda, aumenta la congestión vehicular dentro de la FI y a su vez, la emisión de gases contaminantes, debido a la búsqueda de espacios, día a día. Esto se puede ver empeorado, debido a la mala utilización de espacios reservados o de espacios no disponibles, lo que puede generar diversas molestias.

Es por esto que se propone un sistema basado en técnicas de DL, lo que permitirá la detección automática tanto de espacios disponibles u ocupados, como para la detección de placas patentes y sus respectivos caracteres, buscando optimizar el uso de espacios, además de mejorar la gestión de espacios disponibles, lo que además disminuirá el tiempo de búsqueda y contribuyendo a la salud de los usuarios y medioambiente.

Por ende, este modelo, generará la reducción de tránsito de vehículos, lo que aportará en la disminución de gases contaminantes asociados a la búsqueda de espacios disponibles, haciéndolo seguro y sostenible en el tiempo, considerando, además, normas éticas, de respeto de información personal, en la utilización de las placas patentes promoviendo soluciones que aporten a entornos más eficientes.

## **1.5. Objetivos**

### **1.5.1. General**

Se desarrollará un sistema basado en la utilización de técnicas de DL, para la detección de espacios disponibles dentro de la FI de la Universidad de Concepción. En ella, se utilizarán modelos de visión artificial, como lo es YOLOv8, que permite identificar en tiempo real la utilización de espacios libres u ocupados del estacionamiento, además de reconocer las distintas placas patentes y sus respectivos caracteres, para ello, se entrenará el modelo tomando imágenes en diversas condiciones tales como ángulos, ubicaciones, alturas, iluminación (mañana, tarde y noche) e incluso, condiciones difíciles, como son los días de lluvia, esto, para generar diversos patrones para el entrenamiento. Por otra parte, se generará un dataset propio, creando un trabajo colaborativo entre amigos, familiares y personas cercanas al desarrollo de este proyecto, para la toma de imágenes de placas patentes, lo cual buscará entrenar en diversas condiciones climáticas, iluminación, brillo, alturas, ángulos, el modelo de detección de placas patentes y sus respectivos caracteres.

### **1.5.2. Específicos**

**Objetivo Específico 1:** Crear dataset de imágenes, necesarias para entrenar un modelo de detección de placas patentes y lectura de caracteres de placas patentes.

**Objetivo Específico 2:** Crear dataset de imágenes, necesarias para entrenar un modelo de detección de espacios libres y ocupados dentro de la Facultad de Ingeniería de la Universidad de Concepción.

**Objetivo Específico 3:** Diseñar algoritmo de DL, para la detección de espacios disponibles y ocupados, dentro de la FI de la UDEC.

**Objetivo Específico 4:** Diseñar algoritmos de DL, para el reconocimiento de placas patentes y lecturas de caracteres.

**Objetivo Específico 5:** Evaluar algoritmos para la detección de espacios de estacionamientos libres y ocupados dentro de la FI de la UDEC.

**Objetivo Específico 6:** Evaluar algoritmos para la detección de placas patentes y sus respectivas lecturas de caracteres.

## 1.6. Metodologías

**Metodología Objetivo Especifico 1:** Realizar una recolección de imágenes de vehículos, considerando diferentes condiciones de iluminación, ángulos, distancias, condiciones climáticas y alturas, las cuales se realizarán utilizando smartphones personales (Xiaomi Redmi 13 C y iPhone 14 pro). Posteriormente, se aplicará el proceso de etiquetado manual, utilizando la herramienta CVAT [9], marcando las regiones de interés de las distintas placas patentes y dividiendo en clases de interés como clase Patente y Patente sin lectura, estas etiquetas, se guardarán en formatos de texto, desde la plataforma CVAT, finalmente, el conjunto de datos se dividirá en subconjuntos para entrenamiento, validación y pruebas del modelo. Para ello se recolectaron 716 imágenes reales, divididas en 510 para entrenamiento, 99 para validación y 107 para pruebas. Para esta etapa, se necesitó la coordinación con diferentes personas ligadas al ejecutor del proyecto, los cuales acompañaron, enviaron y gestionaron, datos para el dataset.

**Tabla 2:** Creación dataset inicial, modelo detección de placas patentes.

Subconjunto	Cantidad de imágenes reales
Entrenamiento	510
Validación	99
Pruebas	107
Total	716

**Metodología Objetivo Especifico 2:** Capturar imágenes del estacionamiento de la Facultad de Ingeniería de la Universidad de Concepción, desde múltiples ángulos (aéreo, frontal y lateral), usando cámaras de smartphones propios (Xiaomi Redmi 13 C y iPhone 14 pro) y dron facilitado por la carrera de Ingeniería Civil en Telecomunicaciones, con el acompañamiento del profesor Felipe Lamas y de Don Luis Arteaga. Las ubicaciones específicas incluyen edificios que están alrededor del estacionamiento de la FI, como lo son los edificios de: Sistemas, Tecnológico Mecánico, Ingeniería e Ingeniería Civil Química, además del espacio de estacionamiento de la FI, para tomas a nivel de suelo y utilización de dron. Las imágenes se tomaron en diversos momentos del día y bajo condiciones variadas tanto de iluminación, alturas, condiciones climáticas, ángulos y distancias, luego de esto, se etiquetaron manualmente estas imágenes, mediante la herramienta CVAT, clasificando estas imágenes en clases Libre y Ocupado, estas etiquetas fueron almacenadas en formato de texto desde CVAT, para su compatibilidad

al modelo YOLO, posteriormente se dividirá el dataset en subconjuntos para entrenamiento, validación y prueba del modelo. Para esta etapa, se coordinó con diversos trabajadores de las respectivas áreas a utilizar, respetando horarios y condiciones entregadas para el desarrollo del procedimiento, por lo cual se tuvo que planificar horarios, gestionar con los trabajadores y acompañantes para la toma de imágenes, evidenciando la importancia de la gestión y trabajo colaborativo en estos procesos.

**Tabla 3:** Creación de dataset de imágenes, modelo de detección de espacios disponibles.

Subconjunto	Cantidad de imágenes reales
Entrenamiento	344
Validación	79
Pruebas	84
Total	507

**Metodología Objetivo Especifico 3:** Se diseñó un modelo de detección de espacios disponibles y ocupados, utilizando arquitectura de YOLOv8, entrenado sobre el dataset previamente construido, complementado por el proceso de data argumentation, el cual nos brinda ampliar nuestra cantidad de datos y variables, generando variaciones de iluminación, posición, rotación, aplicando transformaciones de brillo y contraste, desenfoque gaussiano, CLAHE, entre otros. El modelo es un modelo de aprendizaje supervisado, además de la generación de una detención temprana para evitar sobreajustes y así, detener el proceso cuando deje de mejorar, en este caso para 30 épocas.

**Tabla 4:** Separación de dataset para modelo de detección de espacios disponibles en FI.

Subconjunto	Imágenes reales	Aumentadas	Total de imágenes Dataset
Entrenamiento	344	1378	1722
Validación	79	317	396
Pruebas	84	336	420
Total	507	2031	2538

**Metodología Objetivo Especifico 4:** Se diseñó un modelo para la detección de placas patentes y el reconocimiento óptico de caracteres (OCR), enfocado en patentes chilenas, utilizando arquitectura de YOLOv8, entrenado sobre el dataset previamente construido, complementado por el proceso de data argumentation, el cual nos brinda ampliar nuestra cantidad de datos y variables, generando variaciones de iluminación, posición, rotación, aplicando transformaciones de brillo y contraste, desenfoque gaussiano, CLAHE, entre otros, el modelo es un modelo de aprendizaje supervisado, además de la generación de una detención temprana para evitar sobreajustes y así, detener el proceso cuando deje de mejorar, en este caso para 30 épocas, con la finalidad de detectar las diferentes placas patentes, reconociendo las regiones de interés y posteriormente a esto, segmentar los diversos caracteres mediante Tesseract, para su respectiva lectura, reduciendo errores tanto en la detección de sus separadores (puntos, símbolos, escudos, estrellas y palabras asociadas como Carabineros de Chile o Chile), se evaluó el rendimiento del sistema en condiciones reales, para ello se utilizaron las imágenes previamente separadas y no utilizadas para entrenamiento ni validación.

En este modelo, solo fue aumentada la cantidad de datos de entrenamiento, debido a que debemos considerar condiciones reales para sus respectivas detecciones, ya que alguna modificación de estas puede afectar en la detección de un carácter, lo que origina problemáticas totales en el sistema.

**Tabla 5:** Separación de dataset para modelo de detección de placas patentes y lectura de caracteres.

Subconjunto	Imágenes Reales	Imágenes Aumentadas	Total de imágenes dataset
Entrenamiento	510	1530	2040
Validación	99	-	99
Pruebas	107	-	107
Total	716	1530	2246

**Metodología Objetivo Especifico 5:** Se evaluará el rendimiento del modelo de detección de espacios de estacionamiento entrenado anteriormente, se utilizaron imágenes reales del subconjunto de prueba, las cuales no fueron incluidas en los procesos de entrenamiento ni validación, además se obtuvieron diversas métricas, como

precisión, recall, F1. Score, además de obtener la curva ROC y su respectiva área bajo la curva (AUC), para cuantificar la capacidad del modelo bajo diferentes umbrales de confianza. Además, se observarán sus diferentes visualizaciones, las cuales nos permiten observar el comportamiento real del modelo utilizado.

**Metodología Objetivo Especifico 6:** Se evaluará el rendimiento del modelo de detección de placas patentes y sus respectivos caracteres, se utilizaron imágenes reales del subconjunto de prueba, el cual no fueron incluidas en los procesos de entrenamiento ni validación, además se obtuvieron diversas métricas, como precisión, recall, F1. Score, además de obtener la curva ROC y su respectiva área bajo la curva (AUC), para cuantificar la capacidad del modelo bajo diferentes umbrales de confianza. Además, se observarán sus diferentes visualizaciones, las cuales nos permiten observar el comportamiento real del modelo utilizado.

## **1.7. Alcances y limitaciones**

### **1.7.1. Alcances**

Esta memoria de título busca diseñar modelos inteligentes para la detección de placas patentes y sus respectivos caracteres y por otra parte poder detectar espacios libres u ocupados en el estacionamiento de la FI, lo cual busca optimizar la ocupación de los espacios disponibles y además visualizar su disponibilidad, monitoreando los diversos espacios en tiempo real.

Para ello, se utilizan herramientas de visión computacional, para la detección de placas patentes y sus respectivos caracteres, además de la disponibilidad de espacios de estacionamiento, buscando eficiencia y disminución en los tiempos de búsqueda de los espacios disponibles.

El modelo se desarrollará, exclusivamente para la FI, lo cual tiene un flujo que bordea la capacidad máxima del espacio (57 espacios disponibles), consta con una entrada y una salida, por lo que el desarrollo de este modelo nos brinda la posibilidad de poder escalar a toda la Universidad de Concepción, pudiendo optimizar controles y accesos distintos del campus.

Se busca además, mejorar el uso de espacios de estacionamiento, mejorando la problemática del mal uso de espacios disponibles, como también del uso de espacios reservados, para autoridades o personas en situación de discapacidad, por lo que

mediante este sistema inteligente, se busca gestionar los diversos espacios mediante la visualización de ellos y de los patrones de conductas de los usuarios, mediante la detección de sus placas patentes y sus respectivos caracteres, siendo monitoreados de forma continua visualmente.

### **1.7.2. Limitaciones**

Dentro de las limitaciones con las cuales nos encontramos, fue la problemática inicial, para la toma de imágenes, debido a que se debía contar con permisos para acceder a las distintas dependencias que necesitábamos para tomar las imágenes, considerando alturas, lugares y además la entrada y salida de ellos, por lo cual se tuvo que gestionar con varios profesores, para tener acceso a sus oficinas, salas o laboratorios, coordinar con encargados de edificios y laboratorios, para no interrumpir sus labores y no incomodar en ellas, por lo que el acceso fue restringido a diversas franjas horarias como por ejemplo, llegar a las 7:15 AM, horario en el cual llega el encargado del aseo del edificio de Ingeniería, Don Benjamín, el cual nos permitió antes del inicio de sus tareas, poder acceder a las oficinas a tomar imágenes, ocurriendo lo mismo para el acceso al edificio Tecnológico Mecánico, por otra parte, existieron complicaciones para acceder en horarios de entrada al edificio de Sistemas y de Ingeniería Civil Química, por lo que se tuvo que trabajar en franjas horarias de pocos minutos, luego de finalizada las clases de las 12:00 y quedar a la espera, hasta la finalización de estas actividades a las 17:00, en el cual se nos permitió el último ingreso.

En un inicio, el proyecto está ligado solo a la FI, por lo que no incluye otros espacios dentro del campus de la Universidad de Concepción, por lo que no solucionará las problemáticas de otras dependencias.

Por otra parte, no se pudo acceder a la base de datos de vehículos autorizados al acceso del estacionamiento de la FI, por lo que no se pudo contrastar los datos de personas autorizadas al uso del espacio de estacionamiento, lo que limita la información real para comprobar el uso de las dependencias de la FI.

Dentro de los aspectos que no están enmarcados en la Memoria de Título, es la realización de una aplicación móvil, para visualizar el modelo, en tiempo real, además al no existir cámaras dentro de la FI, con vista a los estacionamientos, se determinarán lugares óptimos para su posible instalación a futuro, debido a que existen limitantes presupuestarias, para la instalación de cámaras de vigilancia dentro del espacio de la FI, lo que no nos permite su instalación e implementación, por lo que el modelo se entrenara,

validara y generaran pruebas mediante el uso de fotografías tomadas por celulares propios, para observar su funcionalidad.

Otra de las limitaciones, es la calidad de imágenes tomadas, por los teléfonos, ya que la calidad de estas entre dispositivos puede variar, además de factores como la accesibilidad, iluminación, complican la toma de imágenes, además de que a cierta distancia, se pierden los distintos márgenes dispuestos en el suelo, por lo que su detección se pierde para ciertos lugares, por otra parte, las restricciones horarias para las capturas de imágenes nos generaron diversas problemáticas, debido a que dependíamos de otras personas, al no tener acceso libre, se tuvo que gestionar horarios con los trabajadores, que en algunas oportunidades, tuvieron atrasos, problemas de salud o licencias médicas, lo que nos generó atrasos en el desarrollo del proceso, además de la imposibilidad de toma de imágenes en horarios nocturnos o de madrugada, para así tener un dataset real más variado.

Las limitaciones, representan una barrera a considerar, lo que puede afectar un correcto desarrollo del procedimiento, por lo que se busca informar desde el inicio, las problemáticas que pueden complicar en mayor medida la ejecución, debido a que no son responsabilidad de la persona que ejecuta el procedimiento, sino que son imponderables que no se pueden solucionar de forma individual.

## **2. Marco teórico**

### **2.1. Introducción**

La utilización de IA, ha aumentado en diversas áreas, lo que ayuda a resolver problemáticas de forma eficiente y en tiempo real, incorporando en este caso, métodos de detección mediante el uso de DL, para automatizar procesos y optimizar recursos, para esto se utilizara un modelo de detección en tiempo real, YOLOv8, el cual nos permite reconocer, tanto espacios libres y ocupados de estacionamientos, como placas patentes y sus caracteres, lo que ayuda a monitorear y gestionar diversas áreas.

El reconocimiento de patentes y sus respectivos caracteres, nos brinda la posibilidad de gestionar y controlar el acceso, considerando a los vehículos que acceden a este espacio dentro de la FI, identificando en tiempo real, los espacios disponibles, lo que ayuda a reducir los tiempos de búsqueda y haciendo más eficiente la entrada y salida de automóviles dentro de la facultad, dado lo anterior, se busca evaluar su efectividad y comportamiento en diversas condiciones, en un entorno real como lo es la FI de la Universidad de Concepción y buscando escalarlo a varias zonas del país, aportando en el desarrollo y gestión de espacios comunes a través de la tecnología.

### **2.2. Conceptos Fundamentales**

**Inteligencia artificial (IA) [10]:** La IA, se centra en desarrollar modelos capaces de realizar tareas que requieren normalmente inteligencia humana, esto incluye procesos de razonamiento lógico, resolución de problemas, aprendizaje de patrones, aprendizaje automático, entre otros, lo que imita el funcionamiento del cerebro humano, para aprender y aplicar dichos conocimientos adquiridos, para ello se utilizan algoritmos, modelos estadísticos o redes neuronales, con la finalidad de que pueda obtener un rendimiento óptimo de forma progresiva a través de la experiencia adquirida a través del aprendizaje y su entrenamiento.

**Machine Learning (ML) [10]:** Es una rama de la inteligencia artificial que permite aprender a partir de datos, sin la necesidad de estar explícitamente programados para realizar cada tarea. Para ello, los algoritmos identifican patrones que permiten generalizar distintos comportamientos y predecir diversos resultados.

Estos algoritmos se pueden clasificar en distintos tipos de aprendizaje. El aprendizaje supervisado requiere que los datos utilizados para el entrenamiento estén previamente

etiquetados. En cambio, el aprendizaje no supervisado trabaja con datos sin etiquetar y busca descubrir patrones como agrupamientos o estructuras internas. Por otro lado, el aprendizaje por refuerzo se basa en la toma de decisiones mediante un proceso de prueba y error.

**Deep Learning (DL) [10]:** Es una subcategoría de ML que utiliza redes neuronales profundas, con múltiples capas capaces de aprender representaciones jerárquicas de los datos. A mayor profundidad, mayor es la capacidad del modelo para aprender patrones complejos. Este proceso imita la forma en que funciona nuestro cerebro, mediante neuronas artificiales, donde las redes aprenden patrones de forma automática, permitiendo procesar información y características más complejas.

**Redes Neuronales Convolucionales (CNN) [11]:** Estas redes están inspiradas en la manera en que el sistema visual humano procesa estímulos. Esto permite reconocer diferentes patrones presentes en imágenes, siendo el motor principal del Deep Learning en aplicaciones de visión artificial.

**Visión Artificial [12]:** Es un área de la inteligencia artificial que permite a las máquinas interpretar el entorno visual a partir de imágenes o videos. Involucra tareas como detección de objetos, segmentación de imágenes, clasificación, seguimiento de movimiento y análisis espacial.

**YOLOv8 (You Only Look Once, versión 8) [13]:** Es un modelo de detección de objetos desarrollado por Ultralytics y lanzado en enero de 2023. Es la octava versión de esta arquitectura, caracterizada por realizar tareas de visión computacional, como clasificación, detección y segmentación de imágenes, todo en tiempo real, lo cual lo hace atractivo debido a su precisión y velocidad.

**Reconocimiento Óptico de Caracteres (OCR) [8]:** Es una tecnología que permite extraer texto desde una imagen. Utiliza técnicas como segmentación, normalización y clasificación para transformar píxeles en caracteres que pueden ser reconocidos por una máquina.

**Verdaderos Positivos (TP) [14]:** Casos en los que el modelo identifica correctamente una instancia perteneciente a la clase positiva.

**Verdaderos Negativos (TN) [14]:** Situaciones en las que el modelo identifica correctamente que una instancia no pertenece a la clase positiva.

**Falsos Positivos (FP) [14]:** Ocurren cuando el modelo identifica erróneamente una instancia negativa como positiva.

**Falsos Negativos (FN) [14]:** Se presentan cuando el modelo no detecta una instancia positiva y la clasifica incorrectamente como negativa.

**Precisión [15]:** Proporción de predicciones positivas realizadas por el modelo que son efectivamente correctas.

$$\textit{Precisión} = \frac{\textit{TP}}{\textit{TP} + \textit{FP}}$$

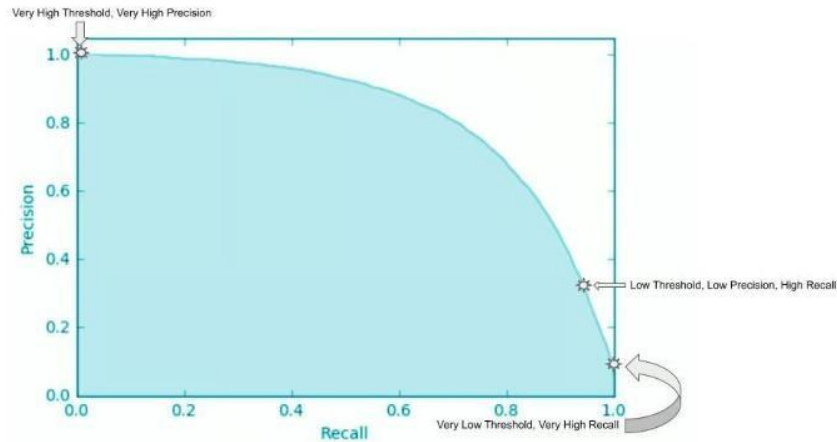
**Recall (Sensibilidad o Tasa de Verdaderos Positivos) [16]:** Mide la capacidad del modelo para identificar correctamente todas las instancias positivas reales.

$$\textit{Recall} = \frac{\textit{TP}}{\textit{TP} + \textit{FN}}$$

**F1 Score [17]:** Métrica que combina *precisión* y *recall* en un único valor, utilizando la media armónica entre ambas. Busca reflejar el equilibrio entre cuán precisas son las predicciones y cuántas instancias positivas fueron correctamente detectadas.

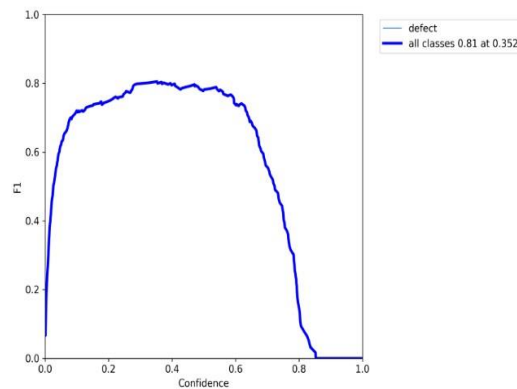
$$\textit{F1 Score} = 2 * \frac{\textit{Precision} * \textit{Recall}}{\textit{Precision} + \textit{Recall}}$$

**Curva Precisión–Recall:** Esta curva representa el comportamiento conjunto de ambas métricas a lo largo de distintos umbrales de decisión, permitiendo visualizar cómo se afectan mutuamente.



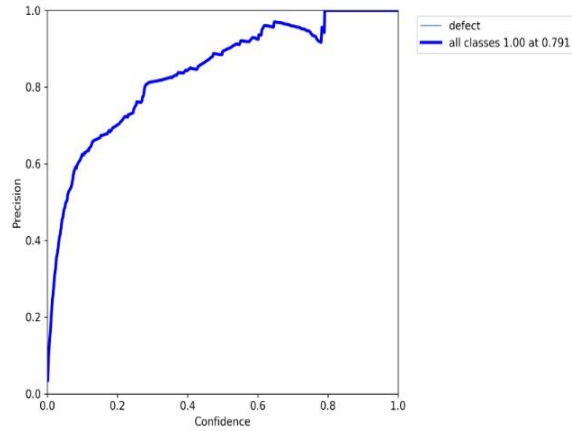
**Figura 2:** Precisión vs Recall, imagen obtenida de [28]

**Curva F1 Score vs Confianza:** La curva F1 frente al umbral de confianza ofrece una visualización de cómo se equilibran *precisión* y *recall* a lo largo de distintos niveles de confianza.



**Figura 3:** F1- Score vs Confianza, imagen obtenida de [29]

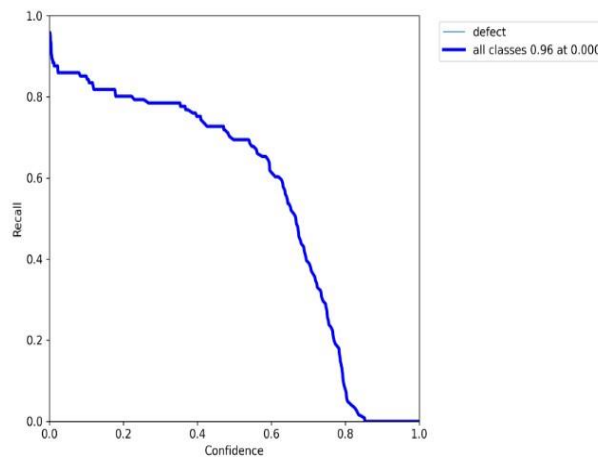
**Curva Precisión vs Confianza:** La curva de precisión en función del umbral de confianza permite observar cómo varía la proporción de verdaderos positivos sobre el total de predicciones positivas, al ajustar el umbral de aceptación. Es útil para controlar la aparición de falsos positivos, ya que permite ver cómo se comporta la precisión al exigir mayor certeza en las predicciones.



**Figura 4:** Precisión vs Confianza, imagen obtenida de [29].

### Curva Recall vs Confianza

La curva de recall respecto al umbral de confianza muestra cómo varía la capacidad del modelo para detectar correctamente los casos positivos reales al modificar el umbral exigido. Disminuir el umbral hace que el modelo sea más permisivo, aumentando el *recall*, ya que incluye más verdaderos positivos.



**Figura 5:** Recall vs Confianza, imagen obtenida de [29].

**Curva ROC [30] [31]:** La curva ROC (Receiver Operating Characteristic) evalúa el rendimiento de un modelo de clasificación binaria bajo distintos umbrales de decisión. El

modelo entrega una probabilidad de pertenencia a la clase positiva, la cual se convierte en una decisión binaria al recorrer distintos umbrales entre 0 y 1: si la probabilidad supera el umbral, se clasifica como positiva; en caso contrario, como negativa.

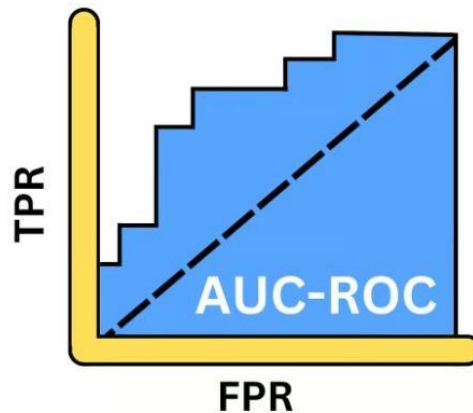
Para cada umbral se calcula la Tasa de Verdaderos Positivos (TPR), que corresponde a la proporción de verdaderos positivos sobre el total de positivos reales:

$$TPR = \frac{TP}{TP+FN}$$

También se calcula la Tasa de Falsos Positivos (FPR), que representa la proporción de falsos positivos sobre el total de negativos reales:

$$FPR = \frac{FP}{FP+TN}$$

Además, se calcula el AUC (Área Bajo la Curva), que resume en un solo valor la capacidad del modelo para distinguir entre clases. Un AUC igual a 1 representa un modelo perfecto, mientras que un AUC cercano a 0.5 indica un comportamiento similar al azar.



**Figura 6:** Curva ROC, imagen obtenida de [30].

Si bien la obtención de estas gráficas se realiza para una sola clase, las métricas pueden extrapolarse a múltiples clases, lo que constituye una herramienta clave para interpretar el comportamiento general del modelo y establecer criterios de evaluación.

### **2.3. Estado del Arte**

Diversas investigaciones han demostrado la efectividad del uso de técnicas de Deep Learning y visión computacional en tareas relacionadas con el control vehicular y la gestión de estacionamientos en entornos urbanos complejos. García Guerrero (2022) [18] desarrolló un sistema inteligente para la detección automática de espacios disponibles, empleando redes neuronales convolucionales (CNN) conectadas a una interfaz de cámaras en tiempo real, alcanzando tasas de precisión superiores al 90% bajo condiciones reales de operación.

De forma similar, Mariotti (2020) [19] aplicó CNN para la detección e identificación de vehículos, validando su efectividad en escenarios urbanos caracterizados por su alta complejidad. Por otra parte, Vizcaya Cárdenas (2018) [20] propuso un modelo de detección simultánea de peatones y vehículos mediante técnicas de Deep Learning, con énfasis en su desempeño bajo condiciones de baja visibilidad y durante horarios nocturnos, lo que constituye un aspecto crítico para garantizar la robustez de sistemas implementados en exteriores.

Desde el ámbito comercial, la empresa Urbiotica [21] ha desarrollado soluciones basadas en sensores y cámaras, integrando software de visión artificial potenciado por redes neuronales profundas para la gestión de estacionamientos y reconocimiento de patentes. No obstante, pese a su alta eficacia, estas soluciones presentan elevados costos de implementación, lo que dificulta su adopción en contextos institucionales educativos.

En relación con el reconocimiento de placas patentes, esta tecnología se ha posicionado como una herramienta fundamental en aplicaciones de seguridad vial, fiscalización automatizada, control de acceso y sistemas inteligentes de detección. Su evolución ha transitado desde enfoques basados en anotaciones manuales hacia metodologías basadas en aprendizaje profundo y arquitecturas convolucionales, incrementando significativamente los niveles de automatización y precisión. En este contexto, Del Pino Castillo (2021) [22] evaluó distintas arquitecturas contemporáneas como YOLOv3, Faster R-CNN y SSD Inception. Su análisis concluyó que YOLOv3 ofrece un balance adecuado entre velocidad de inferencia y precisión, resultando especialmente adecuado para implementaciones en tiempo real.

Adicionalmente, Torres y Riveros (2023) [23] desarrollaron un sistema orientado a la detección de placas clonadas en la ciudad de Bogotá, integrando modelos de detección vehicular basados en YOLO, técnicas de clasificación por color y tipo de carrocería, y

componentes OCR para contrastar la información visual con los datos registrados de la matrícula. Esta solución permitió identificar casos de suplantación vehicular mediante comparación cruzada de atributos visuales, ampliando el potencial del control vehicular automatizado.

En un enfoque complementario, San Martín Mora (2024) [24] implementó un sistema de gestión de estacionamientos utilizando visión computacional, donde se empleó YOLOv5 para la detección de vehículos que ocupaban plazas. Esta solución se integró con una aplicación móvil desarrollada en Flutter, permitiendo el monitoreo en tiempo real y facilitando una administración eficiente de los espacios disponibles. El proyecto incorporó también técnicas de aumento de datos con el fin de mejorar la capacidad de generalización del modelo ante variaciones lumínicas y de perspectiva.

Finalmente, en el ámbito del control de acceso, Carreras Franco (2018) [25] diseñó una arquitectura modular denominada QBarrier, la cual combina detección de vehículos mediante YOLO con reconocimiento óptico de caracteres a través de Tesseract OCR, integrando ambas herramientas para una gestión automatizada y eficiente del ingreso vehicular.

## **2.4. Justificación del Enfoque**

En el contexto de esta memoria de título, se utilizarán técnicas basadas en Deep Learning, bajo la arquitectura de YOLOv8s, la cual proporciona la capacidad de detectar objetos y espacios independientemente de las condiciones presentes en los datos, como iluminación, ángulos, alturas, condiciones climáticas, baja visibilidad, entre otros factores. A través del uso de redes neuronales convolucionales, el modelo aprende patrones y secuencias directamente desde los datos, lo que permite su adaptabilidad para aplicaciones en tiempo real, como la detección de placas patentes, sus caracteres y la disponibilidad de espacios de estacionamiento.

A su vez, YOLO permite un balance entre rendimiento y costo computacional, lo cual es fundamental para mantener una precisión suficiente en las distintas detecciones, siendo una solución factible para ser ejecutada en computadores personales.

Luego de este proceso, se trabajará en la fase de lectura de caracteres, para la cual se utilizará la herramienta Tesseract OCR. Esta permite leer y extraer texto desde las distintas placas patentes detectadas en las imágenes procesadas por el modelo.

### 3. Desarrollo

#### 3.1. YOLOv8 [32] [33]

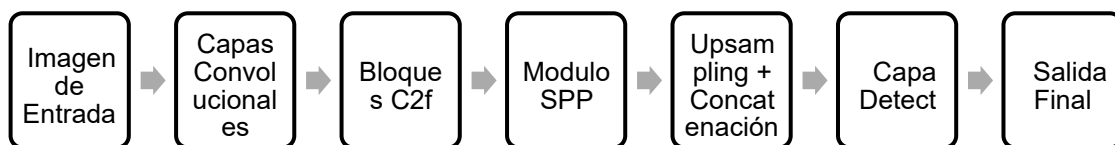
YOLOv8 (You Only Look Once, versión 8) es un modelo avanzado de detección de objetos desarrollado por Ultralytics, orientado a tareas de visión por computador como clasificación, segmentación y detección en tiempo real. Está completamente implementado en PyTorch y ofrece mejoras significativas en precisión, velocidad y facilidad de uso respecto a versiones anteriores, lo que lo convierte en una herramienta ideal para aplicaciones como el reconocimiento de patentes y la gestión automatizada de estacionamientos.

La versión YOLOv8s (small) cuenta con 225 capas y aproximadamente 11.1 millones de parámetros entrenables. Su arquitectura se basa en redes neuronales convolucionales y emplea bloques C2f para mejorar la eficiencia en la extracción y combinación de características, sin aumentar excesivamente la complejidad del modelo.

Uno de sus componentes clave es el módulo SPPF (Spatial Pyramid Pooling Fast), que permite capturar información a distintas escalas sin elevar el costo computacional. A esto se suman mecanismos de upsampling y concatenación inspirados en arquitecturas como FPN y PANet, lo que mejora el rendimiento en la detección de objetos pequeños y en condiciones visuales difíciles.

La capa final, Detect, genera las predicciones a partir de tres escalas diferentes y entrega las coordenadas de las cajas junto a la probabilidad de cada clase. En el contexto del proyecto, se ha configurado para distinguir entre dos clases: espacios de estacionamiento ocupados y espacios de estacionamiento libres.

YOLOv8s destaca por su equilibrio entre rendimiento y eficiencia, siendo especialmente útil en dispositivos con recursos limitados sin comprometer la calidad de las detecciones.



**Figura 7:** Funcionamiento de YOLOv8.

### 3.2. Modelo de detección de patentes y lectura de caracteres

Nuestro modelo fue realizado en primera instancia con la recolección de imágenes, para crear un dataset propio y local, con las especificaciones de patentes chilenas, conociendo y entendiendo como operan, en este caso, las diferentes características, tanto de colores como de caracteres para los diversos automóviles con patentes nacionales, los cuales, desde el año 2007 [26] siguen la lógica de 6 caracteres para automóviles, tanto para civiles, como para las policías, los cuales consideran 4 letras seguidas por dos números, además de contar con una separación cada 2 caracteres, el cual puede ser un punto medio para civiles y una estrella para policía, que fueron los casos que consideraremos, en la creación de nuestro dataset.



**Figura 8:** Patentes chilenas, imagen obtenida de Radio Bio Bio [27].

El lenguaje de programación utilizado para nuestro modelo fue Python, mediante el entorno de desarrollo de PyCharm, en el cual, se utilizó la herramienta de Yolov8, en el proceso de entrenamiento para la detección de la placa patente y su respectiva lectura de caracteres, se consideraron varios procesos, luego de la creación del dataset, se utilizó la herramienta CVAT, para crear etiquetas para la detección de nuestras placas patentes y hacerlas compatibles para la versión Yolo, dentro de las clases consideradas fueron las clases Patente, para una patente con completa detección y clase patente sin lectura, las cuales observábamos obstrucciones hacia su lectura, lo que iba a generar error en la detección, debido a que si existe un carácter erróneo en el proceso, la patente sería inválida, por lo que en estos casos, la lectura no sería posible.

El entrenamiento se llevó a cabo durante 100 épocas, aplicando técnicas de aumento de datos mediante Albumentations. Estas técnicas incluyeron transformaciones de brillo,

contraste, compresión, inversión y desenfoque, simulando diversas condiciones reales de captación. Además, se incorporó un preprocesamiento específico que consistió en la conversión a escala de grises, aplicación de filtros CLAHE para mejorar el contraste local y binarización adaptativa, con el fin de optimizar la posterior etapa de reconocimiento óptico de caracteres (OCR).



**Figura 9:** Ejemplo imagen clase PATENTE SIN LECTURA, imagen propia.



**Figura 10:** Ejemplo clase PATENTE, imagen propia.

### **3.3. Modelo de detección de espacios de estacionamiento, Libre u Ocupado**

Durante la segunda parte del proyecto, se entrenó un modelo para la detección de espacios disponibles, libres y ocupados, utilizando la herramienta de YOLOv8s, el cual busca identificar dicha ocupación, dentro de la Facultad de Ingeniería de la Universidad de Concepción, el cual consta con dos clases: Libre y Ocupado, este dataset fue creado a partir de imágenes dentro del entorno de la Facultad, dado 4 edificios aledaños al

espacio a utilizar, como lo es el Edificio Tecnológico Mecánico, Edificio Ingeniería Civil Química, Edificio de Sistemas y Edificio de Ingeniería, en diversas condiciones, tanto de altura, iluminación, ángulos, obstrucciones, condiciones climáticas, entre otras, lo que nos permite entregar mayor información a nuestro entrenamiento, haciendo un proceso más robusto en su aprendizaje. Inicialmente nuestro modelo contaba con muchas clases, que no aportaban valor real, y nos producían problemáticas, debido a la poca cantidad de información que aportaban dichas imágenes, como, por ejemplo, vehículos mal estacionados u otros que ocupaban doble espacios, o malas ocupaciones, por lo que se optó por eliminar dichas clases, realizando una limpieza de ellas para su no consideración.

Además, con el objetivo de aumentar el volumen de nuestro dataset, lo que nos permitirá tener un modelo más robusto, se aplicó un aumento de datos, lo cual simulaba variaciones, como, por ejemplo, brillo, contraste, desenfoces u otros cambios, lo que permite una real adaptación a cambios en las posteriores condiciones que se pueden presentar, entregando variados escenarios.

El algoritmo fue entrenado utilizando transfer learning, dado pesos preentrenados mediante yolov8.pt, para ello, se generó una entrada de 640 x 640 pixeles, ejecución de 150 épocas con paciencia de 30 épocas, lo que nos permite detener el entrenamiento, si no existe una mejora en dichas épocas, así evitando el sobreajuste en el modelo. El batch para este proceso fue de 8 imágenes, utilizando además memoria de GPU, en un entorno CUDA, lo que nos permite generar un proceso más acelerado, optimizando la capacidad de nuestro computador.

Para asegurar que el monitoreo sea aplicable en un entorno real, buscando la aplicabilidad, se aplicaron criterios para la selección de puntos más adecuados, los cuales consideran factores como visibilidad, cuanta área abarca la imagen, altura de toma, obstrucciones y el ángulo de imagen, para ello, se consideraron los 4 edificios contiguos al sitio donde queremos evaluar y se tomaron distintas imágenes, dentro de las consideraciones comentadas anteriormente, por otra parte, se utilizó un dron para evaluar los espacios desde las alturas, lo que fue posible gracias al apoyo de la carrera Ingeniería Civil en Telecomunicaciones, a través del acompañamiento del Señor Luis Arteaga y el profesor Felipe Lamas, estas tomas aéreas nos permitieron obtener vistas panorámicas casi en un 100%, debido a que en las tomas no consideran 2 estacionamientos al lado de Edificio Tecnológico Mecánico, pero nos ayuda en un inicio

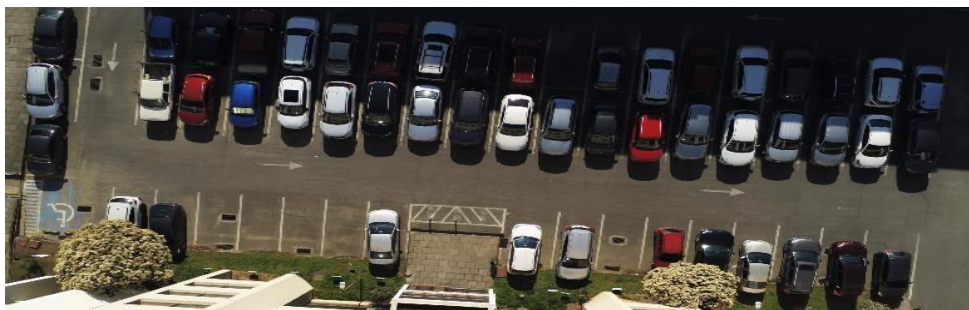
a planificar y diseñar el sistema, además de imágenes de buena calidad para nuestro dataset.



**Figura 11:** Imagen desde Edificio de Ingenieria, imagen propia.



**Figura 12:** Imagen desde Edificio TM, imagen propia.



**Figura 13:** Imagen desde altura mediante dron, imagen propia.



**Figura 14:** Imagen desde Edificio Sistemas, imagen propia.



**Figura 15:** Imagen desde Edificio ICQ, imagen propia.

Luego de esta recolección de imágenes para nuestro dataset, se evaluaron las diversas posiciones de las cuales obtuvimos nuestras imágenes, observando diversos criterios para su evaluación, según su ubicación, posición, lugar, alturas, visibilidad, ángulo de cámaras y obstrucciones que existen en el espacio disponible, este paso es útil para un posible trabajo a futuro, el cual puede generar la mejor ubicación para la instalación de cámaras dentro del espacio disponible en el estacionamiento de la Facultad de Ingeniería de la Universidad de Concepción, y sus edificios que están contiguos a dicho espacio.

**Tabla 6:** Criterios toma de imágenes, estacionamiento FI.

Ubicación	Ángulo de cámara	Visibilidad	Altura utilizada (pisos)	Obstrucciones	Observaciones
Edificio Ing. Civil Química	Frontal	Buena	4 a 5	Mínimas	Ideal para control completo del estacionamiento.
Edificio Sistemas	Lateral parcial	Reducida	2 a 3	Estructuras	Limitado a una sección del estacionamiento.
Tecnológico Mecánico	Frontal	Perdida Parcial lateral y profundidad	2 a 3	Estructuras	Tiene perdida parcial de estacionamientos laterales y perdida en profundidad.
Vista aérea con dron	Panorámico	Casi Completa	No aplica	Ninguna	Útil para planificar y dividir espacios.
Edificio Ingeniería	Limitado, angular o central	Reducida	2 a 4	Estructuras y vegetación	No funcional para visión directa, limitado, mejor detección frente, pero pierde profundidad

## 4. Resultados

Considerando que el presente estudio corresponde a evaluaciones a escala laboratorio por lo cual se presentará un estado del arte y dos consideraciones a lo largo del presente proyecto explicadas a continuación.

### 4.1. Resultados de detección de Placas Patentes

#### 4.1.1. Resultados de simulación

Tras finalizar el entrenamiento se observa una disminución continua de las funciones de pérdida en la fase inicial lo que indica un aprendizaje efectivo. El modelo demuestra capacidad para identificar patrones relevantes estabilizándose posteriormente para generalizar frente a variaciones de imagen iluminación y otras condiciones ambientales.

La curva F1-score vs confianza muestra que el valor máximo del F1-score se alcanza alrededor de un umbral de confianza de 0.67 y luego disminuye bruscamente lo que sugiere que umbrales más altos reducen significativamente el recall sin aumentar de manera relevante la precisión.

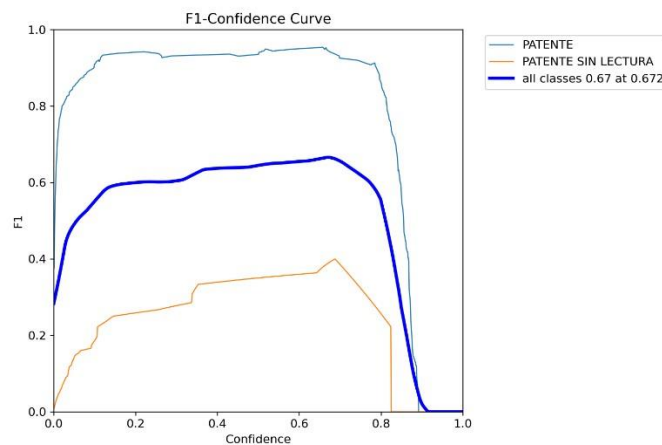
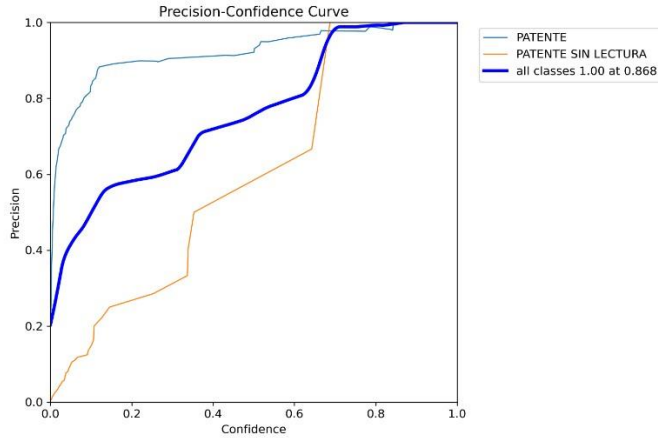


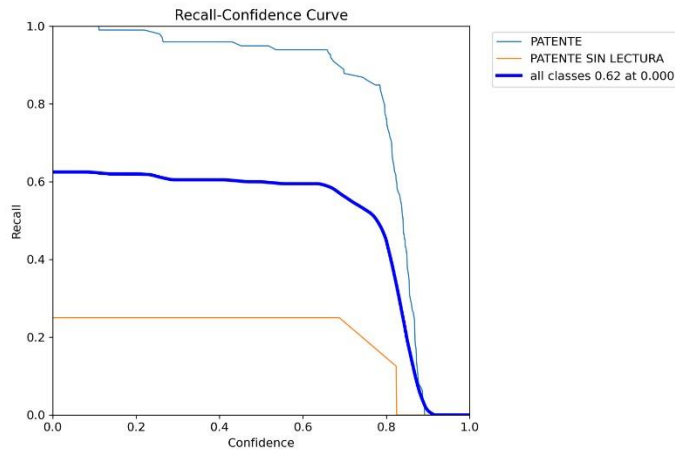
Figura 16: F1 vs Confianza.

La curva precisión vs confianza evidencia un incremento sostenido en la precisión alcanzando un valor de 1.0 aproximadamente al llegar a un umbral de 0.87 lo que indica que todas las predicciones por encima de ese umbral son correctas, aunque probablemente con un recall muy bajo.



**Figura 17:** Precisión vs Confianza.

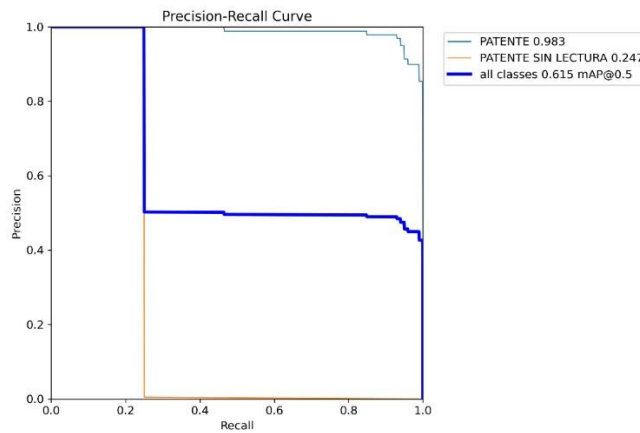
La curva recall vs confianza refleja un comportamiento distinto entre clases observándose un mejor rendimiento para la clase PATENTE y un desempeño limitado para la clase PATENTE SIN LECTURA esto indica que el modelo tiene dificultades para reconocer correctamente patentes poco visibles o con baja calidad lo que puede estar relacionado con un desbalance del dataset o menor número de ejemplos representativos.



**Figura 18:** Recall vs Confianza.

En la curva precisión vs recall, se observa que la clase PATENTE mantiene una alta precisión incluso a niveles elevados de recall lo cual demuestra una buena capacidad del modelo en esa clase mientras que la clase PATENTE SIN LECTURA tiene bajo recall y mayor variabilidad en su precisión lo que indica inestabilidad y problemas de

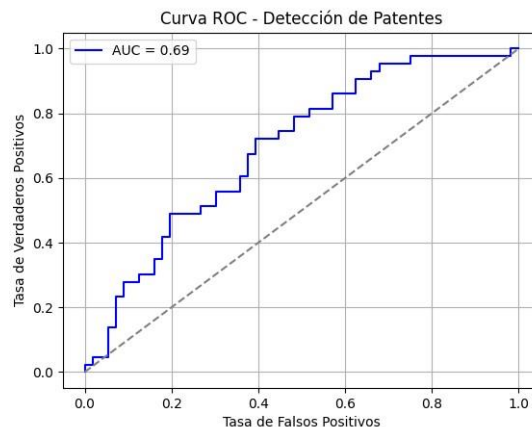
generalización el mAP0.5 alcanza 0.615 lo que representa un rendimiento aceptable para aplicaciones de detección real.



**Figura 19:** Precisión vs Recall.

La curva ROC presenta una forma zigzagueante con un área bajo la curva AUC de 0.69 esta forma irregular puede explicarse por una baja cantidad de muestras, una salida de modelo poco continua o por un dataset limitado que impide una mejor discriminación entre clases un AUC de 0.69, esto indica un rendimiento moderado por sobre el azar pero aún insuficiente para aplicaciones críticas.

Para mejorar este modelo se debe ampliar y balancear el dataset, aplicar técnicas de calibración de probabilidades, ajustar los hiperparámetros o explorar arquitecturas más robustas, además, puede evaluarse el uso de validación cruzada para asegurar la generalización del modelo.



**Figura 20:** Curva ROC.

### **4.1.2. Evaluación humana**

Durante la evaluación manual se evidenciaron errores, inicialmente, en la lectura de caracteres ya que un error en un solo carácter invalida completamente la lectura de la patente lo que podría generar problemas legales en aplicaciones reales. También se observaron errores en la detección de separadores de caracteres que el modelo identificaba incorrectamente como elementos válidos.



**Figura 21:** Detección inicial fallida.

Como medida de mitigación se implementaron filtros que validan únicamente estructuras de caracteres específicas según el tipo de patente y se omiten los separadores, como puntos, estrellas o frases como por ejemplo "CHILE," "CARABINEROS DE CHILE", lo cual mejoró considerablemente los resultados. Sin embargo, persisten fallos en condiciones adversas como lluvia intensa o iluminación deficiente, pero en condiciones normales aumentó considerablemente el nivel de detección debido a que en las primeras detecciones. Era muy común para la detección de los caracteres separadores, que nos entregara de vuelta en la visualización, una detección de caracteres "basura". Por lo que se vio necesario ajustar el modelo para que disminuyera la tasa de error en las detecciones, debido a que perjudicaba al modelo en su totalidad, haciéndolo inestable e inaplicable.



Figura 22: Detecciones de placas patentes.



Figura 23: Detecciones de placas patentes.



Figura 24: Detecciones patentes no elegibles.

## 4.2. Resultados de detección de espacios disponibles

### 4.2.1. Resultados de simulación

La curva F1-score vs confianza muestra un valor máximo de F1-score de 0.42 aproximadamente en un umbral de 0.19, posteriormente este valor disminuye a medida que el umbral aumenta lo que refleja una pérdida en la sensibilidad del modelo, la clase LIBRE presenta un mejor F1-score que la clase OCUPADO lo que indica una mayor estabilidad en la detección de espacios desocupados.

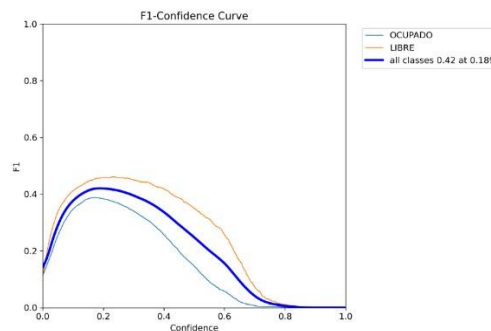
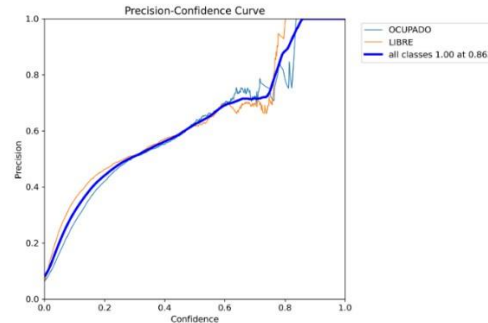


Figura 25: F1- Score vs Confianza.

La curva precisión vs confianza revela que la precisión mejora a medida que aumenta el umbral alcanzando un valor de 1.0 alrededor de 0.863, esto significa que las predicciones sobre este valor son correctas, pero probablemente a costa de un bajo recall.

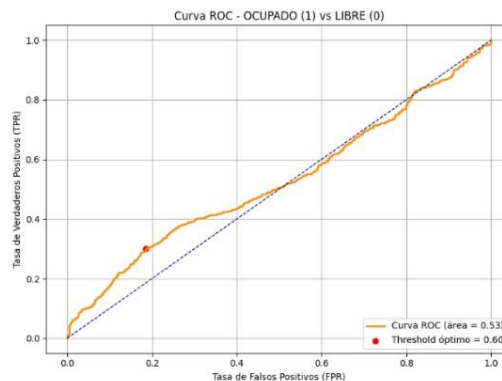


**Figura 26:** Precisión vs Confianza.

La curva ROC para este modelo muestra un AUC de 0.53 por sobre el azar la curva, se mantiene cerca de la diagonal lo que señala que el modelo tiene poca capacidad para discriminar entre clases el umbral óptimo identificado en 0.60, es decir, no mejora significativamente el resultado.

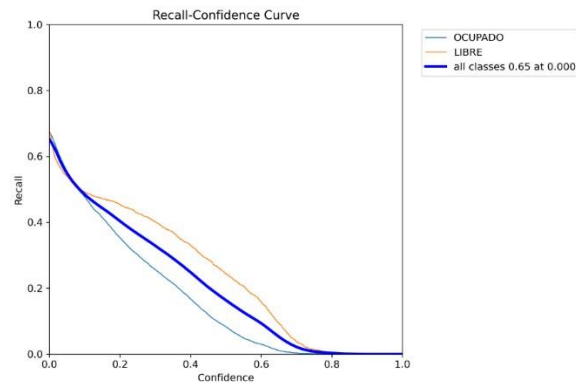
Entre las posibles causas de este bajo rendimiento se encuentra un desbalance entre clases, problemas en el etiquetado una arquitectura inadecuada o imágenes de baja calidad que dificultan el entrenamiento del modelo.

Para mejorar este modelo es necesario revisar el proceso de etiquetado, aplicar técnicas de aumento de datos, balancear las clases del conjunto de entrenamiento y ajustar los hiperparámetros. Actualmente el modelo no es viable para ser utilizado en producción, pero presenta potencial de mejora si se abordan estos problemas.



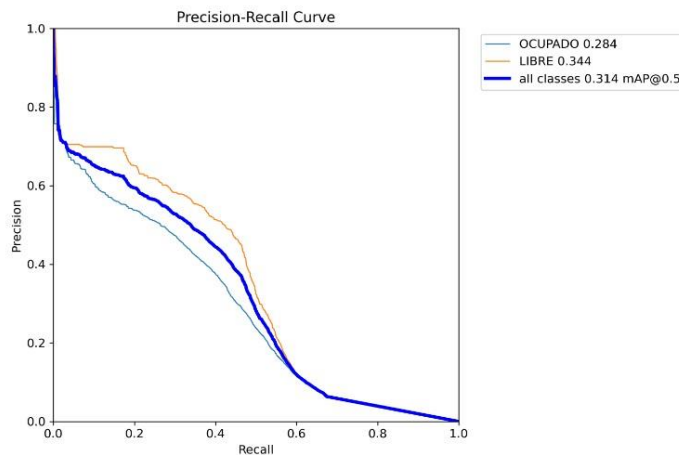
**Figura 27:** Curva ROC.

La curva recall vs confianza indica que el recall es alto a umbrales bajos, pero disminuye rápidamente al aumentar la confianza en especial para la clase OCUPADO, lo cual revela que el modelo tiene dificultades para identificar con certeza los espacios ocupados. Esto podría estar relacionado con sombras, estructuras visuales confusas u objetos similares a vehículos.



**Figura 28:** Recall vs Confianza.

La curva precisión vs recall muestra que la clase LIBRE logra una precisión de 0.344 mientras que la clase OCUPADO solo alcanza 0.284, el mAP@0.5 promedio es de 0.314, la clase LIBRE mantiene mejor estabilidad en la precisión mientras que la clase OCUPADO muestra caídas bruscas a medida que aumenta el Recall, lo que evidencia menor capacidad del modelo para detectar espacios ocupados.

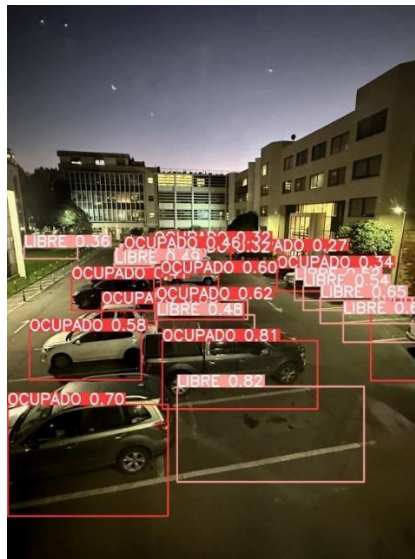


**Figura 29:** Precisión vs Recall.

## 4.2.2. Evaluación Humana

Durante la inspección visual se observaron errores por solapamiento de líneas tapadas por vehículos, así como detecciones erróneas en elementos no válidos como luminarias, también hubo omisiones en espacios válidos no detectados por obstrucciones o detecciones realizadas fuera de zonas habilitadas como áreas verdes.

Se recomienda mejorar la definición de zonas válidas a través de filtros espaciales para evitar la detección de ocupación en sectores no permitidos y con ello reducir los errores de interpretación del sistema.



**Figura 30:** Detección espacios de estacionamiento.



Figura 31: Detección de espacios de estacionamiento.



**Figura 32:** Detección de espacios de estacionamiento.

### **4.3. Estimación económica**

El desarrollo e implementación del modelo, para la detección de placas patentes y para la detección de espacios de estacionamiento disponibles tiene una serie de costos asociados, tanto en la fase de desarrollo del modelo como en su implementación dentro de la FI. A continuación, se presentan de manera detallada los ítems considerados en la evaluación económica, con valores estimados, obtenidos desde referencias reales y actualizadas.

#### **4.3.1. Costos de desarrollo de modelos**

Esta etapa comprende los recursos necesarios para la preparación del conjunto de datos, el entrenamiento del modelo y las pruebas de validación.

Captura de datos e imágenes: Para entrenar los modelos de visión por computador, se requiere un conjunto extenso de imágenes representativas de placas patentes y zonas de estacionamiento. La toma de imágenes se estima en 100 horas de trabajo en terreno, considerando desplazamientos, captura manual y organización del material. El valor hora hombre se estima en \$6000, dando un total de:  $100 \text{ horas} \times \$6000 = \$600000$ , a su vez siguiendo esta misma área del proyecto, se deberá etiquetar manualmente utilizando la

plataforma CVAT, las distintas zonas de interés para cada modelo, el cual utilizo 100 horas de trabajo a un valor de \$6000,  $100 \text{ horas} \times \$6000 = \$600000$ .

Equipamiento computacional: Se requiere un equipo con GPU dedicada para el entrenamiento eficiente de los modelos. En este caso, se considera un computador portátil HP Omen 16 con GPU NVIDIA RTX, procesador i7 y 16 GB de RAM, cuyo valor de mercado es de aproximadamente \$2070000.

Durante el proceso de desarrollo se utiliza el entorno de desarrollo PyCharm estudiantil, el cual tiene un costo de \$0, pero existe un entorno, de la misma plataforma, PyCharm pero del tipo profesional, el cual su valor para la licencia anual tiene un costo de 249 dólares, lo que equivale a aproximadamente \$249000 considerando el valor del dólar hoy en día, tiene un valor aproximado a \$1000.

Costos operacionales estimados: Durante el entrenamiento del modelo se contempla un consumo constante de energía eléctrica y conexión a internet, se estima un gasto total de \$50.000 entre luz e internet utilizado.

**Tabla 7:** Costos desarrollo de modelos.

Implemento o acción	Cantidad u Horas	Costo	Costo Total
Captura de datos e imágenes	100	\$6000	\$600000
Etiquetado de imágenes	100	\$6000	\$600000
Computador portátil HP Omen 16	1	\$2070000	\$207000
PyCharm	Académico	\$0	\$0
PyCharm	Profesional	\$249000	\$249000
Costo luz y electricidad	1 mes	\$50000	\$50000

**Tabla 8:** Costo desarrollo de modelos para caso a y b.

Caso	Costo total	Considerando 10% imponderables
a) Pycharm académico	\$3320000	\$3652000
b) Pycharm profesional	\$3569000	\$3925900

### **4.3.2. Costos de implementación de modelos**

Una vez desarrollado el modelo, se considera su implementación en un entorno real, lo que implica la adquisición e instalación de cámaras de vigilancia, montaje técnico, materiales eléctricos y puesta en marcha del sistema.

Para reflejar de mejor forma la realidad del mercado, se utilizaron precios referenciales actualizados desde tiendas Sodimac y Easy, lo que permite proyectar un presupuesto realista considerando estándares profesionales.

**Tabla 9:** Costo de materiales para implementación de modelos.

Implemento o acción	Detalle	Costo unitario o costo por hora	Costo Total
Cámaras PTZ Hikvision x3	3 cámaras, para instalación en Edificio ICQ, Edificio TM, acceso estacionamiento FI	\$2510000	\$7530000
Costo de Trabajadores	3 trabajadores por instalación	\$300000	\$300000
Canaletas eléctricas	Canaletas plásticas 100x50 mm, 2m (Sodimac)	45 x \$4790	\$215550
Tubería conduit metálica	2 rollos de 50 metros PVC flexible (Sodimac)	2x 86898	\$173796

Accesorios para PVC	Codos, conectores y curvas (60 unid. x \$682 c/u, Sodimac)	60 x \$682	\$40920
Soportes metálicos y anclajes	Fijación de cámaras y canaletas.		\$100000
Elementos de seguridad laboral	Arnés, casco, guantes, andamio, elementos de seguridad		\$150000
Materiales eléctricos generales	Cajas, interruptores, fusibles, conectores.		\$90000
Configuración de red y calibración inicial	8 horas de técnico x \$15000	8x\$15000	\$120000

**Tabla 10:** Costo total de materiales para implementación de modelos.

Costo Total	10 % imponderables	Costo Total
\$8720266	\$872027	\$9592293

### **4.3.3. Total general Proyecto**

Una vez desarrollado el modelo, se considera su implementación en un entorno real, lo que implica la adquisición e instalación de cámaras de vigilancia, montaje técnico, materiales eléctricos y puesta en marcha del sistema.

**Tabla 11:** Costo desarrollo de modelo e implementación, para caso a.

Ítem	Costo	10 % imponderables	Costo Total

Modelo con PyCharm Estudiantil	\$3320000	\$332000	\$3652000
Implementación	\$8720266	\$872027	\$9592293
Total	\$12040266	\$1204027	\$13244293

**Tabla 12:** Costo desarrollo de modelo e implementación, para caso b.

ítem	Costo	10 % imponderables	Costo Total
Implementación	\$8720266	\$872027	\$9592293
Modelo con PyCharm Profesional	\$3569000	\$356900	\$3925900
Total	\$12289266	\$1228927	\$13518193

#### **4.3.4. Observaciones generales**

Los costos presentados corresponden a una posible implementación del modelo, dentro de la FI. No se consideran gastos asociados al mantenimiento periódico, almacenamiento en la nube, procesamiento en servidores externos ni licencias de sistemas operativos.

La estimación de costos se ajustó con precios actualizados del mercado, obtenidos desde tiendas nacionales como Sodimac y Easy, garantizando así una evaluación más representativa de una instalación real, considerando además un 10% en el costo total, en caso de imponderables, que puedan afectar su implementación.

#### **4.4. Discusión**

El análisis de los resultados obtenidos permite identificar fortalezas y debilidades en el desempeño de los modelos desarrollados. El sistema de detección de placas patentes presentó un rendimiento óptimo para aplicaciones en condiciones normales, alcanzando un AUC de 0,69 y un mAP@0.5 de 0,615, lo que demuestra una capacidad para reconocer patentes válidas. No obstante, se evidenciaron limitaciones en condiciones adversas, tales como iluminación deficiente o lluvia intensa, así como dificultades en la detección de caracteres parcialmente obstruidos. Estas limitaciones pueden abordarse mediante la ampliación y balanceo del dataset, ajustes en los hiperparámetros y mejoras en el preprocesamiento de imágenes.

Por otro lado, el modelo de detección de espacios disponibles mostró un rendimiento más bajo, con un AUC de 0,53 y un mAP@0.5 promedio de 0,314, indicando una capacidad limitada para diferenciar de forma confiable entre espacios libres y ocupados. Este resultado se asocia principalmente a problemas de calidad de imágenes, toma de imágenes, debido a que existen lugares donde se pierden demarcaciones de estacionamientos, producto de obstrucciones desde los mismos automóviles y presencia de elementos que inducen a errores de detección. A pesar de ello, el modelo presenta un margen importante de mejora, lo que permitiría alcanzar un nivel de precisión adecuado para su uso real si se optimizan las etapas de recolección y procesamiento de datos.

## **5. Conclusiones**

### **5.1. Sumario**

El sistema desarrollado para la detección de espacios de estacionamiento y reconocimiento de patentes chilenas, basado en modelos YOLOv8 y Tesseract OCR, entregó resultados diferenciados según el tipo de detección realizada. En el caso de los espacios OCUPADO y LIBRE, el modelo alcanzó un F1-score máximo de 0.42 a un umbral de confianza bajo (cercano a 0.2), mientras que la precisión aumentó progresivamente hasta alcanzar el 100% a partir de 0.863 de confianza. La clase LIBRE mostró un mejor desempeño global en comparación con OCUPADO, tanto en F1 como en la curva precisión–recall. Estos resultados indican que el modelo es más confiable al operar con umbrales altos, priorizando predicciones seguras.

En cuanto a la lectura de patentes, el sistema evidenció un rendimiento superior, con una puntuación F1 de 0.67 alcanzada a una confianza de 0.672, y una precisión del 100% a partir de 0.868. La clase PATENTE presentó una curva PR cercana al ideal, con un mAP<sub>0.5</sub> de 0.615, validando la robustez del sistema en entornos controlados. Por otra parte, la clase PATENTE SIN LECTURA reflejó dificultades asociadas a condiciones visuales adversas. En conjunto, los resultados respaldan la viabilidad del sistema para tareas de control y optimización de estacionamientos, especialmente en condiciones donde se garantice buena visibilidad de los elementos a detectar.

### **5.2. Conclusiones**

El sistema desarrollado para la detección de espacios de estacionamiento y reconocimiento de patentes chilenas, basado en modelos YOLOv8 y Tesseract OCR, entregó resultados diferenciados según el tipo de detección realizada. En el caso de los espacios OCUPADO y LIBRE, el modelo alcanzó un F1-score máximo de 0.42 a un umbral de confianza bajo (cercano a 0.2), mientras que la precisión aumentó progresivamente hasta alcanzar el 100% a partir de un umbral de 0.863. La clase LIBRE mostró un mejor desempeño global en comparación con OCUPADO, tanto en F1-score como en la curva precisión–recall. Estos resultados indican que el modelo es más confiable al operar con umbrales altos, priorizando predicciones seguras y minimizando falsos positivos.

En cuanto a la lectura de patentes, el sistema evidenció un rendimiento superior. Se alcanzó una puntuación F1 de 0.67 en un umbral de confianza de 0.672, y una precisión del 100% desde 0.868. La clase PATENTE presentó una curva precisión–recall estable y cercana al comportamiento ideal, con un mAP@0.5 de 0.615, lo que valida su robustez en entornos controlados. Sin embargo, la clase PATENTE SIN LECTURA reflejó mayores dificultades debido a condiciones visuales adversas, como baja calidad de imagen o iluminación deficiente.

Respecto a las curvas ROC, estas permitieron visualizar de forma clara el rendimiento de ambos modelos. Para la detección de patentes, la curva ROC mostró un área bajo la curva (AUC) de 0.69, lo cual indica una buena capacidad de discriminación entre clases. Su forma zigzagueante puede estar relacionada con la cantidad limitada de datos debido a que se utilizaron imágenes sin variantes, solo imágenes reales no utilizadas en el entrenamiento o validación, pero en general, el rendimiento es aceptable. Por otro lado, el modelo de detección de espacios de estacionamiento obtuvo un AUC de 0.53, valor cercano al azar, lo que evidencia que aún no posee una capacidad adecuada de separación entre clases. Esto puede deberse a problemas debido a la calidad de imágenes, pintado de los espacios demarcados, visibilidad de la cámara debido a que a mayor profundidad de la imagen se pierde el contorno de los espacios disponibles o cuando se sobreponen los automóviles se pierde visión.

En términos de viabilidad, la implementación del proyecto es factible desde el punto de vista técnico y económico, ya que los costos estimados de desarrollo e implementación son asumibles para una institución como la Facultad de Ingeniería. Sin embargo, para garantizar su óptimo funcionamiento en un entorno real, se recomienda realizar pruebas piloto en condiciones reales, considerando también la instalación de cámaras en puntos estratégicos para maximizar la cobertura visual.

En conclusión, los resultados actuales muestran que el proyecto tiene potencial para mejorar la gestión de estacionamientos y el control de acceso mediante visión computacional y técnicas de Deep Learning. Si bien el modelo de detección de placas se acerca a un rendimiento óptimo, el de espacios disponibles requiere mejoras antes de su implementación. Con las optimizaciones propuestas, la ejecución del proyecto es recomendable, ya que generará beneficios en eficiencia, reducción de tiempos de búsqueda, control de uso de espacios y disminución de emisiones contaminantes.

### **5.3. Trabajo a futuro**

Como línea de mejora, se propone fortalecer el sistema de detección de espacios de estacionamiento mediante la incorporación de máscaras o zonas de exclusión, que permitan restringir las áreas válidas de análisis. Durante las pruebas, se observaron falsos positivos en la clase LIBRE generados en sectores no habilitados para estacionar (como accesos vehiculares, veredas u otros espacios vacíos), lo cual evidencia la necesidad de guiar al modelo con un mapa de zonas válidas de detección. Esta técnica, ampliamente utilizada en visión por computador, nos permitirá ir filtrando predicciones fuera del área de interés y mejorando así la confiabilidad del sistema en entornos reales.

Además, se sugiere reforzar el entrenamiento del modelo de ocupación, incorporando mayor variedad de escenarios y condiciones ambientales, así como técnicas de aumento de datos centradas en reflejos, sombras y cambios de iluminación. En el caso del sistema de lectura de patentes, una mejora relevante sería entrenar un modelo con mayor cantidad de información y datos para imágenes o videos en situaciones de baja calidad visual o distorsión.

Asimismo, se plantea la posibilidad de integrar algoritmos de seguimiento para visualizar el comportamiento de los conductores dentro de la Facultad. También sería conveniente desarrollar una interfaz móvil que integre ambos modelos, que permita visualizar en tiempo real el estado del estacionamiento y las lecturas de patentes, además de generar reportes automáticos a los usuarios y personal encargado de los distintos espacios.

Finalmente, se recomienda establecer un proceso de evaluación continua en terreno para monitorear el rendimiento del sistema bajo condiciones reales, considerando factores como variabilidad climática, horarios de alta demanda y tipos de vehículos. Esto permitirá ajustar dinámicamente los umbrales de confianza y adaptar el sistema a los desafíos propios de su contexto operativo, asegurando una implementación sostenible y eficaz.

## 6. Bibliografía

- [1] Instituto Nacional de Estadísticas de Chile. (2024, 13 de septiembre). Más vehículos en Chile: Permisos de circulación crecieron 4,6% en el 2023. <https://www.ine.gob.cl/salade-prensa/prensa/general/noticia/2024/09/13/m%C3%A1s-veh%C3%ADculos-enchile-permisos-de-circulaci%C3%B3n-crecieron-4-6-en-el-2023>
- [2] Instituto Nacional de Estadísticas de Chile. (s.f.). Parque de vehículos - Transporte y comunicaciones. <https://regiones.ine.cl/biobio/estadisticas-regionales/economia/transporte-y-comunicaciones/parque-de-vehiculos>
- [3] Diario Concepción. (2024, 24 de noviembre). Plan Más Movilidad: 2025 será un año de inauguraciones, obras y adjudicaciones. <https://www.diarioconcepcion.cl/ciudad/2024/11/24/plan-mas-movilidad-2025-sera-unano-de-inauguraciones-obras-y-adjudicaciones.html>
- [4] Universidad de Chile. (2021). Sector residencial produce más del 90% de la contaminación en Chile. <https://uchile.cl/noticias/190713/sector-residencial-producemas-del-90-de-la-contaminacion-en-chile>
- [5] Autofact Chile. (s.f.). ¿Cómo evitar el congestionamiento vehicular?. <https://checkout.autofact.cl/blog/mi-auto/conduccion/congestionamiento-vehicular>
- [6] Ultralytics. (s.f.). Modelos YOLOv8. <https://docs.ultralytics.com/es/models/yolov8/>
- [7] OMES-VA. (2022). Reconocimiento de matrículas vehiculares con OpenCV y Pytesseract OCR en Python. <https://omes-va.com/reconocimiento-de-matriculavehiculares-opencv-pytesseract-ocr-python/>
- [8] IBM. (s.f.). Reconocimiento óptico de caracteres (OCR). <https://www.ibm.com/eses/think/topics/optical-character-recognition>
- [9] CVAT. (s.f.). <https://www.cvat.ai/>
- [10] IBM. (s.f.). ¿Qué es la inteligencia artificial?. <https://www.ibm.com/mxes/think/topics/artificial-intelligence>

- [11] IBM. (s.f.). Redes neuronales convolucionales.  
<https://www.ibm.com/eses/think/topics/convolutional-neural-networks>
- [12] IBM. (s.f.). Visión por computadora.  
<https://www.ibm.com/eses/think/topics/computer-vision>
- [13] Ultralytics. (s.f.). Introducing Ultralytics YOLOv8.  
<https://www.ultralytics.com/es/blog/introducing-ultralytics-yolov8>
- [14] Ultralytics. (s.f.). Glosario: Matriz de confusión.  
<https://www.ultralytics.com/es/glossary/confusion-matrix>
- [15] Ultralytics. (s.f.). Glosario: Precision.  
<https://www.ultralytics.com/es/glossary/precision>
- [16] Ultralytics. (s.f.). Glosario: Recall.  
<https://www.ultralytics.com/es/glossary/recall>
- [17] Ultralytics. (s.f.). Glosario: F1-Score.  
<https://www.ultralytics.com/es/glossary/f1score>
- [18] García Guerrero, C. D. C. (2022). Diseño de un sistema inteligente para la detección de plazas de estacionamientos libres [Tesis de pregrado, Universidad de Chile]. <https://repositorio.uchile.cl/handle/2250/187054>
- [19] Mariotti, E. N. (2020). Detección e identificación de características de vehículos utilizando algoritmos de machine learning [Tesis de maestría, Universidad Nacional de Cuyo, Instituto Balseiro].
- [20] Vizcaya Cárdenas, R. (2018). Deep learning para la detección de peatones y vehículos [Tesis, Universidad Autónoma del Estado de México].
- [21] Urbiotica. (s.f.). Urbiotica lanza U-Spot Visio.  
<https://urbiotica.com/blog/urbioticalanza-u-spot-visio/>
- [22] Del Pino Castillo, F. J. (2021). Sistema para la detección y el reconocimiento automático de matrículas utilizando técnicas Deep Learning [Tesis de grado, Universidad de Jaén].
- [23] Torres Carmona, D., & Riveros Pinzón, J. C. (2023). Sistema para detección de placas gemeleadas de vehículos en Bogotá mediante inteligencia artificial [Tesis de grado, Pontificia Universidad Javeriana].

- [24] San Martín Mora, A. I. (2024). Sistema de gestión de estacionamientos mediante el uso de herramientas IoT y visión computacional en tiempo real [Tesis de pregrado, Universidad de Concepción]. <https://repositorio.udec.cl/handle/11594/12223>
- [25] Carreras Franco, A. (2018). QBarrier: Control de acceso a parking utilizando aprendizaje profundo [Tesis de grado, Universidad de Extremadura].
- [26] Autofact Chile. (s.f.). ¿Qué dice la ley sobre la placa patente?. <https://www.autofact.cl/blog/mi-auto/ley-transito/placa-patente>
- [27] Biobio Chile. (2024, 30 de agosto). Qué significan los distintos colores de las patentes en Chile. <https://www.biobiochile.cl/noticias/servicios/tomanota/2024/08/30/que-significan-los-distintos-de-colores-de-las-patentes-en-chile-a-quevehiculo-pertenecen.shtml>
- [28] Datacamp. (s.f.). Precision-Recall Curve Tutorial. <https://www.datacamp.com/es/tutorial/precision-recall-curve-tutorial>
- [29] Towards Data Science. (s.f.). A single number metric for evaluating object detection models. <https://towardsdatascience.com/a-single-number-metric-for-evaluating-objectdetection-models-c97f4a98616d/>
- [30] Datacamp. (s.f.). AUC: Area Under the Curve. <https://www.datacamp.com/es/tutorial/auc>
- [31] Ultralytics. (s.f.). Receiver Operating Characteristic (ROC) Curve. <https://www.ultralytics.com/es/glossary/receiver-operating-characteristic-roc-curve>
- [32] Ultralytics. (s.f.). YOLOv8 Usage Examples. <https://docs.ultralytics.com/es/models/yolov8/#yolov8-usage-examples>
- [33] Roboflow. (2023). What is YOLOv8?. <https://blog.roboflow.com/what-is-yolov8/>

# Anexos

## Anexo A

Permisos de circulación otorgados por cada región, para el año 2023.

**Tabla 13:** Permisos de circulación año 2023, datos de INE.

NÚMERO DE VEHÍCULOS EN CIRCULACIÓN MOTORIZADOS Y NO MOTORIZADOS, SEGÚN REGIÓN. AÑO 2023

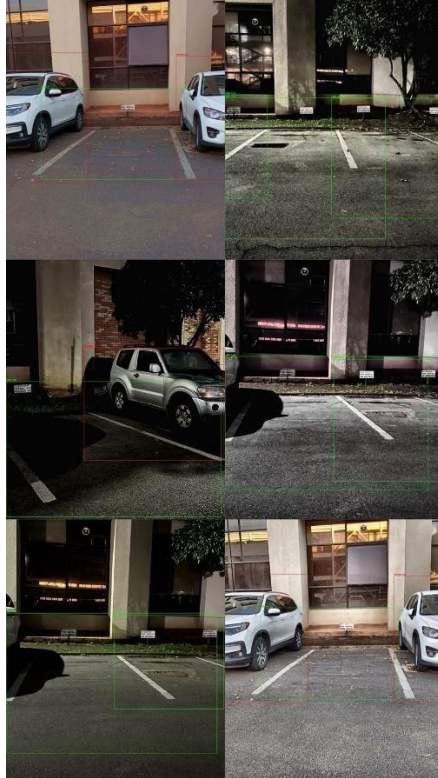
Año	Total	Motorizados	No Motorizados
2019	5.718.409	5.599.733	118.676
2020	5.591.145	5.479.286	111.859
2021	6.102.351	5.980.693	121.658
2022	6.251.852	6.126.980	124.872
2023IP	6.538.727	6.409.428	129.299
<b>Región</b>			
Región de Arica y Parinacota	97.928	96.380	1.548
Región de Tarapacá	162.108	159.225	2.883
Región de Antofagasta	185.128	177.640	7.488
Región de Atacama	110.424	106.984	3.440
Región de Coquimbo	287.854	283.087	4.767
Región de Valparaíso	714.307	700.826	13.481
Región Metropolitana de Santiago	2.400.417	2.367.680	32.737
Región del Libertador General Bernardo O'Higgins	392.451	382.950	9.501
Región del Maule	520.487	507.049	13.438
Región de Ñuble	190.950	186.826	4.124
Región del Biobío	546.483	533.653	12.830
Región de La Araucanía	310.696	304.026	6.670
Región de Los Ríos	145.353	140.653	4.700
Región de Los Lagos	327.245	318.828	8.417
Región de Aysén del General Carlos Ibáñez del Cam	56.158	55.338	820
Región de Magallanes y de la Antártica Chilena	90.738	88.283	2.455

## Anexo B

Detecciones de espacios disponibles u ocupados dentro de la FI, de la Universidad de Concepción.



**Figura 33:** Detección de espacios de estacionamiento.



**Figura 34:** Detección de estacionamiento.

## Anexo C

Código detección de placas patentes y lectura de caracteres.

```
import os import cv2 import gc import re import torch import
numpy as np import pytesseract import matplotlib.pyplot as
plt from glob import glob from tqdm import tqdm from
ultralytics import YOLO from sklearn.metrics import (
accuracy_score, precision_score, recall_score, f1_score,
confusion_matrix, roc_curve, auc, precision_recall_curve
)
```

```

RUTA_BASE = r"C:/Users/Monserrat Pérez/Desktop/patentes_fotos_mt"
CARPETA_SALIDA = os.path.join(RUTA_BASE, "resultados",
"CARPETAIMAGENESMTFINAL")
CARPETA_ROI = os.path.join(CARPETA_SALIDA, "regiones_patente")
CARPETA_CURVAS = os.path.join(CARPETA_SALIDA, "curvas")
for carpeta in [CARPETA_SALIDA, CARPETA_ROI,
CARPETA_CURVAS]:
    os.makedirs(carpeta, exist_ok=True)

pytesseract.pytesseract.tesseract_cmd = r"C:\Program
Files\Tesseract-OCR\tesseract.exe"
def
entrenar_yolov8():
    model = YOLO("yolov8s.yaml")    model.train(
data=os.path.join(RUTA_BASE, "data.yaml"),    epochs=150,
patience=30,    imgsz=640,    batch=4,
workers=0,    device=0,
project=os.path.join(RUTA_BASE, "entrenamiento_patentes"),
name="desde_cero",    exist_ok=True
    )    return os.path.join(RUTA_BASE,
"entrenamiento_patentes",
"desde_cero", "weights", "best.pt")
def
preprocesar_roi(roi):
    gris = cv2.cvtColor(roi, cv2.COLOR_BGR2GRAY)
clahe = cv2.createCLAHE(clipLimit=3.0)    mejorado =
clahe.apply(gris)    mejorado =
cv2.bilateralFilter(mejorado, 9, 75, 75)    mejorado
= cv2.equalizeHist(mejorado)
    _, binarizado = cv2.threshold(mejorado, 0,
255, cv2.THRESH_BINARY + cv2.THRESH_OTSU)
return binarizado
def
formato_valido(texto):
    return (    re.fullmatch(r"[A-
Z]{2}\d{4}", texto) or
re.fullmatch(r"[A-Z]{4}\d{2}", texto) or
re.fullmatch(r"[A-Z]{3}\d{2}", texto)
    )

```



```

def evaluar_modelo(pesos):      modelo =
YOLO(pesos)      imagenes =
sorted(glob(os.path.join(RUTA_BASE,
"test/images/*.jpg")))      y_true,
y_pred, y_score = [], [], []
    for img_path in tqdm(imagenes, desc=" Evaluando"):
        img = cv2.imread(img_path)      nombre =
os.path.basename(img_path)      mensaje, color =
"PATENTE NO ELEGIBLE", (0, 0, 255)      texto_ocr,
es_valida, conf_ocr = "", False, 0.0
        if img.shape[1] <
600:
            img = cv2.resize(img, (640, 512))
            resultado = modelo(img)[0]
if not resultado.bboxes.data.shape[0]:
    y_true.append(0)      y_pred.append(0)
y_score.append(0.0)      cv2.putText(img, mensaje, (20,
40), cv2.FONT_HERSHEY_SIMPLEX, 1, color, 2)
cv2.imwrite(os.path.join(CARPETA_SALIDA, nombre), img)
continue
    for i, box in enumerate(resultado.bboxes.data):
        cls, _, x1, y1, x2, y2 = int(box[5]), float(box[4]),
*box[:4].cpu().numpy()
if cls == 1:
    mensaje = "PATENTE NO ELEGIBLE"
y_true.append(0)
y_pred.append(0)
y_score.append(1.0)      break
if cls == 0:
    roi = img[int(y1):int(y2), int(x1):int(x2)]
if roi.size == 0: break      proc = preprocesar_roi(roi)
texto = pytesseract.image_to_string(proc, config='-
-psm 8 -c
tesseract_char_whitelist=ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789')
texto = re.sub(r"(CARABINEROSDECHILE|CHILE)", "",
texto.upper().strip()      texto = ''.join(c for c in
texto if c.isalnum())      es_valida =
formato_valido(texto)

```

```

        conf_ocr = 1.0 if es_valida else 0.0
        mensaje = texto if es_valida else "PATENTE NO
ELEGIBLE"
        color = (0, 255, 0) if es_valida else
(0, 0, 255)
        texto_ocr = texto
y_true.append(1 if es_valida else 0)
y_pred.append(1 if es_valida else 0)
y_score.append(conf_ocr)
        break
        cv2.putText(img, mensaje, (20,
40), cv2.FONT_HERSHEY_SIMPLEX, 1, color, 2)
if texto_ocr:
        cv2.putText(img, f"Detectado: {texto_ocr}", (20, 75),
cv2.FONT_HERSHEY_SIMPLEX, 0.8, (255, 255, 0), 2)
cv2.imwrite(os.path.join(CARPETA_SALIDA, nombre), img)
        return y_true, y_pred,
y_score

# ===== MAIN ===== if __name__ ==
"__main__":
    torch.cuda.empty_cache()
gc.collect()
    pesos = entrenar_yolov8()
y_true, y_pred, y_score = evaluar_modelo(pesos)
graficar_metricas(y_true, y_pred, y_score)

```

## Anexo D

Código detección de espacios disponibles de estacionamiento.

```

import subprocess
import sys
import importlib
import os
import tempfile
import shutil
import cv2
import torch
import numpy as np
import random
import matplotlib.pyplot as plt
import albumentations as A
import pandas as pd
from glob import glob
from ultralytics import YOLO
import tqdm

```



```

import multiprocessing
from multiprocessing import freeze_support
multiprocessing.set_start_method('spawn', force=True)
BASE = r"C:/Users/Monserrat
Pérez/Desktop/DATASET_ESTACIONAMIENTO_TL"
RESULTADOS = r"C:/Users/Monserrat
Pérez/Desktop/RESULTADOS_ESTACIONAMIENTO_TL"
os.makedirs(RESULTADOS, exist_ok=True) RUTA_YAML
= os.path.join(BASE, "data.yaml")
CLASES = ["OCUPADO", "LIBRE"]
FORMATOS_IMG = [".jpg", ".jpeg", ".png", ".bmp"]
RUTAS = {fase: os.path.join(BASE, fase) for fase in ["train",
"val", "test"]} def
limpiar_dataset(ruta_base):
    img_dir = os.path.join(ruta_base, "images")
    lbl_dir = os.path.join(ruta_base, "labels")
    imgs = {os.path.splitext(f)[0] for f in os.listdir(img_dir) if
os.path.splitext(f)[1].lower() in FORMATOS_IMG}
    lbls = {os.path.splitext(f)[0] for f in os.listdir(lbl_dir) if
f.endswith(".txt")} for nombre in imgs - lbls: for
ext in FORMATOS_IMG:
        path = os.path.join(img_dir, f"{nombre}{ext}")
if os.path.exists(path): os.remove(path) for nombre in
lbls - imgs:
        path = os.path.join(lbl_dir, f"{nombre}.txt")
if os.path.exists(path): os.remove(path) for
archivo in os.listdir(lbl_dir):
    if not archivo.endswith(".txt"): continue
path = os.path.join(lbl_dir, archivo)
with open(path, "r") as f: lines =
f.readlines()
    filtradas = [line for line in lines if not
line.startswith("2 ")] if filtradas:
with open(path, "w") as f:
    f.writelines(filtradas)
else:
    os.remove(path) for ext
in FORMATOS_IMG: img_path =
os.path.join(img_dir,
os.path.splitext(archivo)[0] + ext)
    if os.path.exists(img_path): os.remove(img_path)
def aumentar_imagenes(ruta_base):
    aug = A.Compose([
        A.RandomBrightnessContrast(p=0.7),
        A.HorizontalFlip(p=0.6),
        A.RandomGamma(p=0.5),
        A.MotionBlur(p=0.4),
        A.ImageCompression(quality_lower=75, quality_upper=100,
p=0.5),
        A.CLAHE(p=0.4),

```

```

        A.HueSaturationValue(p=0.3)
    ])
    img_dir = os.path.join(ruta_base, "images")
    lbl_dir = os.path.join(ruta_base, "labels")
    for path in tqdm(glob(os.path.join(img_dir, "*")),
                    desc=f"Aumentando {ruta_base}"):
        img = cv2.imread(path)
        if img is None: continue
        base = os.path.splitext(os.path.basename(path))[0]
        label_path = os.path.join(lbl_dir, f"{base}.txt")
        if not os.path.exists(label_path): continue
        for i in range(5):
            aug_img = aug(image=img)["image"]
            nuevo = f"{base}_aug{i}.jpg"
            cv2.imwrite(os.path.join(img_dir, nuevo),
                        aug_img)
            shutil.copy(label_path,
                        os.path.join(lbl_dir, f"{base}_aug{i}.txt"))
def main():
    print(" Preparando datos...")
    for fase in RUTAS:
        limpiar_dataset(RUTAS[fase])
        aumentar_imagenes(RUTAS[fase]) # Reescribir
        data.yaml con solo 2 clases with open(RUTA_YAML,
        "r", encoding="utf-8") as f:
            lineas = f.readlines()
            with open(RUTA_YAML, "w", encoding="utf-8") as f:
                for linea in lineas:
                    if linea.strip().startswith("names:"):
                        f.write("names:\n 0: OCUPADO\n 1: LIBRE\n")
            break
        f.write(linea)
    print(" Entrenando modelo YOLOv8s...")
    model = YOLO("yolov8s.pt")
    model.train(
        data=RUTA_YAML,
        epochs=150,
        imgsz=640,
        batch=8,
        patience=30,
        workers=0,
        amp=False,
        device=0 if torch.cuda.is_available() else "cpu",
        project="runs",
        name="estacionamientos_TL",
        exist_ok=True
    )
    if __name__ == "__main__":
        freeze_support()
    main()

```