

UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA

Diseño y desarrollo de un sistema de videovigilancia optimizado con inteligencia artificial y plataforma web integrada: Enfocado en Reconocimiento de Patentes

Autor:

Catalina Zelaya Zelaya

Profesor Guía:

Mauricio Araya López

Profesor Correferente:

Marcos Zúñiga Barraza

November 5, 2025



CONSTANCIA DE VALIDACIÓN Y CONFIDENCIALIDAD DE MONOGRAFÍA A REPOSITORIO ACADÉMICO

1.- IDENTIFICACIÓN DEL TRABAJO ACADÉMICO

Tipo de monografía (marcar una opción): Memoria o trabajo de título; Tesis de Postgrado;

Título del trabajo: Diseño y desarrollo de un sistema de videovigilancia optimizado con inteligencia artificial y plataforma web integrada: enfoque en reconocimiento de patentes.

Nombre del candidato(a): Catalina Victoria Zelaya Zelaya

Carrera / Grado: Ingeniería Civil Telemática

Campus: Casa Central ; **Departamento:** Electrónica/Telemática

2.- VALIDACIÓN DEL PROFESOR GUÍA/DIRECTOR DE TESIS

Yo, Mauricio Araya López , en mi calidad de profesor(a) guía/director(a) del trabajo académico mencionado anteriormente **DEJO CONSTANCIA** que:

- He revisado esta versión del documento y corresponde a la versión final aprobada del trabajo.
- El trabajo cumple con los requisitos académicos y de formato establecidos por la institución

3.- EVALUACIÓN DE CONFIDENCIALIDAD POR PROPIEDAD INDUSTRIAL

El trabajo **NO contiene información que amerite confidencialidad** y puede ser publicado de inmediato en repositorio con acceso abierto.


El trabajo **CONTIENE** información con potenciales implicancias de propiedad industrial o intelectual y requiere un periodo de confidencialidad (embargo) por:

6 meses; 12 meses; 2 años; 3 años; 5 años; 10 años


Fundamentación de la necesidad de confidencialidad (obligatorio si se solicita embargo):

4.- FIRMAS

Profesor(a) guía o director(a) de memoria o tesis:

Fecha: _____ ; **Firma:** _____ 

Estudiante o Candidato(a):

Fecha: _____ ; **Firma:** _____ 

Este formulario debe ser insertado como página 2 de la memoria o tesis, completado y firmado por estudiante y profesor(a) antes de la entrega en portal PRISMA de Biblioteca USM.

Agradecimientos

Antes de comenzar, me gustaría agradecer profundamente a mi madre Sonia Zelaya, por siempre estar de mi lado respaldando mis decisiones y mis capacidades. En cada momento de mi vida alentándome a ser mejor con un apoyo incondicional. También quiero agradecer enormemente a mis tíos Andrés Zelaya e Inés García, por su apoyo en cada ámbito de mi vida, y por ser unos segundos padres para mi.

También quiero agradecer a Alexis Alva, Javiera Lorca y Nicholas Bernal, por acompañarme, apoyarme y ayudarme durante mis años de aprendizaje, gracias por estar conmigo en todo momento creyendo en mi. Además quiero agradecer a mi familia extendida, tíos, tías y primos, quienes han confiado siempre en mis capacidades durante mi etapa universitaria.

Es muy importante para mi agradecer también a mi profesor guía, Mauricio Araya, por su apoyo durante la realización de este trabajo, además de las funcionaras de una universidad María Ibacache y Jacqueline Contreras por su apoyo constante durante mis años universitarios.

TABLA DE CONTENIDOS

1	Introducción	4
1.1	Marco General	4
1.2	Problemática a resolver	5
1.3	Acercamiento a la Solución	6
2	Estado del Arte y de la Técnica	7
2.1	Marco Teórico	8
2.1.1	Multilayer Perceptron (MLP)	8
2.1.2	Artificial Neural Network (ANN)	9
2.1.3	Deep Learning (DL)	9
2.1.4	Batch Normalization	10
2.1.5	Convolutional Neural Network (CNN)	10
2.2	Soluciones Existentes	11
2.2.1	PoliScan Surveillance	11
2.2.2	SentiVeillance	11
2.2.3	Parking App	12
3	Propuesta de Solución	13
3.1	Perfil de Usuario	13
3.2	Diseño y arquitectura del sistema propuesto	13
3.2.1	Diagrama de Contexto	13
3.2.2	Diagrama de Arquitectura	14
3.3	Módulos	14
3.4	Requerimientos	15
3.4.1	Interfaz	15
3.4.2	Sistema	15
4	Desarrollo de la Solución	16
4.1	Diagrama de flujo	16
4.2	Detección de la patente	16
4.2.1	Arquitectura	17
4.2.2	Implementación	18
4.3	Reconocimiento de caracteres	20
4.3.1	Arquitectura	20
4.3.2	Entrenamiento	21

4.4	Datos	22
4.4.1	Datos de patentes	22
4.4.2	Datos de caracteres	22
5	Validación	23
6	Conclusiones y Trabajo Futuro	25
6.1	Conclusiones	25
6.2	Trabajo Futuro	25
	Bibliography	26

1 Introducción

1.1 Marco General

Los casos policiales son aquellos delitos conocidos por la policía y que consideran detenciones por flagrancia y denuncias formales realizadas por la ciudadanía en Carabineros de Chile o la Policía de Investigaciones. Normalmente luego de una denuncia de robo en lugar habitado el personal de Carabineros o Policía de Investigaciones busca testigos y/o cámaras de seguridad que podrían ser de ayuda, para finalmente presentar los hechos ante los tribunales de justicia. En la Subsecretaría de la Prevención del Delito [1], se pueden encontrar la tasa de casos policiales de Delitos de Mayor Connotación Social(DMCS), informados por ambas policías durante el año 2022, presenta un crecimiento de 44,6% respecto del año 2021. Los delitos que más aumentan son: robo con violencia o intimidación (63,1%), robo por sorpresa (61,2%), robo en lugar no habitado (56,4%) y robo en lugar habitado (44,6%).

A nivel país, las denuncias realizadas por la población durante el año 2022, presentan un aumento de 46,1%, mientras que, las detenciones se incrementan en un 35,6%, respecto del año 2021. Durante el año 2022, todas las regiones presentan aumentos en la tasa de DMCS, respecto del año 2021. Las regiones que registran el mayor crecimiento son: la Región de Atacama (63,9%), la Región de Arica y Parinacota (59,3%) y la Región de Valparaíso (58,1%).

1.2 Problemática a resolver

En Chile, la seguridad es una problemática que ha ido agravándose cada vez más con el pasar de los años, y el robo en lugar habitado está dentro de los delitos que ha ido en aumento. Hoy en día se siguen buscando métodos contingentes para poder enfrentar este problema, la mayoría de estos son legislaciones que buscan sancionar el mal actuar de algunos individuos, pero no disminuye la inseguridad de las víctimas. Es por esto que se busca una solución con enfoque en la prevención de estos delitos, así de esta forma las personas puedan sentir seguridad dentro de sus hogares.

Este trabajo se sitúa en el contexto de la vigilancia dentro de los condominios. Uno de los problemas que se ha detectado dentro de la vigilancia de estos lugares, es que normalmente las personas están encargada de más de una tarea simultáneamente, dentro de estas la vigilancia de las cámaras de seguridad y el control de acceso al recinto. De hecho, se debe tener en cuenta, que en [2] se estima que el tiempo de atención de adultos sanos varía de 10 a 20 minutos, además en un trabajo de investigación realizado por la Universidad de California en conjunto con Microsoft [3] ha descubierto que cada vez que una persona se distrae le toman más de 20 minutos volver a concentrarse en la tarea que se estaba ejecutando. Esto sugiere que las distracciones y el enfoque en tareas variadas, pueden reducir la efectividad del trabajo realizado por el personal, al afectar la concentración. Ante esta situación, surge la necesidad de una herramienta que optimice el proceso de vigilancia y gestión de acceso, automatizando estos procesos se lograría disminuir las tareas, y por ende las distracciones.

La empresa Jacquard SPA se ha puesto el desafío de devolverles el sentimiento de seguridad a las personas, con esto llegaron a la conclusión de que normalmente es una persona quien se encarga de vigilar las cámaras de seguridad, mientras realizan otras tareas que solo entorpecen la función principal. Decidieron crear un sistema de seguridad al cual llamaron JacSec, que apoye al encargado de la seguridad en el condominio, de manera que este lo alerte de algún evento inusual que podría estar ocurriendo. **La empresa, como resultado de este trabajo, espera un MVP que muestre una interfaz web donde ellos pueden recibir las notificaciones de los distintos eventos anómalos que ocurran en el condominio**, además de que puedan acceder a las cámaras de seguridad donde ocurran estos eventos. Para lograr desarrollar una solución que pueda ser útil en la prevención del delito, como equipo del "Programa de Memorias Multidisciplinarias" (PMM) se busca desarrollar una plataforma web que cuente con aplicaciones modulares que utilicen inteligencia artificial, esto para apoyar en la vigilancia de zonas escogidas en el condominio y en el control de acceso de vehículos al recinto. De esta forma, el personal a cargo de la vigilancia podría ser alertado en el momento por la plataforma, lo cual ayudaría a disminuir las probabilidades de eventos no detectados, debido a las posibles distracciones.

Por otro lado, una de las ventajas que esta solución quiere entregar es que no sea necesario cambiar todo el sistema de seguridad que se encuentra instalado en el recinto, más bien reutilizar este sistema y adaptar el desarrollo de la solución a este. En el desarrollo de esta memoria se presentará específicamente la automatización del registro en las entradas del condominio, por medio de la detección de la patente.

1.3 Acercamiento a la Solución

Para trabajar en la solución del problema se debe reconocer el funcionamiento actual del sistema de videovigilancia que es utilizado en los condominios privados. Este cuenta con cámaras de seguridad conectadas a un sistema central de grabado de video digital (o por sus siglas en inglés DVR) [4], que a su vez se conecta a una pantalla donde se podrán visualizar las imágenes. Al hacer un estudio de mercado se pudo encontrar tecnologías existentes como Mavis, Watsor o Frigate, que podrían ayudar a resolver algunas de nuestras problemáticas, o también se pudieron encontrar productos como las cámaras de vigilancia inteligentes, a un costo accesible. En ellas se ve la principal competencia para el proyecto.

Por esto, uno de los objetivos principales es hacer uso de la instalación que normalmente se encuentra previamente instalada en condominios hoy en día, uniendo a este, dos sistemas independientes de detección optimizados con inteligencia artificial, utilizando dispositivos de borde, como lo es la Jetson Nano, unificados por una plataforma web. Esto con el fin de pasar un flujo de video a través de una secuencia de detectores para poder aumentar la confiabilidad del sistema de seguridad actual, lo cual es algo que haría la diferencia con las plataformas anteriormente nombradas, por el lado de las detecciones, ya que mejoraría la detección de intrusos en la propiedad, además de facilitar el registro del control de acceso de vehículos por medio del reconocimiento de patentes. Por otro lado, la plataforma web que se propone es una versión donde se podrán ver los eventos relevantes de ambos sistemas de detección, el control de acceso con sus respectivos datos, además de la ubicación de la cámara donde se están presentando conflictos y una notificación con una imagen de lo que está ocurriendo.

2 Estado del Arte y de la Técnica

El reconocimiento automático de patentes es una tarea que logra extraer los caracteres de una patente vehicular para poder así hacer la lectura, que al día de hoy es utilizada en gestión de tráfico, vigilancia de la seguridad digital, el reconocimiento de vehículos y gestión de aparcamientos en grandes ciudades. [5] Esta tarea es un problema complejo debido a factores como la calidad de las imágenes, condiciones de iluminación deficientes, ángulo en que la imagen fue tomada y condiciones meteorológicas. Existen variadas formas de abarcar la problemática utilizando aprendizaje automático, tales como el uso de segmentación, procesamiento morfológico con escala de grises o redes neuronales haciendo comparación de caracteres.

Para reconocer los caracteres de la matricula desde la detección del vehículo, se debe centrar el análisis, por una parte en la detección y extracción de esta, y por otra parte en el reconocimiento de los caracteres.

2.1 Marco Teórico

2.1.1 Multilayer Perceptron (MLP)

El Perceptrón es uno de los modelos fundamentales del aprendizaje automático y constituye la base de las redes neuronales artificiales modernas. Propuesto por Frank Rosenblatt en 1958 [6], este modelo representa una de las primeras implementaciones prácticas de una neurona artificial inspirada en el funcionamiento del cerebro humano. Su diseño buscaba simular la forma en que las neuronas biológicas procesan la información, mediante un conjunto de entradas ponderadas que se combinan linealmente a través de pesos sinápticos, para luego pasar por una función de activación que determina la salida final. En su versión más simple, el perceptrón produce una salida binaria, lo que le permite realizar tareas de clasificación lineal. En los experimentos originales, Rosenblatt utilizó el perceptrón para el reconocimiento de patrones visuales simples, como la identificación de letras o figuras geométricas, empleando un dispositivo denominado Mark I Perceptron, que imitaba una “retina artificial” capaz de detectar estímulos visuales. Estos experimentos demostraron el potencial del modelo para aprender a partir de ejemplos, ajustando los pesos mediante un proceso iterativo conocido como regla de aprendizaje del perceptrón.

Sin embargo, a pesar de su relevancia histórica, el perceptrón simple presenta una limitación fundamental: solo puede resolver problemas linealmente separables, es decir, aquellos en los que las clases pueden dividirse mediante una línea recta o un hiperplano en el espacio de características. Este hecho fue demostrado teóricamente por Minsky y Papert (1969)[7], quienes evidenciaron que el perceptrón no podía resolver problemas como el de la compuerta lógica XOR, lo que provocó un estancamiento temporal en la investigación sobre redes neuronales. Para superar estas limitaciones surgió el Perceptrón Multicapa (MLP, por sus siglas en inglés), una extensión que introduce una o más capas ocultas entre la entrada y la salida, junto con funciones de activación no lineales como la sigmoide, ReLU o tanh. Estas características permiten al MLP modelar relaciones no lineales y complejas entre las variables, aumentando significativamente su capacidad de representación. Gracias a esto, el MLP se consolidó como una herramienta poderosa para tareas de clasificación, regresión, reconocimiento de patrones y procesamiento de señales, estableciendo las bases conceptuales y matemáticas sobre las cuales se desarrollan las arquitecturas más avanzadas del aprendizaje profundo (Deep Learning).

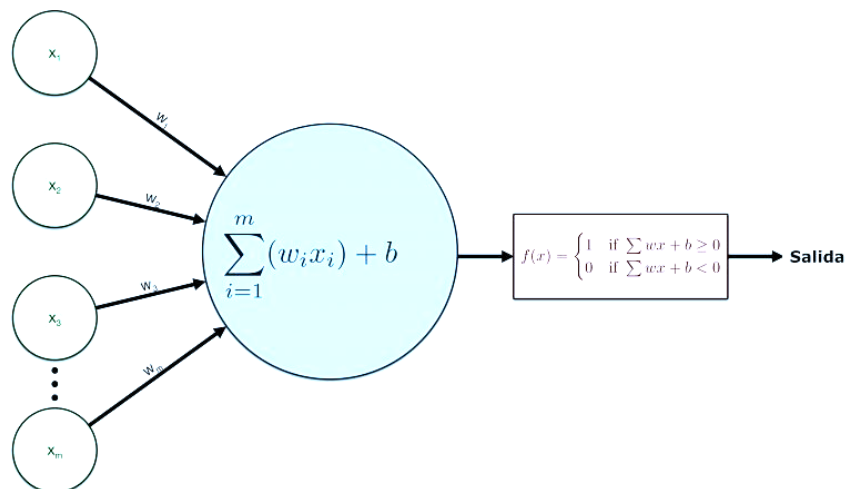


Figure 2.1: Esquema genérico de un perceptrón.

2.1.2 Artificial Neural Network (ANN)

Una Red Neuronal Artificial, o *Artificial Neural Network* (ANN) [8], es un modelo computacional inspirado en la estructura del cerebro humano que se utiliza en el aprendizaje automático. Está compuesta por capas de nodos interconectados (neuronas artificiales) que procesan información de entrada mediante pesos y funciones de activación para generar una salida. Las ANN pueden adaptarse dinámicamente a problemas de alta complejidad. Son capaces de reproducir la interacción dinámica de múltiples factores simultáneamente; también pueden sacar conclusiones sobre una base individual y no como tendencias promedio. Estas herramientas pueden ofrecer ventajas específicas con respecto a las técnicas estadísticas clásicas. La familia de ANN, cuando se selecciona y utiliza adecuadamente, permite maximizar lo que se puede derivar de los datos disponibles.

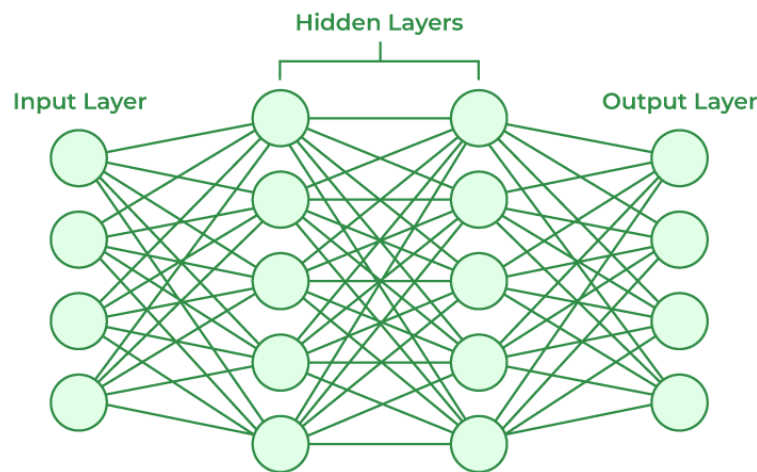


Figure 2.2: Esquema genérico de una red neuronal artificial.

2.1.3 Deep Learning (DL)

El aprendizaje profundo, o *deep learning* [9] es una rama del aprendizaje automático (*machine learning*), a su vez una subdisciplina de la inteligencia artificial, que utiliza redes neuronales artificiales con múltiples capas para identificar patrones complejos en grandes conjuntos de datos, simulando el funcionamiento del cerebro humano. Estos métodos han mejorado drásticamente el estado del arte en el reconocimiento de voz, el reconocimiento visual de objetos, la detección de objetos y muchos otros. El aprendizaje profundo descubre una estructura en grandes conjuntos de datos mediante el uso del algoritmo de retropropagación para indicar cómo una máquina debe cambiar sus parámetros internos que se utilizan para calcular la representación en cada capa a partir de la representación en la capa anterior.

2.1.4 Batch Normalization

El entrenamiento de redes neuronales profundas se complica por el hecho de que la distribución de las entradas de cada capa cambia durante el entrenamiento, a medida que cambian los parámetros de las capas anteriores. Esto ralentiza el entrenamiento al requerir tasas de aprendizaje más bajas y una cuidadosa inicialización de parámetros, y hace que sea notoriamente difícil entrenar modelos con no linealidades saturadas. Nos referimos a este fenómeno como desplazamiento covariable interno y abordamos el problema normalizando las entradas de la capa. La técnica *Batch Normalization* [10] hace que la normalización sea parte de la arquitectura del modelo y realiza la normalización para cada mini-lote de entrenamiento. La normalización por lotes nos permite usar tasas de aprendizaje mucho más altas y ser menos cuidadosos con la inicialización. También actúa como un regularizador, en algunos casos eliminando la necesidad de abandono. Aplicada a un modelo de clasificación de imágenes de última generación, la normalización por lotes logra la misma precisión con 14 veces menos pasos de entrenamiento, y supera al modelo original por un margen significativo.

2.1.5 Convolutional Neural Network (CNN)

Una red neuronal convolucional [11], es un tipo especializado de algoritmo de aprendizaje profundo diseñado principalmente para tareas difíciles que requieren reconocimiento de patrones basado en imágenes, por lo que realiza el reconocimiento de objetos, incluyendo la clasificación, detección y segmentación de imágenes. Las *CNN* se emplean en una variedad de escenarios prácticos, como vehículos autónomos, sistemas de cámaras de seguridad y otros. Estas se distinguen de otras redes neuronales por su rendimiento superior con entradas de señal de imagen, voz o audio. Con cada capa, se aumenta su complejidad, identificando grandes porciones de la imagen. Las capas anteriores se centran en características simples, como colores y bordes. A medida que los datos de la imagen progresan a través de las capas, la red comienza a reconocer elementos o formas más grandes del objeto hasta que finalmente identifica el objeto previsto. Las capas convolucionales otorgan a las *CNN* sus características invariantes de traducción, lo que les permite identificar y extraer patrones y características de los datos, independientemente de las variaciones en la posición, la orientación, la escala o la traducción.

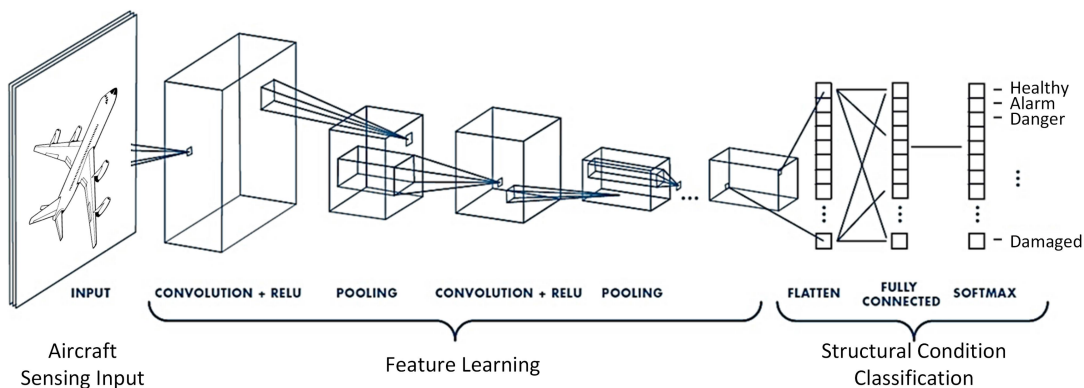


Figure 2.3: Ilustración de una red neuronal convolucional.

2.2 Soluciones Existentes

2.2.1 PoliScan Surveillance

PoliScan Surveillance es un sistema privado de la empresa Vitronic para la detección automatizada de matrículas, puede capturar y procesar matrículas con costos mínimos de personal. Esta solución se enfoca principalmente en agencias estatales, como policías o autoridades, para identificar vehículos en ambas direcciones de un camino sin intervenir directamente en el flujo de tráfico como se muestra en la fig 2.4. Además de no ser una solución disponible en Chile, ésta también está diseñada para escenarios de mucho flujo de vehículos, como podría ser un camino o carretera.



Figure 2.4: Ejemplo de funcionamiento de PoliScan Surveillance.

2.2.2 SentiVeillance

SentiVeillance es un *software* desarrollado por Neurotechnology como un sistema de seguridad completo enfocado en el control de acceso, desde el reconocimiento del personal con identificación biométrica, y hasta el reconocimiento de patentes internas. Ésta solución cuenta con una interfaz gráfica que se puede ver en la Fig 2.4, que además de mostrarnos lo que ocurre en las cámaras, también realiza un resumen de datos obtenidos. Es importante tener en cuenta SentiVeillance está disponible una demostración para su descarga, pero tiene requerimientos mínimos de *hardware* y *software*, y luego de la demostración ésta ofrece una suscripción.

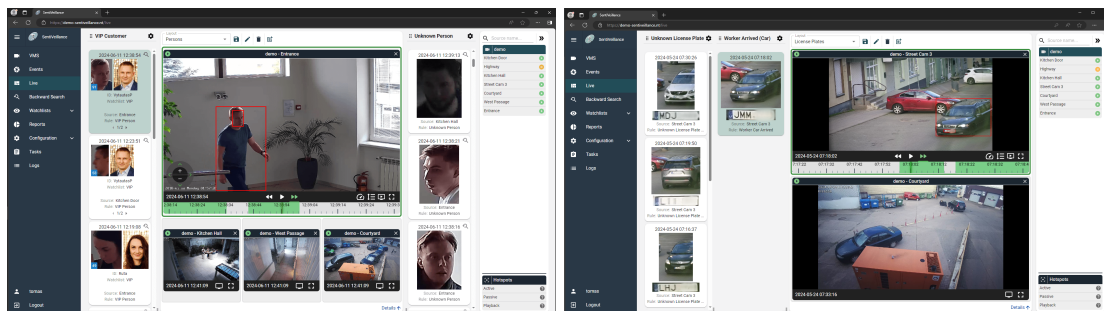


Figure 2.5: Interfaz de SentiVeillance.

2.2.3 Parking App

Parking App es un sistema de control de acceso completo que cuenta con cámaras de vigilancia, totems y aplicación para registrar el tiempo que se permanece dentro del estacionamiento, también cuenta con un sistema de identificación con reconocimiento facial para acceso de personal. Ésta solución se encuentra disponible para cualquier región, pero requiere de una instalación, además de un cobro por distintas suscripciones.

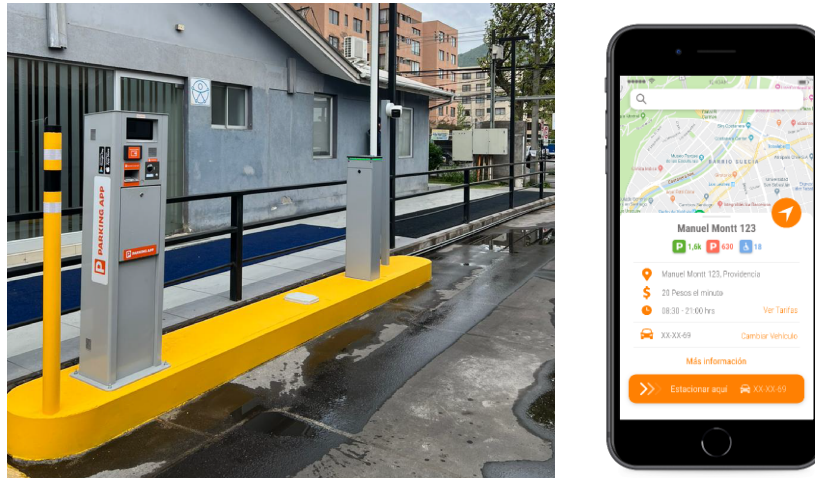


Figure 2.6: Funcionamiento de Parking App.

3 Propuesta de Solución

El proyecto JacSec creado por la empresa Jacquard, es desarrollado en contexto del programa de "Memorias Multidisciplinarias" de la Universidad Técnica Santa María, donde se presenta una idea para implementar en un periodo de un año. Como propuesta de solución se tomaron en cuenta los siguientes puntos.

3.1 Perfil de Usuario

Una de las preocupaciones de la empresa al presentar el problema fue la necesidad de tener una solución para distintos tipos usuarios. Por esto, para entregar una solución adecuada se debe identificar a los usuario que harán uso de esta. Como lo son:

- **Encargado de Seguridad del Condominio:** Persona mayor de edad, no excluyente un grupo social, económico o cultural, calificada y capacitada como guardia de seguridad. Este tipo de usuario busca ser más eficiente en su trabajo sin agregar carga innecesaria en tareas que puedan afectar su atención. Se debe tener en cuenta que el usuario no necesariamente cuenta con conocimientos computacionales.
- **Administrador del Condominio:** Persona mayor de edad, no excluyente un grupo social, económico o cultural, calificada y capacitada como administrador de condominios. Este tipo de usuario busca mantener la seguridad de los habitantes del condominio. Se debe tener en cuenta que el usuario busca representar las necesidades los habitantes, en términos de presupuesto e instalaciones.

3.2 Diseño y arquitectura del sistema propuesto

3.2.1 Diagrama de Contexto

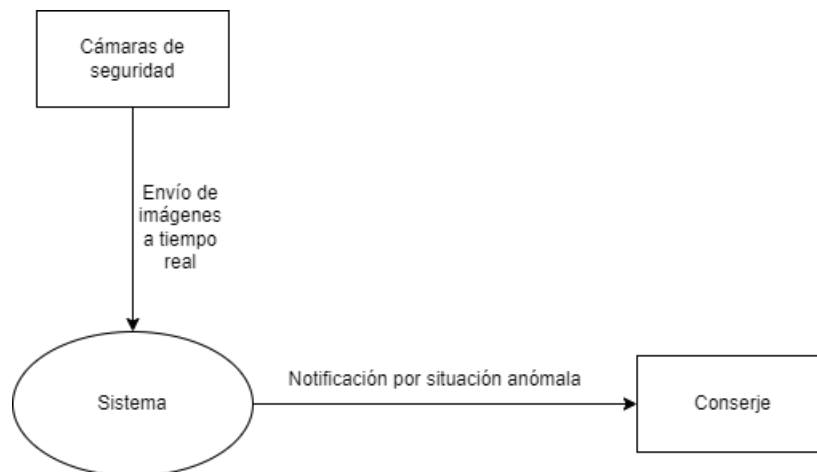


Figure 3.1: Diagrama de Contexto

3.2.2 Diagrama de Arquitectura

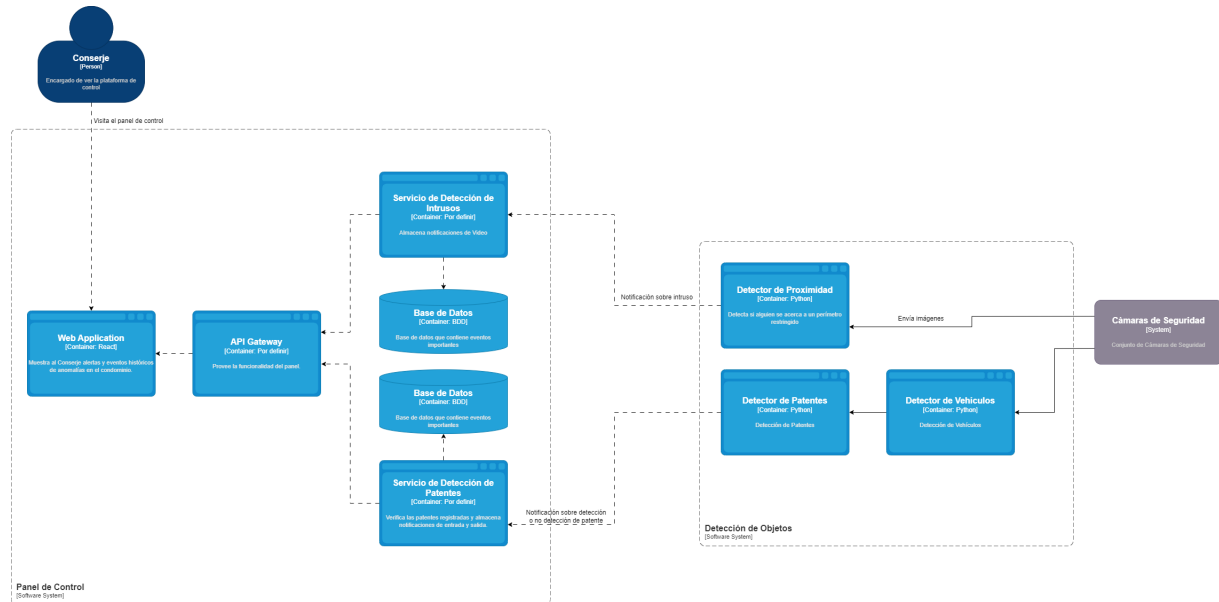


Figure 3.2: Diagrama de Arquitectura

3.3 Módulos

Para la implementación de la solución se generaron los siguientes módulos:

- **Módulo de detección de objetos (M1):** Programa que se encargará de la detección de distintos tipos de objetos tanto con el uso de la GPU como de la CPU. Este módulo es una pieza fundamental dentro del sistema, ya que será el que se encargue de alimentar con información a los módulos siguientes para su posterior procesamiento. Particularmente, detectará personas y patentes, pero se podría extender a otros objetos. Por otro lado, como este módulo será uno de los primeros en interactuar con las imágenes de las cámaras, además tendrá que encargarse de detectar el mal funcionamiento de estas.
- **Módulo de manejo de frames de las cámaras (M2):** Programa que se encargará del manejo de los frames provenientes del módulo de detección. Se encargará de mostrar todas las imágenes procesadas en una sola interfaz gráfica. Este módulo es una pieza fundamental dentro del sistema, ya que será el que se encargue de mostrar todas las cámaras conectadas en una sola interfaz gráfica.
- **Módulo de detección en zonas prohibidas (M3):** Programa que se encargará de detectar si algún objeto se encuentra dentro de la proximidad de un muro perimetral. Este módulo procesará las personas detectadas por el módulo detección de objetos en las áreas designadas como prohibidas, para luego realizar el correspondiente procesamiento y ver si la persona está dentro de esta área, para luego emitir la alerta correspondiente.

- **Módulo de detección de patentes (M4):** Programa que se encarga del reconocimiento de los dos formatos de patentes vehiculares que se encuentran circulando actualmente en Chile, para hacer el respectivo análisis y verificación del acceso. Este módulo procesará la imagen de la patente detectada, para luego analizar el contenido de esta y verificar si tiene acceso o no.
- **Módulo de Panel de Control (M5):** Plataforma web de control web para notificar al conserje la detección de anomalías en el condominio. Este módulo está pensado y diseñado para poder conectar al usuario con la aplicación mediante una interfaz gráfica intuitiva donde pueda ser notificado en caso de existir algún imprevisto.

3.4 Requerimientos

3.4.1 Interfaz

Evento	Descripción	Iniciador	Parámetros	Respuesta
Detección de una persona	Una de las cámaras de seguridad detecta a una persona en una zona restringida	Intruso en zona restringida	Imagen de video	Notificación de alerta de intruso
Detección de patente	La cámara de seguridad de la entrada del condominio detecta la llegada de un auto	Auto en la entrada del condominio	Imagen	Notificación de entrada de vehículo

Table 3.1: Eventos Externos

Respuesta	Descripción	Parámetros
Notificación de alerta de intruso	Alerta de detección de un intruso	Texto/imagen
Notificación de entrada de vehículo	Notificación de entrada de un auto	Texto con la patente del auto que va a ingresar

Table 3.2: Respuestas del Sistema

3.4.2 Sistema

Uno de los requerimientos más importantes para el sistema es la reutilización de las cámaras de seguridad dentro del condominio, por lo que se debe considerar el uso de un DVR [4] para la visualización de todas las cámaras de forma simultánea. Se debe evaluar el uso de un sistema embebido como lo es la Jetson Nano, para facilitar la modularidad del proyecto, además de ser una alternativa mucho más económica de desarrollo.

4 Desarrollo de la Solución

Para efectos de esta memoria el desarrollo de la solución se centra en el trabajo realizado en el **Módulo de detección de patentes (M4)** presentado en el capítulo anterior 3.3.

4.1 Diagrama de flujo

En términos generales la solución se ejecutará siguiendo el flujo presentado en la fig 4.1, tomando como entrada la imagen de un vehículo, detectado por el **Módulo de detección de objetos (M1)** presentado en 3.3, la imagen será procesada por el modelo encargado de la detección del área de la patente entregando una imagen de esta, para luego ser procesada por el modelo encargado del reconocimiento de los caracteres. Finalmente se obtiene como salida una cadena de caracteres que representa la patente del vehículo.

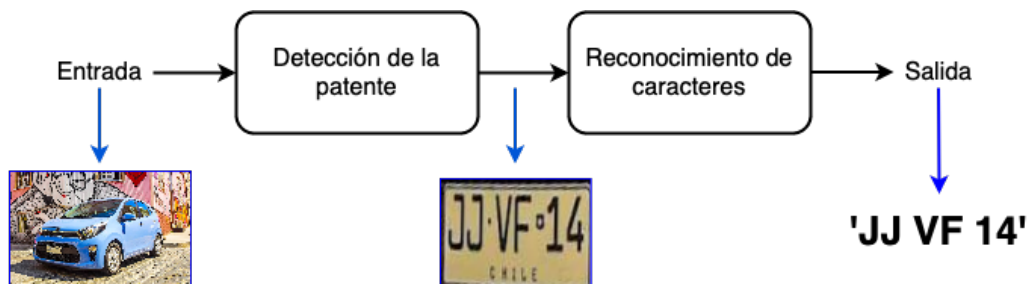


Figure 4.1: Detección y extracción de la placa

4.2 Detección de la patente

Para lograr obtener los caracteres de la patente, lo primero que se debe hacer es conseguir detectar la patente, dentro de la imagen detectada del vehículo. Para esto se utilizó el modelo Improved Warped Planar Object Detection Network (IWPOD-Net) [12], modelo centrado en la detección de objetos planos deformados. Este busca la matrícula y aplica una transformación afín por detección, lo que permite la rectificación del área de la matrícula en un rectángulo que se asemeja a una vista frontal, sin necesidad de adaptarlo para un escenario específico. Cabe destacar que esta no es el último modelo desarrollado en esta línea, ya que existe otra versión como en [13] donde se presentan cambios en la arquitectura agregando a ésta un filtro sobel, lo que modificaría la salida de la red, el trabajo mencionado anteriormente no posee el modelo ni los pesos de manera pública, lo que implicaría la implementación desde el documento, por lo que la utilización del modelo es inviable para este caso, debido a las restricciones de tiempo que posee el desarrollo del proyecto.

4.2.1 Arquitectura

La arquitectura del modelo IWPOD-Net se muestra en la fig 4.2, en esta se pueden ver capas convolucionales con *batch normalization* [10], de donde viene el término "CONVBATCH" presente las capas de la arquitectura. Además de eso, se pueden ver capas maxpooling y resblock, donde esta última está conformada por capas CONVBATCH con activación ReLU y con activación Lineal. Una vez procesada la información de la imagen se obtienen de las capas especializadas, las probabilidades de encontrar una patente ("LP PROBABILITY") y los parámetros de localización ("LOCALIZATION PARAMETERS").

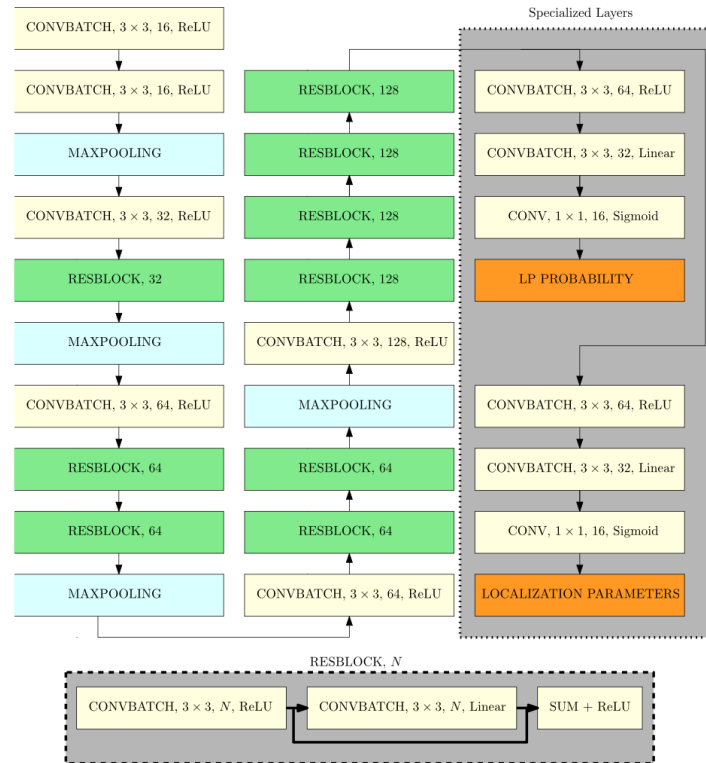


Figure 4.2: Arquitectura detallada del modelo IWPOD-Net que proviene de [12]

La salida de la red entrega siete valores, donde el primer valor indica la probabilidad de encontrar un objeto en la celda correspondiente, y los seis valores siguientes se relacionan con una transformada afín utilizada para la localización de la patente. Por esto es que la función de pérdida debe involucrar los parámetros relacionados tanto con la clasificación como con las tareas de localización. Para seleccionar las celdas (m, n) , la capa de salida debe detectar un objeto. Primero se redimensionan las esquinas p_i de la patente anotada a la resolución de salida, luego se encoge el cuadrilátero redimensionado (con respecto al centroide) y se seleccionan todas las celdas (m, n) dentro de esta región como etiquetas positivas para la patente. Se puede ver un ejemplo de tales celdas, de color blanco, en la fig 4.3. Para la pérdida de localización, la idea principal es encontrar una transformación afín que mapee un cuadrado canónico a las esquinas de la patente anotada. Para esto, se deja q_1, q_2, q_3 y q_4 como vértices correspondientes de un cuadrado unitario canónico centrado en el origen.

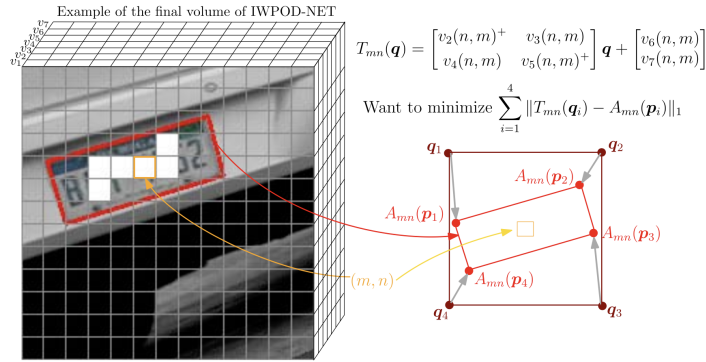


Figure 4.3: Ilustración de la transformada afín, utilizada para localizar la patente en el volumen final de IWPOD-Net obtenido de [12]

4.2.2 Implementación

En la figura 4.4, se encuentran los ejemplos procesados por IWPODNet. Dentro de estas imágenes hay distintos tipos de autos y motocicletas, estas fueron elegidas para poder probar el modelo en distintos contextos.



Figure 4.4: Imágenes de vehículos utilizadas para probar el modelo IWPODNet.

Se puede apreciar en la figura 4.5 el resultado de la implementación del modelo de detección de patente, este nos entrega las patentes de todos los vehículos mostrados anteriormente. Un punto a destacar es que el IWPODNet fue capaz de extraer y rectificar las patentes sin importar el tamaño ni los colores de estas. Esto es muy importante ya que esto demuestra que el modelo será capaz de generalizar bajo diversas condiciones.



Figure 4.5: Resultados de las patentes obtenidas utilizando el modelo IWPODNet.

Se puede ver en la figura 4.6 que el modelo reconoce la forma de la patente, a pesar de que en la imagen del auto a la izquierda de la figura se pueden ver partes que podrían ser interpretados como una patente, esto debido a los números y/o caracteres. Además de esto se aprecia como es capaz de rectificar la patente, mostrando una imagen con perspectiva frontal, mientras la imagen original se encuentra en un ángulo distinto.



Figure 4.6: Ejemplo procesado por la red IWPODNet, donde se muestra que el modelo es capaz de obtener la patente, a pesar del ángulo en el que ésta se encuentra y de las partes que también podrían ser consideradas patentes.

4.3 Reconocimiento de caracteres

Una vez que ya se obtuvo el área de la patente en el vehículo detectado, debemos hacer el reconocimiento de los caracteres en la placa, para esto se evaluaron modelos ligeros aptos para ser utilizados en sistemas embebidos. MobileNet [14], es una red neuronal convolucional ligera para la detección de objetos, en este caso será entrenada con números y caracteres del alfabeto español. Se debe tener en cuenta que previo al uso del modelo, la imagen será pre-procesada para segmentar los caracteres utilizando OpenCV [15], por lo que el modelo solo se encarga de clasificar los caracteres. Cabe destacar que existen otras opciones para realizar esta tarea, como es el caso del modelo presentado en [16] para la detección específica de patentes, aunque el trabajo mencionado se acerca más al contexto del proyecto, se entrenaron ambos utilizando los mismos recursos, y los primeros resultados de MobileNet fueron notablemente mejores.

4.3.1 Arquitectura

La arquitectura del modelo MobileNet en la tabla 1 de fig 4.7, se basa en convoluciones separables en profundidad, excepto para la primera capa, que es una convolución completa. Al definir la red en términos tan simples, podemos explorar fácilmente las topologías de la red para encontrar una buena red. Se puede ver como todas las capas son seguidas por una *batchnormalization* [10] y no linealidad ReLU con la excepción de la capa final completamente conectada que no tiene no linealidad y se alimenta a una capa softmax para su clasificación.

Table 1. MobileNet Body Architecture

Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
5× Conv dw / s1	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$
FC / s1	1024×1000	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

Figure 4.7: Arquitectura del modelo MobileNet que proviene de [14].

4.3.2 Entrenamiento

Antes del entrenamiento es importante llevar a cabo técnicas de pre-procesamiento de los datos. Para comenzar se modificó el tamaño de la imagen de entrada para MobileNet, ya que esta, que originalmente es de 224×224 se redujo a 80×80 , esto debido a las restricciones de cómputo que se poseen. Luego se convierten las etiquetas de las imágenes de matriz 1D en etiquetas *one-hot encoding*, como requerimiento del modelo. Posteriormente, se divide el conjunto de datos en un subconjunto de entrenamiento (90% del conjunto total) y un subconjunto de validación (10% del conjunto total), esto para evitar el sobreajuste y monitorear la precisión del modelo. Finalmente se utilizan técnicas básicas de *"data augmentation"*, cuidando no usarlas en exceso para afectar los resultados, como podría ocurrir haciendo una rotación vertical a "6", que resultaría en "9".

A continuación, se construye una arquitectura MobileNet con pesos pre-entrenados en el conjunto de datos Imagenet [17], se reemplazan las últimas capas de salida para adaptarlas al contexto, así la capa de salida conformaría de 36 neuronas, asociados a 36 caracteres. Mientras que la capa de entrada se modifica para coincidir con las dimensiones de la imagen de entrada modificadas en el pre-procesamiento de los datos ($80 \times 80 \times 3$).

Finalmente, en la figura 4.8 se puede apreciar por un lado, una gráfica de *Accuracy* con respecto a las épocas de entrenamiento, donde se puede ver como tan solo en la quinta época el entrenamiento alcanza un valor de 0,95. Por otro lado, una gráfica de pérdida con respecto a las épocas de entrenamiento, donde se puede ver como por la décima época el error disminuye a 0,09.

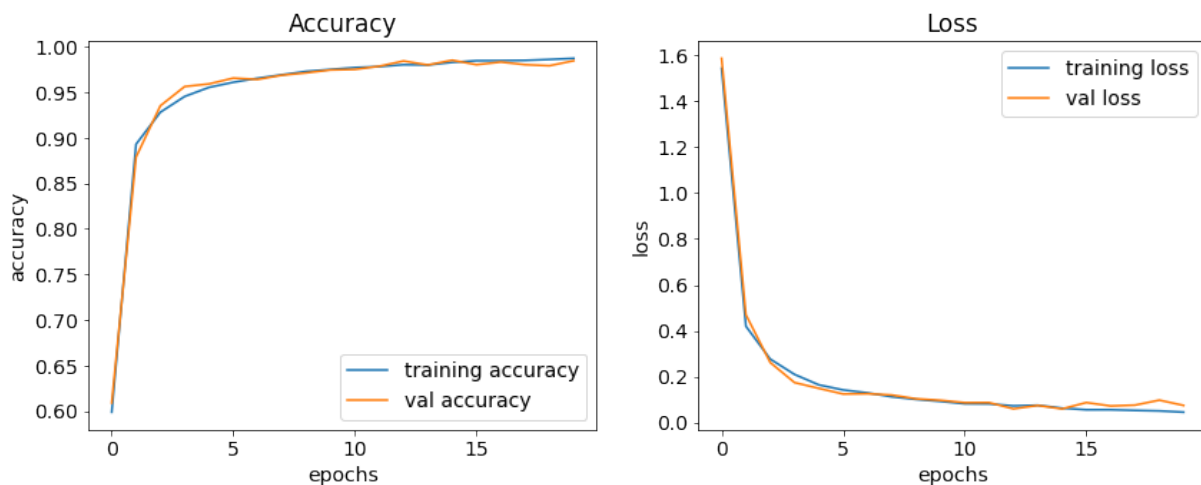


Figure 4.8: Gráficas de resultados de entrenamiento de MobileNet.

Cabe destacar que se establecieron 30 épocas en total, pero debido a un *"EarlyStopping"* el entrenamiento se detuvo en la época 20, a consecuencia de que la pérdida en la validación no disminuía.

4.4 Datos

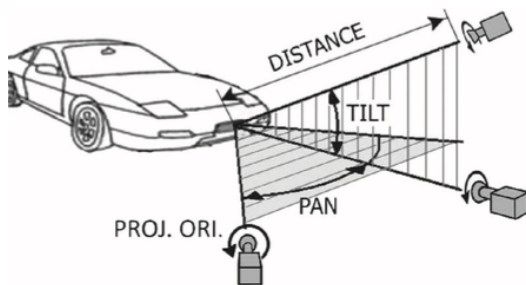
4.4.1 Datos de patentes

El conjunto de datos *Application-Oriented License Plate Recognition* (AOLP) [18] contiene 2049 imágenes, con diferentes ubicaciones, tiempo, tráfico y condiciones ambientales. Todo el conjunto de datos se clasifica en tres subconjuntos, y cada uno de estos nos ofrece un buen alcance de muestras para representar las tres aplicaciones principales, es decir, control de acceso (AC), aplicación de la ley (LE) y patrullaje en carretera (RP).

En **Control de Acceso** refiere casos en los que un vehículo atraviesa un pasaje a una velocidad reducida o se encuentra parado completamente, como en una estación de peaje o una entrada/salida de una región. En **Aplicación de la Ley** refiere a los casos en los que el vehículo viaja a una velocidad regular o alta pero rompe las leyes de tráfico, alguna señal de tráfico o un límite de velocidad, y es capturado en el momento por una cámara de carretera. En **Patrullaje en Carretera** refiere a casos en donde la cámara está instalada en un vehículo de patrulla que toma imágenes de los vehículos con puntos de vista y distancias arbitrarias.

Para cada subconjunto el dataset entrega una carpeta con las imágenes, otra con los caracteres de la patente y otra con los puntos de localización de la patente.

Las imágenes en el dataset se clasifican por rangos y alcances dependiendo de factores establecidos en [18], debido a que el IWPODNet fue entrenado con estos datos, los ángulos por los cuales se registró el modelo concuerdan con los del conjunto, los cuales se pueden ver en la tabla y el esquema de la figura 4.9.



	AC	LE	RP
Pan	$-30^\circ \sim 30^\circ$	$-40^\circ \sim 40^\circ$	$-60^\circ \sim 60^\circ$
Tilt	$0^\circ \sim 60^\circ$	$20^\circ \sim 70^\circ$	$0^\circ \sim 50^\circ$
Size (in width ratio)	0.20 ~ 0.25	0.10 ~ 0.20	0.10 ~ 0.40
Ave. Illum. Intensity	60 ~ 130	40 ~ 150	40 ~ 150
Distance	< 5 m	< 15 m	< 15 m
Proj. Orientation	< 10°	< 15°	< 30°
Number of Samples	681	757	611

Figure 4.9: Variables y alcances de variación para las tres aplicaciones principales, generalizadas a partir del dataset AOLP obtenido de [18].

4.4.2 Datos de caracteres

El conjunto de datos utilizado para el entrenamiento de la red MobileNet, contiene 34575 imágenes divididas en 36 clases de caracteres. Estas imágenes incluye diferentes fuentes para 9 números y 27 letras del abecedario. Algunos ejemplos de estas imágenes se encuentran en la 4.10.

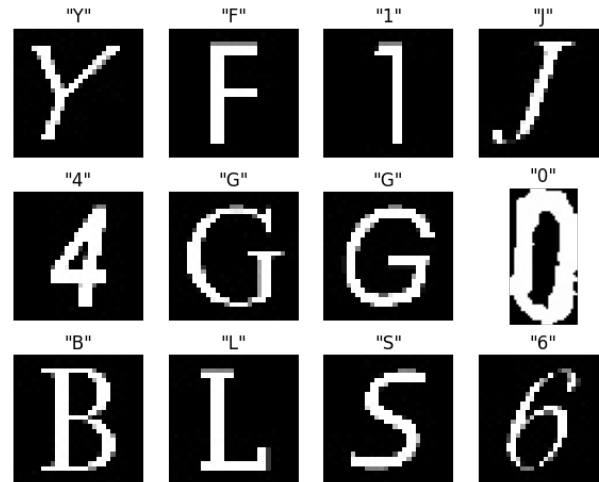


Figure 4.10: Visualización de ejemplos del dataset.

5 Validación

Como se puede apreciar en la figura 5.1 se utilizó la implementación del modelo IWPODNet para la extracción de la patente. Para la validación de los resultados utilizaremos una patente que contenga solo números y letras del abecedario en español, esto debido a es el caso que más se acerca al contexto de este proyecto.



Figure 5.1: Extracción de la patente en un ejemplo, utilizando el modelo IWPODNet.

Una vez se obtuvo la patente, como se muestra en la figura 5.2 se realiza segmentación de la imagen utilizando OpenCV [15] donde se detectan los caracteres, luego separa cada uno de estos para el reconocimiento de la letra o número.



Figure 5.2: Pasos necesarios realizados con OpenCV para identificar y extraer los caracteres de la patente.

Por último, como se puede ver en la figura 5.3, se logra el reconocimiento de los caracteres utilizando el modelo MobileNet, obteniendo así la patente del vehículo.

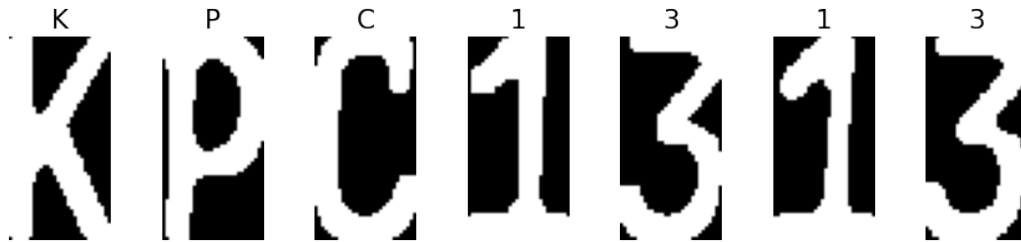


Figure 5.3: Reconocimiento de los caracteres de la patente, mediante el modelo MobileNet.

Se logró el reconocimiento de los caracteres de la patente, por lo que se puede validar la automatización del control de acceso.

Para evaluar un contexto un poco más realista, teniendo en cuenta que las cámaras podrían captar más de una patente, se realizó una prueba utilizando una imagen con dos patentes a la vista que se muestra en la figura 5.4.



Figure 5.4: Imagen de prueba para un caso en el que se deban leer dos patentes.

En la figura 5.5, se muestra como se utilizó el modelo IWPODNet, para extraer las dos patentes presentes en la imagen de la figura 5.4. Al obtener dos imágenes independientes, estas pueden pasar por separado por el proceso explicado anteriormente, tanto la segmentación de caracteres como el reconocimiento de estos utilizando MobileNet.



Figure 5.5: Patentes extraídas de una imagen con dos vehículos con patentes visibles.

6 Conclusiones y Trabajo Futuro

6.1 Conclusiones

Una de las principales motivaciones del desarrollo continuo en el área de la inteligencia artificial es la automatización de tareas y procesos, dado que mejora la eficiencia y la productividad, que mayormente se enfoca en áreas industriales. A pesar de esto, también podemos automatizar pequeñas tareas individuales que implican una gran cantidad de tiempo, como es el caso del trabajo que se ha desarrollado. En este trabajo, se presenta una de las posibles soluciones para la implementación de un sistema de control de acceso vehicular automático. La solución implementada es lo suficientemente ligera como para ser instalada en sistemas embebidos, y a pesar de que finalmente no se utilizaron en este desarrollo, abre paso para que pueda ser utilizado en otro proyecto.

Además se puede concluir que se logró la solución propuesta, por medio de la detección de la patente y el reconocimiento de los caracteres de esta. Sin embargo no fue posible realizar una validación en patentes chilenas, debido a la falta de un conjunto de datos de prueba, aún así se realizaron pruebas en patentes similares que contaran solamente con caracteres del alfabeto español. Con esto se puede concluir que fue posible implementar el Módulo de detección de patentes mencionado en el capítulo 3.3 de este documento.

6.2 Trabajo Futuro

Una de las tareas principales que se debe tomar en cuenta luego del desarrollo de esta solución, es realizar una verificación de la solución para evaluarlo en un contexto real, debido a que el trabajo presentado representa un módulo dentro de un sistema. Por esto es necesario verificar el sistema completo utilizando en conjunto con el DVR, para luego poder hacer pruebas necesarias en contexto real. Para realizar esta verificación del sistema, se buscará retroalimentación tanto de la empresa como de los usuarios finales, lo que nos permitirá considerar este prototipo como una primera versión funcional. Para esto se debería establecer un entorno de desarrollo dedicado para llevar a cabo pruebas de cada módulo del sistema.

Bibliography

- [1] Subsecretaría de prevención del delito.
- [2] Korn J. H. Wilson, K. Attention during lectures: Beyond ten minutes. *teaching of psychology*, 2007.
- [3] Gloria Mark, Shamsi Iqbal, Mary Czerwinski, and Paul Johns. Focused, aroused, but so distractible. pages 903–916, 02 2015.
- [4] Digital video recorder, definition of sciencedirect.
- [5] Elias Ccoto Huallpa, Abel Angel Sullon Macalupu, Jorge Eddy Otazu Luque, and Jorge SÁnchez-Garces. DeterminaciÃdel mejor algoritmo de detecciÃde matrÃculas en ambientes controlados y no controlados. *RISTI - Revista IbÃde Sistemas e Tecnologias de InformaÃ*, pages 83 – 99, 03 2023.
- [6] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65 6:386–408, 1958.
- [7] M. Minsky and S. Papert. *Perceptrons; an Introduction to Computational Geometry*. MIT Press, 1969.
- [8] M. M. Hammad. Artificial neural network and deep learning: Fundamentals and theory, 2024.
- [9] Bengio Y. Hinton G. LeCun, Y. Deep learning. page 436–444, 2015.
- [10] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift, 2015.
- [11] Keiron O’Shea and Ryan Nash. An introduction to convolutional neural networks, 2015.
- [12] Sergio M. Silva and Cláudio Rosito Jung. A flexible approach for automatic license plate recognition in unconstrained scenarios. *IEEE Transactions on Intelligent Transportation Systems*, 23(6):5693–5703, 2022.
- [13] Nguyen Dinh Tra, Nguyen Cong Tri, and Phan Duy Hung. Improving warped planar object detection network for automatic license plate recognition, 2022.
- [14] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications, 2017.
- [15] G. Bradski. The OpenCV Library. *Dr. Dobb’s Journal of Software Tools*, 2000.
- [16] Sergey Zherzdev and Alexey Gruzdev. Lprnet: License plate recognition via deep neural networks, 2018.
- [17] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.
- [18] Gee-Sern Hsu, Jiun-Chang Chen, and Yu-Zu Chung. Application-oriented license plate recognition. *IEEE Transactions on Vehicular Technology*, 62(2):552–561, 2013.