

SISTEMA AUTOMATIZADO DE ESTACIONAMIENTO

Sebastián Alejandro Gregoracci

Juan Pablo Valeo

Este Trabajo Final de carrera fue presentado al Departamento de Electrónica y
Computación de la Facultad de Ingeniería de la Universidad Nacional de
Mar del Plata el día X de Marzo del año 2020, como requisito parcial para
la obtención del título de

Ingeniero en Electrónica

Director: Dra. Leticia María Seijas

Co-Director: Ing. Juan Alberto Etcheverry

El presente trabajo de tesis fue realizado en el Laboratorio de Comunicaciones
del Departamento de Electrónica y Computación, ICyTE, Facultad de
Ingeniería, Universidad Nacional de Mar del Plata.

*COMPLETAR la "DEDICATION" que se encuentra junto con "title." en el
Master.tex .*

Índice general

1. Introducción	1
2. Estructura y funcionamiento del sistema completo	5
2.1. Introducción	5
2.2. Funcionamiento	7
2.2.1. Ingreso	8
2.2.2. Egreso	10
2.3. Relevamiento de estacionamientos	11
2.3.1. Equipamiento disponible	12
2.3.2. Mecanismo de detección de los vehículos	13
2.3.3. Techo y piso	14
2.3.4. Característica de las vías de entrada-salida	16
2.3.5. Disposición del espacio en la zona de ingreso-egreso	17
2.4. Descripción del estacionamiento para ajustes y pruebas	18
3. Primer núcleo: Reconocimiento Automático de Patentes (ALPR)	21
3.1. Introducción	21
3.2. Sistemas de Reconocimiento Automático de Patentes ALPR	22
3.2.1. Estado del arte	22
3.2.2. Etapas generales de los sistemas ALPR	23

3.3.	Descripción de las patentes y los conjuntos de datos	29
3.3.1.	Patentes Argentinas	29
3.3.2.	Conjuntos de datos	35
3.3.2.1.	Set de 20 imágenes de patentes del MERCOSUR	36
3.3.2.2.	Set de 20 imágenes de patentes antiguas	36
3.3.2.3.	Set de 165 imágenes de patentes del MERCOSUR	36
3.3.2.4.	Set de 165 imágenes de patentes antiguas	37
3.3.2.5.	Set de 10 imágenes de motocicletas con patentes del MERCOSUR	37
3.3.2.6.	Set de 10 imágenes de motocicletas con patentes antiguas	37
3.4.	Elección y ajuste del sistema para SAE	38
3.4.1.	Software disponibles	38
3.4.2.	Funcionamiento de los sistemas elegidos	39
3.4.2.1.	OpenALPR	39
3.4.2.2.	OpenCV 3 License Plate Recognition	49
3.4.3.	Experimentación y puesta a punto de los sistemas	54
3.4.3.1.	OpenALPR	54
3.4.3.2.	OpenCV 3 License Plate Recognition	62
3.4.4.	Comparación de los modelos	63
3.5.	Conclusiones	67
4.	Segundo núcleo: Placa I/O	69
4.1.	Componentes principales	69
4.1.1.	Microcontrolador	70
4.1.2.	Módulo de comunicación WiFi	70
4.2.	Fuente de alimentación	72

4.3. Diseño de la etapa lógica y de comunicación	73
4.4. Diseño e implementación del PCB	75
4.5. Periféricos	78
4.5.1. Barreras Infrarrojas	79
4.5.2. Detector de presencia magnético	81
5. Prototipo desarrollado	85
5.1. Introducción	85
5.2. Prototipo inicial	85
5.3. Prototipo final	87
5.3.1. Fase de conexión	89
5.3.2. Envío y recepción de mensajes	92
5.3.3. Vía de ingreso	95
5.3.4. Vía de egreso	100
6. Conclusiones y trabajos futuros	105
6.1. Conclusiones	105
6.2. Trabajos futuros	108
6.2.1. Mejoras	108
6.2.2. Nuevas funcionalidades	109
7. COSAS QUE LUEGO DEBEN SACARSE DEL ESCRITO	111
7.1. Cosas a CORREGIR del Escrito 22-02-2020	111
7.2. Aspectos aprendidos durante el desarrollo del Proyecto Final -no solo poner cosas técnicas	112

Índice de tablas

2.1. Proporción de estacionamientos que poseen y no poseen barrera. .	13
2.2. Característica de la iluminación de los estacionamientos relevados.	13
2.3. Proporción de estacionamientos que poseen y no poseen cámaras de video.	13
2.4. Proporción de los mecanismos de detección utilizados en los esta- cionamientos.	13
2.5. Proporción de estacionamientos techados y al aire libre dentro de la muestra.	15
2.6. Proporción de estacionamientos según el tipo de piso.	15
2.7. Proporción de estacionamientos según el tipo de vías de acceso. .	16
3.1. Resultados de la detección con el entrenamiento original.	55
3.2. Variación de los resultados de los set de 20 imágenes.	60
3.3. Resultados obtenidos para los sets de 165 imágenes.	60
3.4. Resultados obtenidos para el análisis de los sets con imágenes de motocicletas.	61
3.5. Resultados del reconocimiento desarrollado con la configuración por defecto.	62
3.6. Resultados de la detección con el entrenamiento desarrollado. . . .	63

Índice de figuras

2.1. Producto pensado a futuro.	6
2.2. Esquema general de funcionamiento del sistema.	8
2.3. Funcionamiento del sistema en el ingreso.	9
2.4. Funcionamiento del sistema en el egreso.	10
2.5. Proporciones de: (a) los establecimientos que poseen y no poseen barreras, (b) de las características de la iluminación en los mismos y (c) de estacionamientos que utilizan o no cámaras de video. . .	12
2.6. Proporciones de los tipos de mecanismos de detección de vehículos utilizados.	14
2.7. Proporciones de: (a) los establecimientos techados y descubiertos y (b) de los tipos de piso utilizados.	15
2.8. Proporciones de establecimientos que poseen una sola vía de acce- so, dos vías independientes o dos vías de ingreso y dos de egreso. .	17
2.9. Croquis del estacionamiento de prueba.	18
3.1. Etapas generales de un sistema ALPR [10].	24
3.2. Imagen del vehículo que posee la patente a analizar, antes de co- menzar el proceso de reconocimiento.	26
3.3. Imagen del vehículo que posee la patente a analizar, en escala de grises.	26

3.4. Imagen del vehículo que posee la patente a analizar, luego de aplicarle el proceso de binarización.	27
3.5. Proceso de preprocesamiento de la región candidata: imagen original (a), en escala de grises (b) y binarizada (c).	27
3.6. Resultado del proceso de segmentación de caracteres.	28
3.7. Patente reconocida, resultante de la aplicación de un sistema ALPR.	29
3.8. Patente del MERCOSUR: características.	31
3.9. Patente del MERCOSUR: medidas de seguridad.	31
3.10. Patente del MERCOSUR para automóviles y motocicletas.	31
3.11. Patente argentina antigua correspondiente a automóviles.	33
3.12. Patente argentina antigua correspondiente a motovehículos.	34
3.13. Obtención del valor LBP de un pixel analizado [33].	41
3.14. Posibles patentes detectadas.	42
3.15. Detección y análisis de los posibles caracteres.	43
3.16. Detección de bordes horizontales (a) y verticales (b), y detección de patente (c).	44
3.17. Caracteres limpios y segmentados.	44
3.18. Fragmento del proceso de reconocimiento de los caracteres.	46
3.19. Fragmento de la etapa de postprocesamiento junto con el listado de patentes obtenido como resultado del proceso completo.	47
3.20. Resultados parciales de la etapa inicial del software basado en OpenCV.	50
3.21. Proceso de búsqueda de contornos y eliminación de aquellos que no poseen dimensiones de caracteres.	51
3.22. Grupos de caracteres coincidentes.	51
3.23. Proceso desarrollado en la etapa de detección de patente.	52
3.24. Resultado de la remoción de caracteres internos.	53

3.25. Resultado entregado por la fase de post-procesamiento.	54
3.26. Ejemplo de máscara que puede ser seteada.	58
3.27. Patente con palabras internas apreciables.	65
3.28. Reconocimiento de caracteres en patente con palabras internas apreciables.	66
4.1. Microcontrolador Atmega328P utilizado.	70
4.2. Módulo de comunicación WiFi ESP-01 utilizado.	71
4.3. Fuente de alimentación de tensión continua del sistema.	72
4.4. Esquema circuital de la etapa de lógica y comunicación.	73
4.5. Adaptación de niveles de tensión entre los periféricos y el micro- controlador.	74
4.6. PCB generado a partir del diseño final.	77
4.7. Barrera infrarroja utilizada: receptor (izquierda) y transmisor (de- recha).	79
4.8. Conexiones de la barrera infrarroja: receptor (izquierda) y trans- misor (derecha).	80
4.9. Detector magnético utilizado.	81
4.10. Conexiones de la barrera infrarroja: receptor (izquierda) y trans- misor (derecha).	82
5.1. Comparación entre diferentes máquinas de estado [49].	88
5.2. Comunicación entre UCC, Módulo y Placa.	89
5.3. Proceso de establecimiento de la conexión entre UCC, módulo Wi- Fi y Placa I/O.	90
5.4. Control de ACK de la UCC.	93
5.5. Recepción de mensajes desde Placa I/O.	94
5.6. Envío de mensajes desde Placa I/O.	94

5.7. Procedimiento sobre la Vía de ingreso de la UCC.	95
5.8. Procedimiento sobre la Vía de ingreso de la Placa I/O.	96
5.9. Tabla “Regular” de la base de datos desarrollada.	98
5.10. Etapa de egreso de la Placa I/O.	101
5.11. Etapa de egreso de la UCC.	102

Agradecimientos

Al Pipi y Leticia.

Capítulo 1

Introducción

A lo largo de los últimos años, el parque automotor de la República Argentina creció a grandes pasos. Solo en seis años, entre 2011 y 2017, aumentó en un 30 % [1]. Cada vez hay más autos circulando por las calles y los espacios disponibles para estacionar sobre las mismas se están agotando. A pesar de ello, los autos se siguen fabricando y patentando. Es por esto que las playas de estacionamiento cobran fuerza.

Por otra parte, en nuestro país, el mercado inmobiliario es uno de los mejores refugios de capital, y las cocheras resultan una gran alternativa para quienes desean invertir[2].

Debe destacarse, además, la falta de avances tecnológicos (o algún sinónimo de “automatización”) en los establecimientos de la ciudad y también a nivel país. Se hace hincapié en esto debido a que la tendencia actual en el plano internacional es automatizar los espacios de estacionamiento, ya sea en aeropuertos, cocheras, centros de compras, edificios privados, etc [3].

Es por estas razones que se propone desarrollar un Sistema Automatizado de Estacionamiento (SAE) destinado en primera instancia a controlar y gestionar el ingreso, egreso y estadía de vehículos en playas de estacionamiento, pero que a

futuro podrá aplicarse a cocheras de edificios comerciales y privados, control de ingreso y egreso a barrios cerrados, etc.

El proyecto es impulsado por una empresa privada de la ciudad de Mar del Plata quien, en base a una investigación realizada, determinó que los sistemas implementados actualmente en más del 80% de las playas de estacionamiento del país son muy elementales, y requieren de la intervención de un operador para registrar en forma manual el ingreso y egreso de los vehículos. Esto genera una serie de vulnerabilidades al momento de asegurar el control total de las transacciones.

El objetivo de este trabajo es el desarrollo de un proyecto que cumpla con los requerimientos planteados por la empresa dentro del marco de un proyecto final de carrera de la Universidad Nacional de Mar del Plata.

Las condiciones que debe cumplir el prototipo inicial a diseñar y posteriormente desarrollar quedan impuestas por el escenario en el que se lo va a aplicar. Inicialmente se considera un estacionamiento techado, por lo que se estima que la iluminación es siempre la misma (artificial). Además, el establecimiento cuenta con una cabina para el operador, en la cual se encuentra el equipo encargado de recopilar la información enviada desde los distintos componentes del sistema, para la posterior toma de decisiones.

Para la planificación y desarrollo del proyecto consideramos apropiada la aplicación de metodologías ágiles de trabajo [4].

Estas técnicas se basan en un desarrollo iterativo e incremental, con iteraciones cortas y entregas de prototipos al cliente en forma periódica. La gran interacción con el cliente es una de las ventajas de trabajar de esta forma, permitiendo refinar el producto a partir de la realimentación de los mismos. Debido a que el proyecto es de interés para una empresa privada, se propuso implementar este tipo de metodología de trabajo ya que posibilita obtener un prototipo que cumpla en

forma adecuada todos los requerimientos planteados por la misma.

Asimismo, otra ventaja de estas técnicas es la flexibilidad ante el cambio. Es posible realizar modificaciones en diferentes etapas a lo largo del proyecto sin la necesidad de volver a empezar desde el inicio. Esto se debe a que se trabaja sobre núcleos pequeños que, una vez que fueron completados, evolucionan mediante el agregado de nuevas funcionalidades.

Entre las metodologías ágiles existentes, se ha seleccionado para aplicar el método Kanban [4] [5]. Esto se debe a que el mismo, además de las ventajas ya mencionadas, permite una óptima planificación de las tareas a desarrollar y una mejor visualización del trabajo en proceso, pendiente y realizado. Esta técnica favorece una organización mucho más eficiente facilitando el seguimiento detallado de cada etapa del proyecto tanto a nivel individual como grupal y favoreciendo el trabajo en equipo con fuerte interacción entre los distintos actores.

En cuanto a la organización del trabajo se presentan dos núcleos principales: el reconocimiento automático de patentes de vehículos, conocido como ALPR por su nombre en inglés *Automatic License Plates Recognition*, y el diseño y construcción de la placa principal que permitirá administrar a nivel de hardware todos los componentes del sistema.

El capítulo 2 describe la estructura y funcionamiento del sistema completo a partir del cual se desarrollará el prototipo. El capítulo 3 presenta los aspectos vinculados a ALPR desde los sistemas existentes, su estructura general, etapas y características, hasta la evaluación, elección y puesta a punto del sistema a utilizar. El capítulo 4 abarca la descripción de los periféricos a utilizar, los componentes principales de la placa principal del sistema y, el diseño y la construcción de la misma. El capítulo 5. . . (COMPLETAR CON TODOS LOS CAPÍTULOS QUE HAYA).

Capítulo 2

Estructura y funcionamiento del sistema completo

2.1. Introducción

El propósito de este proyecto final es diseñar un prototipo a partir del cual la empresa pueda continuar con su desarrollo y obtener un producto. La versión comercial pensada a futuro en conjunto con la empresa puede observarse en el esquema de la figura [2.1](#).

El objetivo de este capítulo es describir la estructura y funcionamiento del sistema completo a partir del cual se desarrollará el prototipo final.

- Este no es el definitivo, ya que tiene los colores invertidos y líneas del AutoCAD.

internos al estacionamiento con código de barra.

Por último, dentro de la cabina del operador del sistema, se tiene la Unidad Central de Control (UCC), encargada de interactuar con todos los componentes del sistema y procesar los datos obtenidos por los mismos. Entre las tareas que desarrolla se encuentran obtener el número de la patente del vehículo que ingresa o egresa del estacionamiento y de gestionar la base de datos. Además, en la cabina también se cuenta con una impresora de tickets fiscales y un lector de código de barra.

Uno de los componentes fundamentales del sistema es la Placa I/O **CAM-BIAR NOMBRE DE LA PLACA** que concentra todos los periféricos de la zona de ingreso/egreso, menos las cámaras, y que se comunica con la UCC mediante WI-FI.

2.2. Funcionamiento

En la figura 2.2 puede observarse el esquema de funcionamiento del sistema a un nivel general. El mismo se explica más detalladamente a lo largo de las dos siguientes secciones.



Figura 2.2: Esquema general de funcionamiento del sistema.

A continuación, se describen los módulos de ingreso y egreso del sistema.

2.2.1. Ingreso

Como se observa en la figura 2.3, el proceso comienza con la detección de un vehículo que desea ingresar al estacionamiento. La misma se realiza mediante un detector magnético ubicado en el suelo, bajo el pavimento. Este posee una bobina (el lazo) cuya inductancia varía cuando un gran elemento metálico se acerca a ella. Esa variación hace que se modifique la frecuencia de oscilación del oscilador que posee el equipo. Esto produce la activación de un contacto seco que se utiliza para indicarle al sistema que un vehículo intenta ingresar al establecimiento.

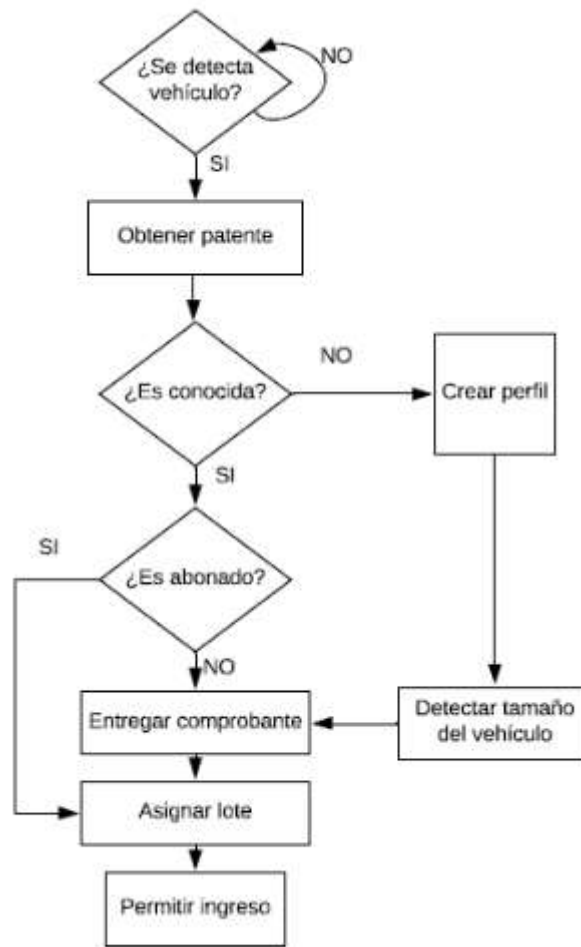


Figura 2.3: Funcionamiento del sistema en el ingreso.

A continuación, se realiza la obtención de la patente mediante un juego de dos cámaras IP que se encuentran filmando continuamente. La misma es buscada en la base de datos y, en caso de no ser encontrada, se crea un perfil para el vehículo. Además, se determina el tamaño del mismo y se lo añade al perfil, con el objetivo de determinar la tarifa a cobrar (automóvil, camioneta o motocicleta).

Luego, se procede a asignarle al conductor, mediante una pantalla, el lote en el que debe estacionarse, la hora de ingreso, la patente y la tarifa. Junto a la pantalla se encuentra una expendedora de tickets internos al estacionamiento, que poseen un código de barra y la misma información mostrada anteriormente.

Los mismos se pueden retirar sin bajarse del vehículo.

En caso de que el usuario sea abonado, no se le entrega el ticket debido a que el sistema reconoce la condición de abonado al detectar la patente.

Luego de que el usuario retire el ticket, se levanta la barrera y se permite el acceso.

2.2.2. Egreso

Al igual que el de entrada, el sistema de egreso comienza con la detección de un vehículo que desee retirarse del establecimiento. El mismo puede observarse en la figura 2.4. Una vez que esto sucede, se procede a obtener la patente del vehículo, con el fin de determinar si la tarifa ya fue pagada.

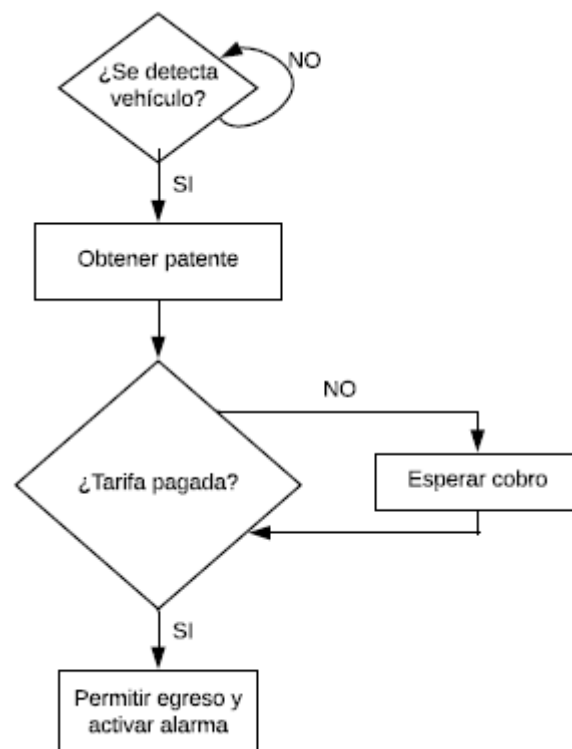


Figura 2.4: Funcionamiento del sistema en el egreso.

Mediante un lector de códigos de barra ubicado en la cabina del operario se reconoce el horario de ingreso a partir del ticket de entrada, se determina el tiempo de uso del servicio y, por lo tanto, el monto a cobrarle al cliente.

Después de realizar el pago, a cada cliente se le entrega un comprobante con validez legal y se le permite el egreso. Sí el sistema comprueba que el cliente aún no realizó el pago, no le permite la salida hasta que esto suceda.

En el caso de los abonados, al detectar la patente el sistema verifica que la tarifa haya sido pagada. De ser así, se permite el egreso.

Se implementa una alarma luminosa y sonora para indicar el egreso de los vehículos.

2.3. Relevamiento de estacionamientos

El diseño del estacionamiento de prueba presentado en la figura 2.1 fue construido en base a los resultados del relevamiento de una muestra (que se consideró representativa) de estacionamientos ubicados en la ciudad de Mar del Plata. Este relevamiento permitió definir qué aspectos considerar en el prototipo para cubrir las necesidades de automatización real y de interés para la empresa solicitante del sistema.

Los establecimientos inspeccionados se encuentran en diferentes zonas de la ciudad, poseen distintas características edilicias, son de diferentes capacidades y disponen de distintos sistemas de atención al cliente y de control de acceso. Además, mientras que en algunos estacionamientos se ingresó y se conversó con el personal a cargo, en otros se realizaron inspecciones visuales desde el exterior de los mismos.

Entre las características que se tuvieron en cuenta al hacer el relevamiento se encuentran:

- El equipamiento con el que contaban: barreras, cámaras e iluminación
- Mecanismo de detección de llegada de los vehículos
- Características edilicias: techo y condición del piso (podría afectar la toma de fotos)
- Si la vía de entrada-salida era una sola o si existían dos independientes
- Disposición del espacio en la zona de ingreso-egreso

2.3.1. Equipamiento disponible

A continuación, se presentan tres tablas junto con tres graficos de torta que muestran los resultados del relevamiento respecto al equipamiento disponible en los dieciséis establecimientos inspeccionados. En la tabla 2.1 (fig. 2.5 a) se observa la proporción de estacionamientos que poseen y no poseen barreras, en la tabla 2.2 (fig. 2.5 b) se muestra cómo es la iluminación de los mismos y, finalmente, en la tabla 2.3 (fig. 2.5 c) se presenta la proporción de establecimientos que cuentan y de los que no cuentan con cámaras de video.

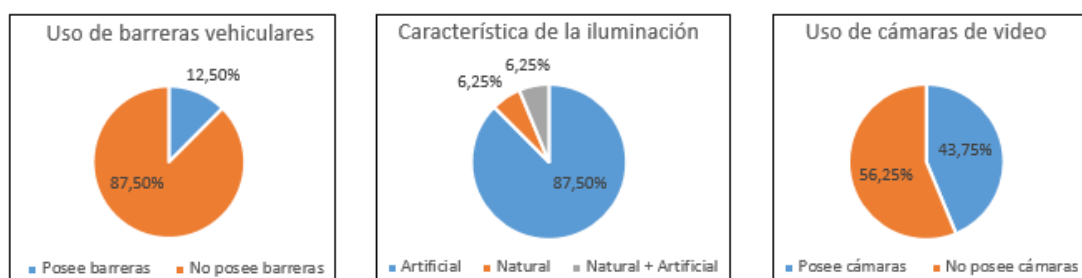


Figura 2.5: Proporciones de: (a) los establecimientos que poseen y no poseen barreras, (b) de las características de la iluminación en los mismos y (c) de estacionamientos que utilizan o no cámaras de video.

Los estacionamientos de la ciudad que actualmente poseen cámaras de video

Barrera	Sí	No
Proporción de estacionamientos	12.5 %	87.5 %

Tabla 2.1: Proporción de estacionamientos que poseen y no poseen barrera.

Iluminación	Artificial	Artificial + Natural	Natural
Proporción de estacionamientos	87.5 %	6.25 %	6.25 %

Tabla 2.2: Característica de la iluminación de los estacionamientos relevados.

Cámaras	Sí	No
Proporción de estacionamientos	43.75 %	56.25 %

Tabla 2.3: Proporción de estacionamientos que poseen y no poseen cámaras de video.

no las utilizan para el mismo propósito que se les da en este proyecto. En la mayoría de ellos son usadas por cuestiones de seguridad, para observar la llegada de los vehículos o para ambas simultáneamente (varias cámaras con diferentes propósitos).

2.3.2. Mecanismo de detección de los vehículos

En esta sección, se presenta la forma en que se realiza la detección del vehículo en los accesos de los estacionamientos relevados. Esto se muestra en la tabla 2.4, que se encuentra acompañada por el gráfico de la figura 2.6.

Mec. Detección	Visualmente (operador)	Visualmente (operador + cámaras)	Detector magnético	Botón
Proporción de estacionamientos	62.5 %	25 %	6.25 %	6.25 %

Tabla 2.4: Proporción de los mecanismos de detección utilizados en los estacionamientos.

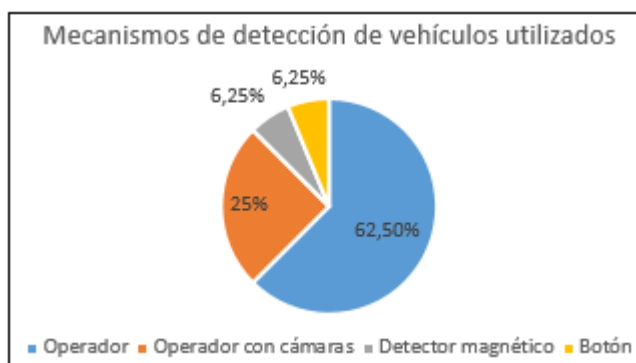


Figura 2.6: Proporciones de los tipos de mecanismos de detección de vehículos utilizados.

La primera categoría hace referencia al caso en el que la detección implica que un operador se encuentra vigilando en forma visual el ingreso al establecimiento constantemente. Debido a esto, en la mayoría de los estacionamientos la cabina del operador es vidriada y se encuentra en una zona cercana a las vías de entrada-salida.

La segunda categoría se refiere a que la persona encargada de la operación del establecimiento cuenta con un sistema de cámaras apuntando hacia las vías de ingreso-egreso. Esto le permite a la misma visualizar cuando un vehículo entra o sale del establecimiento en un monitor dentro de la cabina. En este caso, esta última puede encontrarse alejada de los accesos al estacionamiento.

La última categoría considera el uso de detectores magnéticos en los ingresos. Estos detectan la circulación de un vehículo debido a las variaciones magnéticas que el contenido ferroso de los automóviles les provoca.

2.3.3. Techo y piso

A continuación, se muestra la proporción de estacionamientos techados y al aire libre, junto con información acerca del piso utilizado en los mismos. Esto

puede observarse en las tablas 2.5 y 2.6, que se encuentran acompañadas por los gráficos de la figura 2.7.

Debido a que se plantea un estacionamiento de prueba que sea adaptable a la mayor cantidad de escenarios posible, es importante conocer estas proporciones. Además, es necesario determinar las características del piso que generalmente se utiliza en este tipo de establecimientos. Esto se debe a que hay que asegurar que el sistema no tenga problemas con el mismo al fotografiar las chapas patentes.

Estacionamiento techado	Sí	No
Proporción de estacionamientos	87.5 %	12.5 %

Tabla 2.5: Proporción de estacionamientos techados y al aire libre dentro de la muestra.

Tipo de piso	Cemento liso	Otro
Proporción de estacionamientos	87.5 %	12.5 %

Tabla 2.6: Proporción de estacionamientos según el tipo de piso.

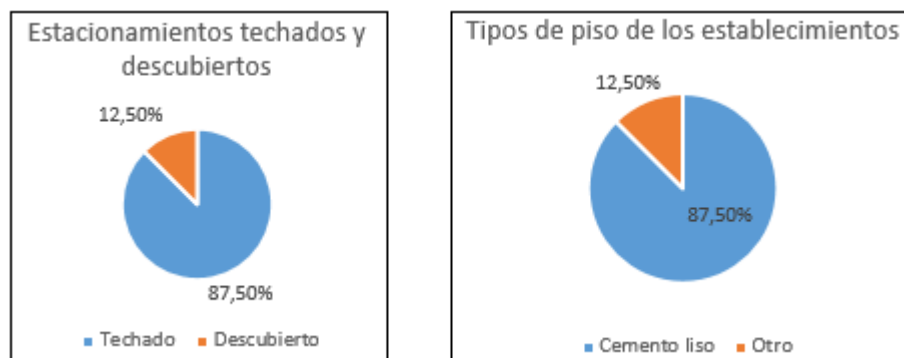


Figura 2.7: Proporciones de: (a) los establecimientos techados y descubiertos y (b) de los tipos de piso utilizados.

Cabe destacar que en los dos establecimientos en el que el piso no es cemento liso se cuenta con un piso de tierra y uno compuesto por baldosas cerámicas blancas. Este último podría afectar a las cámaras debido a que puede reflejar la luz.

2.3.4. Característica de las vías de entrada-salida

En esta sección, se pretende determinar si existe una única vía utilizada para ambos propósitos o si existen dos rutas independientes. Es necesario conocer esta información debido a que el equipamiento del sistema para cada uno de estos casos puede que sea diferente. En la tabla 2.7 se observa el porcentaje de estacionamientos que poseen una única vía de acceso y el de establecimientos que poseen dos independientes. Además, el mismo puede visualizarse en la figura 2.8.

Vías de ingreso-egreso	Única	Dos independientes	Dos entradas y dos salidas
Proporción de estacionamientos	62.5 %	31.25 %	6.25 %

Tabla 2.7: Proporción de estacionamientos según el tipo de vías de acceso.

Existen algunos casos particulares, como es el de los establecimientos de menor nivel de afluencia de vehículos que igualmente poseen 2 vías de acceso. Al ser más pequeños, es más complicado maniobrar en el interior y suelen utilizarse ambas vías en forma indistinta. No hay una definida una ruta para la entrada y una para la salida. Igualmente, se los consideró en la categoría de “Dos independientes”.

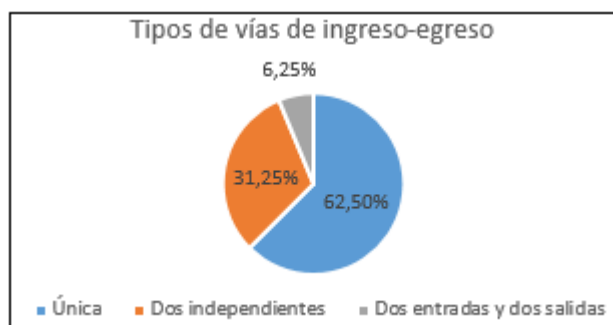


Figura 2.8: Proporciones de establecimientos que poseen una sola vía de acceso, dos vías independientes o dos vías de ingreso y dos de egreso.

2.3.5. Disposición del espacio en la zona de ingreso-egreso

Esta es la característica de los establecimientos en la que mayor diversidad se encuentra. En la mayoría de los casos analizados (11) las vías de ingreso-egreso tienen forma de pasillo, de un largo que permite el acceso de uno o dos vehículos. Sin embargo, mientras que dos de ellos posibilitan la circulación de dos automóviles en paralelo, los nueve restantes se asemejan al garaje de una vivienda y solo permiten la circulación de uno.

Por otra parte, está el caso de aquellos estacionamientos que no tienen un lugar destinado para la atención de los clientes y la misma se realiza sobre el área de ingreso. Entre los analizados se encuentran dos en esta condición. Además, se tienen establecimientos cuya entrada/salida es en forma de rampa, sobre la cual pueden ubicarse los vehículos para no esperar ser atendidos sobre la calle. En el relevamiento realizado solo hay uno correspondiente a este caso.

Finalmente, están aquellos estacionamientos que poseen más de una vía de entrada-salida, por lo que combinan algunas de las características antes mencionadas. Entre los lugares inspeccionados, se tienen dos con esta particularidad.

2.4. Descripción del estacionamiento para ajustes y pruebas

En base a la investigación realizada sobre las características de los establecimientos ubicados en la ciudad de Mar del Plata se decidió simular un estacionamiento en el garaje de una casa que se tenía a disposición. Un croquis del mismo puede observarse en la figura 2.9.

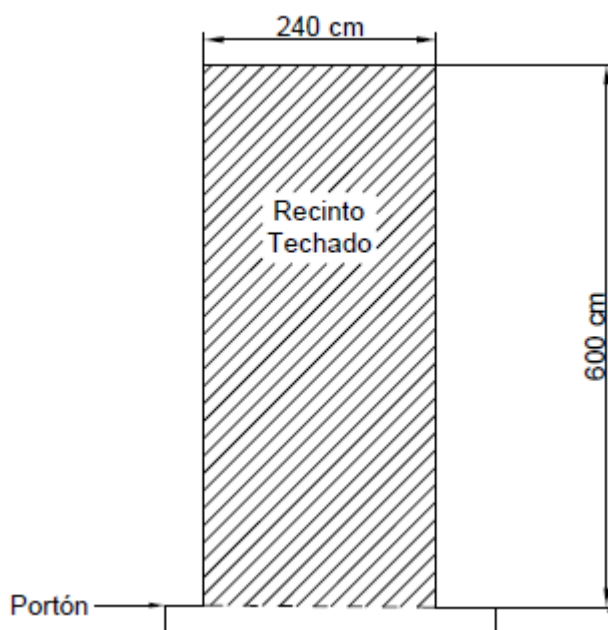


Figura 2.9: Croquis del estacionamiento de prueba.

La elección realizada se fundamenta en que el establecimiento prototipo posee características similares a la de la mayoría de los que fueron inspeccionados. Las mismas se detallan a continuación:

- Tiene forma de pasillo, con espacio para el ingreso de dos vehículos en fila

- Es techado y posee luz artificial
- El piso está compuesto de baldosas negras lisas. Las mismas, al igual que el piso cemento liso no afectan al momento de tomar las fotografías de la chapa patente

COSAS QUE FALTA HACER LUEGO DE COMPRAR LOS EQUIPOS Y

ARMAR EL SISTEMA:

- Ubicacion de las camaras y demas perifericos
- Agregar imagenes?

Capítulo 3

Primer núcleo: Reconocimiento Automático de Patentes (ALPR)

3.1. Introducción

En el desarrollo del prototipo, el reconocimiento de la patente, tanto del vehículo que ingresa como del que egresa, es una de las etapas fundamentales.

El reconocimiento automático de patentes es un problema típico que involucra varias ramas de estudio, principalmente al área de Reconocimiento de Patrones y el campo de la Visión artificial, y que ya ha sido estudiado ampliamente [6] [7] [8] [9].

Una de las principales dificultades que se presentan es que los escenarios pueden ser cambiantes, como podía ocurrir en el caso de un Sistema de Transporte Inteligente (ITS), donde el reconocimiento de patentes permite identificar vehículos en movimiento y obtener varias matriculas simultáneamente. [7] [10].

En nuestro caso, a pesar de que se considera que el sistema puede aplicarse a diversos tipos de estacionamientos, el escenario es más acotado: la cámara se encuentra en una posición fija y el vehículo frenando a baja velocidad. En este

contexto, se tienen en cuenta otras posibles problemáticas, como las variaciones lumínicas (día/noche), la iluminación propia de la placa y la existencia de distintos modelos de placas patentes en nuestro país: argentina antigua (1995 - 2016) y Mercosur (2016- presente). Además, se consideran diferentes tipos de vehículos: automóviles, camionetas y motocicletas.

El uso de un sistema que recoja los datos de la matrícula de un vehículo y lo almacene en una base de datos en formato de texto evita la necesidad de ocupar espacio de memoria debido al almacenamiento de videos o imágenes y facilita la consulta de los datos de cada uno de los clientes del establecimiento en el que se monte el mismo.

El objetivo de este capítulo es presentar todo lo relacionado con el desarrollo del sistema reconocedor de patentes para el Sistema Automatizado de Estacionamientos SAE. Se describen las características generales de los sistemas ALPR, los tipos de software disponibles, el sistema elegido, la parametrización e implementación del mismo, las características de los conjuntos de datos utilizados y las características principales del sistema implementado, como el tratamiento de video.

3.2. Sistemas de Reconocimiento Automático de Patentes ALPR

3.2.1. Estado del arte

La detección de patentes de vehículos de manera automática, puede ser utilizada en varias aplicaciones como el multado de vehículos, el control de accesos y sistemas de seguridad, entre otras. Es por eso que en la actualidad existe una amplia cantidad de empresas dedicadas al reconocimiento de patentes. Entre ellas,

podemos destacar a PIPS TECHNOLOGY, empresa fundada en el Reino Unido que ofrece no solo las aplicaciones previamente mencionadas, sino que además posee un sistema de transporte inteligente, control de congestionamiento, entre otras [11]. Por otra parte, se encuentra la empresa canadiense GENETEC, la cual, al igual que la anterior, ofrece soluciones de ALPR ya mencionadas como los sistemas de seguridad y el control de accesos [12]. Por último, Neural Labs es una empresa radicada en Barcelona, España. Además de las aplicaciones ya mencionadas, la misma utiliza el reconocimiento de patentes para el control de clientes en gasolineras y peajes [13].

Particularmente en Argentina, si bien los sistemas ALPR son utilizados en algunas aplicaciones como el control de flujo o el multado, los mismos no se encuentran ampliamente difundidos, como es el caso en el plano internacional. Dentro de nuestro país, podemos mencionar la compañía VISART, la cual es una de las pocas empresas nacionales que se encuentra trabajando en el área [14]. Además, en Argentina, aun no se han logrado grandes avances acerca del uso de sistemas ALPR en playas de estacionamiento. Por lo tanto, la implementación desarrollada en este proyecto final resulta innovadora para nuestro país.

3.2.2. Etapas generales de los sistemas ALPR

El proceso realizado por todo sistema ALPR involucra una serie de etapas generales que pueden ser llevadas a cabo a partir de diferentes metodologías [6] [10] [15]. Las mismas son las que se muestran en la figura 3.1, y se las describe a continuación:

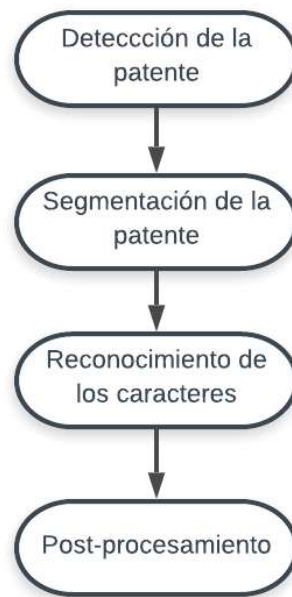


Figura 3.1: Etapas generales de un sistema ALPR [10].

Etapas 1: Detección de la patente

Esta etapa depende en gran medida de ciertas características de la placa tales como forma, color, dimensiones, etc. Además, esta fase es afectada por las condiciones del entorno: iluminación, calidad de la cámara con la que se captura la imagen o video, las características del ambiente, etc. Normalmente, incluye una etapa de pre-procesamiento que prepara la imagen para el proceso de reconocimiento.

El pre procesamiento consiste en la aplicación de diferentes procedimientos a la imagen, los cuales pueden variar dependiendo el algoritmo de detección que se utilice. Entre estos podemos destacar:

- **Transformación en una imagen en escala de grises:** si la imagen a analizar se encuentra por ejemplo a color (RGB), contiene información que no es relevante, ya que el análisis de características para la identificación

se realiza sobre una extracción de información acerca del valor numérico de intensidad de cada pixel y no de acuerdo al color que representa. Por lo tanto, los pixeles de la imagen transformada toman valores entre 0 y 255 pasando por las diferentes tonalidades de grises, donde 0 equivale al color negro y 255 al blanco. Esto se observa en la figura 3.3.

- **Binarización:** se realiza a través de métodos como el de Wolf Jolion [16] y el de Otsu [17], entre otros. La binarización consiste en una reducción de información de una imagen digital en la que los únicos valores posibles son verdadero (1) o falso (0), los cuales corresponden a los colores blanco y negro respectivamente. Para determinar a cuál de estos valores corresponde la información de cada pixel, en el proceso de binarización se establece un umbral, que es un valor dentro de la escala de grises (0 a 255). El mismo es comparado con el valor de cada pixel y si este supera dicho umbral será un 1 y en caso contrario un 0. Además, cabe destacar, que también se pueden especificar umbrales en base a otra tonalidad, si es que se busca un objeto de un color específico. Mediante esta técnica pueden separarse objetos o regiones que pueden ser de interés, del resto de la imagen. El resultado puede verse en la figura 3.4.
- **Aplicación de filtros:** pueden ser utilizados tanto para la remoción del ruido de la imagen como también, para realzar detalles de la misma antes de procesarla [18].

El paso siguiente es identificar la patente en la imagen, mediante algún algoritmo de detección como por ejemplo “edge detection”, “template matching” o “Artificial Neural Network” (ANN) [10][7]. Estos algoritmos de detección, pueden combinarse con los métodos de multiescalamiento y/o ventana deslizante para encontrar la patente, independientemente del tamaño que esta posea, dentro de la

imagen. Una vez identificada la patente en la imagen se la extrae para aplicarle una transformación a escala de grises y, luego, se la binariza con el objetivo de resaltar en blanco los caracteres y los bordes de la misma. Esto se observa en la figura 3.5.



Figura 3.2: Imagen del vehículo que posee la patente a analizar, antes de comenzar el proceso de reconocimiento.



Figura 3.3: Imagen del vehículo que posee la patente a analizar, en escala de grises.



Figura 3.4: Imagen del vehículo que posee la patente a analizar, luego de aplicarle el proceso de binarización.



Figura 3.5: Proceso de preprocesamiento de la región candidata: imagen original (a), en escala de grises (b) y binarizada (c).

Etapla 2: Segmentación de los caracteres de la patente

En esta etapa, las diferentes imágenes que contienen patentes que son adquiridas por el sistema se redimensionan al mismo tamaño, de manera que el tamaño de los caracteres sea similar en todas ellas. El objetivo de esta fase es buscar las partes blancas de la imagen binarizada que cumplan con las especificaciones de los caracteres de la patente. Es por esto que en todo sistema ALPR se deben establecer algunos datos de las matriculas previamente para que el mismo quede configurado adecuadamente. Una vez que se detectan todos los caracteres de la placa, estos son aislados para luego identificarlos. El resultado del proceso de

segmentación se observa en la figura 3.6.



Figura 3.6: Resultado del proceso de segmentación de caracteres.

Etapa 3: Reconocimiento de los caracteres (OCR)

Esta fase recibe los caracteres segmentados de la etapa anterior. Como su nombre lo indica, es la encargada de procesar cada uno de ellos y determinar a qué carácter alfanumérico corresponde. Para ello, pueden utilizarse numerosas herramientas de reconocimiento, tanto de software libre como de origen comercial, siendo la más conocida la llamada Tesseract OCR [19].

Etapa 4: Post-procesamiento

En algunos sistemas, el resultado de la etapa de reconocimiento no es solo la información obtenida del carácter analizado, sino que puede ser una lista de posibles valores con un porcentaje de confianza asociado a cada uno de ellos. En esta fase, el objetivo es tomar los resultados obtenidos y generar una lista de posibles resultados para las patentes, ordenándolas según el porcentaje de confianza. En caso de que el sistema no entregue una lista de posibilidades para los caracteres, solo se obtendrá un resultado. En la figura 3.7, se observa el resultado del proceso realizado por un sistema ALPR. Se muestra la imagen original junto con la cadena de caracteres que se corresponde con la patente.



Figura 3.7: Patente reconocida, resultante de la aplicación de un sistema ALPR.

3.3. Descripción de las patentes y los conjuntos de datos

3.3.1. Patentes Argentinas

Actualmente, existen dos modelos de patentes que se encuentran en vigencia en el país [20] [21] [22].

Patente del MERCOSUR

Las patentes del MERCOSUR poseen las características que pueden observarse en las figuras 3.8, 3.9 y 3.10, y que son listadas a continuación:

- Se implementa a partir del año 2016
- Está dotada de un arreglo de 7 caracteres que consta de letras y números y conforma un serial, embozado en alto relieve
- Elementos de seguridad que posee:

- Bandera del país
- Emblema del MERCOSUR
- Marca de agua
- Tipo ensure
- Estampado en caliente con lámina de seguridad con efecto difractivo y onda sinusoidal
- Colores:
 - Fondo de color blanco
 - Caracteres de color negro
- Dimensiones:
 - Vehículos:
 1. Largo: $400\text{mm} \pm 2\text{mm}$
 2. Alto: $130\text{mm} \pm 2\text{mm}$
 3. Espesor: $1\text{mm} \pm 0,2\text{mm}$
 - Motovehículos:
 1. Largo: $200\text{mm} \pm 2\text{mm}$
 2. Alto: $170\text{mm} \pm 2\text{mm}$
 3. Espesor: $1\text{mm} \pm 0,2\text{mm}$
- Tipo de letra:
 - Fuente: FEEngschrift
 - Alto del carácter en vehículos: 65mm
 - Alto del carácter en motovehículos: 53mm



Figura 3.8: Patente del MERCOSUR: características.



Figura 3.9: Patente del MERCOSUR: medidas de seguridad.



Figura 3.10: Patente del MERCOSUR para automóviles y motocicletas.

Patente argentina antigua

Las patentes antiguas poseen las características que pueden observarse en las figuras 3.11 y 3.12, y que se listan a continuación:

- Se implementa entre el los años 1995 y 2016
- Está dotada de un arreglo de 6 caracteres que consta de letras y números y conforma un serial, embozado en bajo relieve
- Medidas de seguridad para evitar reproducción y/o adulteración que posee:
 - Material de fabricación: aluminio
 - Escudo Nacional a color, en el extremo superior izquierdo
 - La palabra “ARGENTINA”, impresa en su parte media superior en color celeste
 - Sellos circulares con la inscripción “RNPA” distribuidos en forma uniforme en los extremos blancos superior e inferior, los cuales pueden visualizarse haciendo variar la incidencia de luz sobre la misma. Se encuentran indicados en la figura 3.11 con la letra (c)
 - En caso de extravío, robo o hurto de la chapa patente, debe solicitarse una nueva. La única diferencia entre esta y la original es que la nueva, entre las letras y los números, lleva grabado en tamaño menor, pero de igual impresión, un carácter que determina la versión de la placa. Por ejemplo, “D” es de duplicado y “T” es de triplicado, entre otros. Se encuentra indicado en la figura 3.11 con la letra (d)
- Colores:
 - Banda central de color negro mate

- Caracteres sobre la banda central y de color blanco
- Bordes superior e inferior de color blanco reflectante
- Dimensiones:
 - Vehículos:
 1. Largo: 294mm
 2. Alto: 129mm
 - Motovehículos:
 1. Largo: 150mm
 2. Alto: 130mm
- Tipo de letra:
 - Fuente: LicensePlate
 - Ancho de los caracteres en vehículos: 32mm
 - Alto de los caracteres en vehículos: 67mm



Figura 3.11: Patente argentina antigua correspondiente a automóviles.



Figura 3.12: Patente argentina antigua correspondiente a motovehículos.

A lo largo de esta sección, se plateó una numerosa lista de características correspondientes a cada uno de los modelos de patentes vigentes en nuestro país. Sin embargo, hay ciertos puntos que es necesario remarcar:

- Mientras que las matrículas del MERCOSUR poseen caracteres negros sobre fondo blanco, en el modelo antiguo es a la inversa
- En las patentes nuevas, la letra “O” y el “0” son fácilmente diferenciables debido a que este último no se encuentra completo, debido a que posee un corte en la zona superior derecha
- Respecto a las dimensiones de las mismas, el alto es similar en ambas. Sin embargo, el nuevo modelo es aproximadamente diez centímetros más ancha
- En cuanto a la calidad, debe mencionarse que las primeras patentes entregadas correspondientes al modelo del MERCOSUR tuvieron defectos de fabricación. Esto provocó la pérdida de pintura y el descoloramiento de las mismas [23]

3.3.2. Conjuntos de datos

Para probar el sistema OpenALPR y configurar sus parámetros de manera de obtener la mayor efectividad posible en su respuesta, se generaron dos conjuntos preliminares: un set de fotos para las patentes del Mercosur y otro para el modelo antiguo de matrículas argentinas. Ambos conjuntos están compuestos por 20 imágenes traseras de automóviles y camionetas.

Esto se debe a que las motocicletas solo poseen patente en su parte trasera. Por otra parte, cabe destacar que el entorno de todas las imágenes es diferente, ya que no fueron tomadas en un mismo lugar. Por último, todas las fotos fueron tomadas con un smartphone con una resolución de 4128 x 2322 píxeles, y luego se las re dimensionó a 1280x720 píxeles. Esta resolución se adoptó de manera de reducir el tiempo de procesamiento del sistema. La misma es configurable y admite modificaciones en caso de ser necesario. Estas imágenes fueron tomadas bajo estas condiciones, debido a que fueron pruebas preliminares y no se disponía aún de la cámara a utilizar.

Una vez finalizadas las pruebas de los diferentes parámetros del sistema, se decidió ampliar ambos sets de prueba a un tamaño de 165. Mediante estos dos nuevos conjuntos, se comprobó de mejor forma que los parámetros se hayan establecido correctamente. Las imágenes de estos sets fueron tomadas en condiciones que hacen que los mismos sean representativos del estacionamiento de prueba. Las imágenes se tomaron a distancias entre uno y cuatro metros respecto del vehículo, desde la parte trasera, a aproximadamente la altura de la matrícula, a vehículos que se encontraban detenidos y con condiciones de iluminación similares e incluso, en algunos casos, más complejas.

Por último, se crearon dos sets pequeños de 10 imágenes de motocicletas cada uno, para poder analizar el comportamiento del sistema para este tipo de

vehículos.

Todos los conjuntos de imágenes establecidos, se detallan a continuación.

3.3.2.1. Set de 20 imágenes de patentes del MERCOSUR

Dentro de este conjunto, las imágenes poseen las siguientes características:

- Todas las fotos son durante el día
- Nueve fotos fueron tomadas en línea recta a la patente y once desde un costado
- Se tomaron dos imágenes a un metro del vehículo, diecisiete a dos metros y una a tres metros de distancia aproximadamente

3.3.2.2. Set de 20 imágenes de patentes antiguas

Dentro de este conjunto, las imágenes poseen las siguientes características:

- Todas las fotos son durante el día
- Once fotos fueron tomadas en línea recta a la patente y nueve desde un costado
- Se tomaron seis imágenes a un metro del vehículo, seis a dos metros y ocho a tres metros de distancia aproximadamente

3.3.2.3. Set de 165 imágenes de patentes del MERCOSUR

Dentro de este conjunto, las imágenes poseen las siguientes características:

- Se tomaron 160 imágenes de día y 5 de noche
- 70 fotos fueron tomadas en línea recta a la patente y 95 desde un costado

- Se tomaron 33 imágenes a un metro del vehículo, 98 a dos metros, 33 a tres metros y una a cuatro metros de distancia aproximadamente

3.3.2.4. Set de 165 imágenes de patentes antiguas

Dentro de este conjunto, las imágenes poseen las siguientes características:

- Se tomaron 160 imágenes de día y 5 de noche
- 82 fotos fueron tomadas en línea recta a la patente y 83 desde un costado
- Se tomaron 48 imágenes a un metro del vehículo, 90 a dos metros, 26 a tres metros y una a cuatro metros de distancia aproximadamente

3.3.2.5. Set de 10 imágenes de motocicletas con patentes del MERCOSUR

Dentro de este conjunto, las imágenes poseen las siguientes características:

- Todas las fotos son durante el día
- 5 fotos fueron tomadas en línea recta a la patente y 5 desde un costado
- Se tomaron 3 imágenes a un metro del vehículo, 6 a dos metros y 1 a tres metros de distancia aproximadamente

3.3.2.6. Set de 10 imágenes de motocicletas con patentes antiguas

Dentro de este conjunto, las imágenes poseen las siguientes características:

- Todas las fotos son durante el día
- 4 fotos fueron tomadas en línea recta a la patente y 6 desde un costado
- Se tomaron 4 imágenes a un metro del vehículo, 4 a dos metros y 2 a tres metros y de distancia aproximadamente

3.4. Elección y ajuste del sistema para SAE

El objetivo de esta sección, es presentar los diferentes tipos de software que se encuentran disponibles en la actualidad. Luego, se analizará el funcionamiento de dos de ellos en base a lo desarrollado en la sección 3.3.2, donde se explicó el funcionamiento general de un sistema ALPR. Estos sistemas fueron elegidos debido a que, al realizar una investigación sobre las herramientas disponibles, resultaron ser las que se encuentran más difundidas. Además, se expondrán los resultados de la experimentación realizada para ambos sistemas. Por último, se compararán ambos software, demostrando las razones por las que se decidió trabajar con la herramienta OpenALPR.

3.4.1. Software disponibles

Actualmente, existen numerosos trabajos realizados acerca del ALPR en los cuales cada autor lo aborda desde su perspectiva y aporta sus descubrimientos, problemáticas y resultados al investigar y experimentar con él.

Además, se tiene acceso a una gran cantidad de herramientas que permiten realizar el reconocimiento. Las mismas pueden ser sistemas pagos como “Anylines License Plate Scanner” [24], “Plate Recognizer” [25] u “OpenALPR” [26]. En todos estos, no solo se ofrece la detección de patentes, sino que además se entregan otras funcionalidades como el almacenamiento y control en la nube, la posibilidad de integración a cualquier otro proyecto, o bien una aplicación para smartphone, entre otras.

Por otra parte, se pueden encontrar sistemas gratuitos, pero que no son de código abierto como lo es VISART [14]. Estos presentan la desventaja de que el proyecto se debe adaptar a lo que el software ofrece, debido a que este último no

se puede modificar.

Por último, se pueden encontrar sistemas gratuitos de código abierto, entre los que podemos destacar “OpenCV 3 License Plate Recognition” [27] y “OpenALPR” en su versión libre [28].

Todas estas herramientas tienen un principio de funcionamiento similar y deben ser adaptadas al lugar en el que se las pretende utilizar. Esto se debe a que las patentes de los diferentes países, estados o provincias pueden tener diversas fuentes, colores o estar escritas en distintos idiomas.

Por lo tanto, la ventaja de los sistemas pagos se encuentra en la gran variedad de países que contemplan para el reconocimiento. Pero, debido a que su uso implica una tarifa mensual o son muy costos, el desarrollo de este trabajo se focaliza en la utilización de un sistema gratuito de código abierto, el cual se busca adaptar inicialmente a las condiciones de las patentes argentinas y, a futuro, se ampliará a diferentes tipos de matrículas. Las opciones analizadas fueron “OpenCV 3 License Plate Recognition” y la versión libre de OpenALPR.

3.4.2. Funcionamiento de los sistemas elegidos

3.4.2.1. OpenALPR

Este es un software gratuito de código abierto, aunque también existe su versión comercial, utilizado para el reconocimiento automático de patentes de vehículos. OpenALPR está escrito en C++ y tiene la capacidad de analizar imágenes fijas, videos y videos en tiempo real para identificar patentes en ellos y proveerla en forma de texto como salida [29].

Este sistema utiliza diferentes librerías para obtener el resultado, entre las que podemos destacar OpenCV [30] para el procesamiento de la imagen y Tesseract [19] para el reconocimiento de los caracteres.

Etapas del sistema

Este sistema funciona bajo una arquitectura de tipo pipeline, es decir, consiste en ir transformando un flujo de datos secuencialmente a través de diferentes etapas. En este caso, la entrada del sistema es una imagen y la salida la patente del vehículo. La documentación oficial [31] indica que, para obtener la respuesta, el software cuenta con 8 etapas, las cuales se comparan a continuación con las etapas generales de un sistema ALPR descritas en la sección 3.2.2.

Etapas 1: Detección de la patente

Esta fase abarca las dos primeras etapas del OpenALPR que son la detección y la binarización. Esta fase funciona de la misma forma que se explicó para un sistema general, ya que se aplica un pre-procesamiento a la imagen para transformarla a escala de grises, y luego buscar la patente en dicha imagen. Para realizar esta búsqueda, el sistema utiliza un algoritmo denominado LBP (Local Binary Patterns) como algoritmo de detección.

Este es un operador de textura simple pero muy eficiente [32]. Este algoritmo se encarga de extraer las características de la imagen, las cuales son lo que en visión artificial se denomina descriptores. El algoritmo, a partir de una imagen en escala de grises, calcula un valor para cada pixel de la imagen basándose en la vecindad de dicho pixel, es decir, en aquellos pixeles que lo rodean. Para hacer esto le otorga un 1 o un 0 a los pixeles de la vecindad si es que superan o no el valor del pixel central, para luego, a partir del valor binario generado, obtener el resultado LBP del pixel analizado pasando dicho valor a hexadecimal. Esto se puede observar en la figura 3.13.

De esta manera, el sistema es capaz de determinar la o las regiones de una o más posibles patentes en la imagen. Esto se observa en la figura 3.14, donde los rectángulos de color rojo son el resultado de esta primera etapa.



Figura 3.14: Posibles patentes detectadas.

Por último, se toma un recorte de la patente de la imagen en escala de grises y se la binariza para luego ser tratada en las siguientes etapas. Cabe destacar que el sistema crea múltiples imágenes binarizadas para evitar la pérdida de un carácter en el caso de una imagen muy brillante u oscura. Para realizar la binarización, este sistema utiliza los métodos de Wolf-Jolion [16] y Sauvola [34]. El método de Wolf-Jolion propone un sistema para la localización, mejora y binarización del texto en documentos multimedia. En este caso, la detección se realiza aplicando una medida de gradientes acumulados, los cuales se suelen binarizar con el método de Otsu [35]. En este caso, el sistema utiliza el método Sauvola en lugar del de Otsu. El mismo aplica una ecuación de binarización de cálculo, a partir de un umbral, y utilizando la media, el nivel de desviación y la intensidad de los píxeles de la imagen. Si bien este algoritmo es más lento que el de Otsu, en imágenes con

grises muy sutiles y degradados, proporciona resultados mucho más nítidos, más claros y limpios[36].

Etapas 2: Segmentación de caracteres

En OpenALPR esta fase abarca cuatro etapas: el análisis de caracteres, la detección de bordes de la patente, el enderezamiento y la segmentación de caracteres. En este caso, el sistema no recurre a aislar los caracteres directamente, sino que efectúa un procedimiento previo para lograr obtener mejores resultados en el reconocimiento.

En primer lugar, se buscan en las imágenes binarizadas todas las manchas que coincidan con el alto y ancho establecidos para los caracteres de la patente, tal como se observa en la figura 3.15.



Figura 3.15: Detección y análisis de los posibles caracteres.

Una vez ubicados los caracteres, se utiliza el algoritmo de Canny combinado con la transformada de Hough. Mientras el primero permite localizar contornos en la imagen, el segundo se usa para encontrar objetos en una imagen, tales como rectas, circunferencias o eclipses [18].

El objetivo de la aplicación de estos algoritmos es detectar los bordes de la patente para luego realizar el enderezamiento de la imagen y segmentar los caracteres de forma más eficiente. Este resultado, se puede observar en las figuras 3.16(a), 3.16(b) y 3.16(c) donde se pueden ver las líneas ganadoras, tanto verticales como horizontales, luego de aplicar la transformada de Hough.



Figura 3.16: Detección de bordes horizontales (a) y verticales (b), y detección de patente (c).

Por último, para segmentar los caracteres, se utiliza un histograma vertical para encontrar los espacios entre los caracteres de la patente, lo que permite leer y procesar cada carácter por separado. Además, esta etapa se encarga de limpiar los caracteres al remover puntos desconectados, descalificar regiones de caracteres por ser muy chicas e intenta remover los bordes de la patente, de manera de que no sean calificados como un “1” o una “I”. El resultado de todos estos procesos es el que se visualiza en la figura 3.17.



Figura 3.17: Caracteres limpios y segmentados.

Etapa 3: Reconocimiento de caracteres

Esta etapa de reconocimiento de caracteres se corresponde con la séptima etapa de OpenALPR, que lleva el mismo nombre. La misma no presenta ninguna diferencia con lo explicado en la etapa general, ya que reconoce los caracteres y, para cada uno de ellos, entrega todas las posibilidades que hay junto a un

porcentaje de confianza.

Como motor de OCR, este software utiliza Tesseract. Esta herramienta se encarga, en primer lugar, de almacenar los contornos pertenecientes a los objetos de la imagen binaria que se le aporta. Luego, dichos objetos son ordenados en diferentes líneas de texto según la ubicación dentro de la imagen (coordenadas X,Y) de los contornos almacenados. Estas líneas, ahora son divididas en palabras, donde si los caracteres de esta presentan un ancho de separación fijo, cada uno de ellos se introduce en una celda de carácter. En caso contrario se divide únicamente en palabras según valores de espaciado entre ellas predefinidos.

A partir de este momento, el proceso entra en dos fases. En la primera de ellas, se hace un intento por reconocer, mediante un clasificador, cada una de las palabras (o carácter) que conforman el texto. Las que se reconocen positivamente se pasan a un clasificador adaptativo como patrón de entrenamiento, consiguiendo mayor capacidad de acierto al avanzar en el análisis del texto.

La segunda fase consiste en una nueva revisión del texto para intentar reconocer las palabras o caracteres en las que se haya fallado en la primera fase. Esto se hace debido a que el clasificado fue mejorado con los resultados positivos de la primera fase [37].

Una parte de esta etapa se puede visualizar en el fragmento de la figura 3.18.

```

charpos1 line0: threshold 0: symbol A, conf: 88.131264 font: gernany (index 0) size 11px - A conf: 88.131264
charpos2 line0: threshold 0: symbol C, conf: 92.047928 font: serbia (index 2) size 11px - C conf: 92.047928
- G conf: 85.349182
- 0 conf: 81.200813
- 0 conf: 79.586433
-----
charpos3 line0: threshold 0: symbol 9, conf: 91.221046 font: netherlands (index 1) size 11px - 9 conf: 91.221046
charpos4 line0: threshold 0: symbol 0, conf: 94.763901 font: netherlands (index 1) size 11px - 0 conf: 94.763901
charpos5 line0: threshold 0: symbol 5, conf: 93.525917 font: serbia (index 2) size 11px - 5 conf: 93.525917
charpos6 line0: threshold 0: symbol 5, conf: 92.155922 font: gernany (index 0) size 11px - 5 conf: 92.155922
- 3 conf: 78.249588
-----
charpos7 line0: threshold 0: symbol P, conf: 97.415161 font: serbia (index 2) size 11px - P conf: 97.415161
-----
charpos2 line0: threshold 1: symbol C, conf: 94.280418 font: serbia (index 2) size 11px - C conf: 94.280418
- G conf: 85.037682
- 0 conf: 80.338776
-----
charpos3 line0: threshold 1: symbol 9, conf: 90.989983 font: netherlands (index 1) size 11px - 9 conf: 90.989983
charpos4 line0: threshold 1: symbol 0, conf: 94.891357 font: netherlands (index 1) size 11px - 0 conf: 94.891357
charpos5 line0: threshold 1: symbol 5, conf: 93.865128 font: serbia (index 2) size 11px - 5 conf: 93.865128
charpos6 line0: threshold 1: symbol 5, conf: 91.019753 font: gernany (index 0) size 11px - 5 conf: 91.019753
- 3 conf: 77.651123
-----
charpos7 line0: threshold 1: symbol P, conf: 92.990585 font: serbia (index 2) size 11px - P conf: 92.990585
-----
charpos1 line0: threshold 2: symbol A, conf: 86.198639 font: gernany (index 0) size 11px - A conf: 86.198639

```

Figura 3.18: Fragmento del proceso de reconocimiento de los caracteres.

Etapa 4: Post-procesamiento

Corresponde a la etapa final de este sistema y lleva el mismo nombre que la última de las etapas generales mencionada en la sección 3.2.2. En esta instancia se establece un umbral, que es un puntaje que el software calcula para el carácter basándose en la confianza y las ocurrencias de dicho carácter al reconocer los caracteres de las múltiples imágenes binarias que se generan. Entonces, si el valor resultante del carácter se encuentra por debajo del umbral, este queda rechazado como posibilidad y se lo descarta.

Luego, el sistema se encarga de determinar la mejor combinación de los resultados positivos, entregando una lista de posibles patentes ordenándolas de mejor a peor según su confianza. Esta lista, es generada a partir de permutaciones entre las posibilidades que son válidas para cada carácter. Cabe destacar que el número de patentes presente en la misma se puede seleccionar, sabiendo que no siempre se va a obtener esa cantidad máxima.

Por último, esta etapa posee la capacidad de validar el formato de las combinaciones que se obtienen. Esto hace referencia a que se le puede decir al sistema que la patente es de determinada forma (por ejemplo: [letra][letra][letra]-[numero][numero][numero]) para que el mismo entregue solo respuestas que coincidan con dicha forma. De esta manera, se evitan las posibles confusiones entre letras y números, ya que si se reconoce un numero donde se estableció que debía haber una letra, ese resultado queda inmediatamente descartado. El resultado final se puede observar en la figura 3.19.

```

PostProcess Line 0 Letter: 1 A -- score: 128.66 -- occurrences: 4
PostProcess Line 0 Letter: 2 C -- score: 226.708 -- occurrences: 6
PostProcess Line 0 Letter: 2 G -- score: 90.7334 -- occurrences: 3
PostProcess Line 0 Letter: 2 0 -- score: 81.062 -- occurrences: 3
PostProcess Line 0 Letter: 2 0 -- score: 52.2589 -- occurrences: 2
PostProcess Line 0 Letter: 2 ~ -- score: 0.413567 -- occurrences: 1
PostProcess Line 0 Letter: 3 9 -- score: 205.919 -- occurrences: 6
PostProcess Line 0 Letter: 4 6 -- score: 238.561 -- occurrences: 6
PostProcess Line 0 Letter: 5 5 -- score: 231.198 -- occurrences: 6
PostProcess Line 0 Letter: 6 S -- score: 218.901 -- occurrences: 6
PostProcess Line 0 Letter: 6 3 -- score: 45.9007 -- occurrences: 2
PostProcess Line 0 Letter: 6 ~ -- score: 4.09929 -- occurrences: 2
PostProcess Line 0 Letter: 7 P -- score: 160.811 -- occurrences: 4
-- PostProcess Permutation Time: 0.384257ms.
Top 10 Possibilities: AC965SP : 92.0356 <---
Top 10 Possibilities: AG965SP : 83.1648
Top 10 Possibilities: A0965SP : 80.6548
3 total permutations
PostProcess Time: 0.573921ms.
PostProcess Analysis Complete: AC965SP -- MATCH: 0
Result Generation Time: 0.257995ms.
Total Time to process image: 600.32ms.
plate0: 3 results
- AC965SP confidence: 92.0356
- AG965SP confidence: 83.1648
- A0965SP confidence: 80.6548

```

Figura 3.19: Fragmento de la etapa de postprocesamiento junto con el listado de patentes obtenido como resultado del proceso completo.

Opciones de funcionamiento del sistema

El software OpenALPR es capaz de reconocer matrículas a partir de tres diferentes fuentes: imágenes fijas, videos o videos en tiempo real. Cabe destacar

que el proceso de obtención de la patente es idéntico para los tres casos. Esto se debe a que, en el caso de los videos, lo que se hace es analizar los diferentes frames del mismo para encontrar la patente. Es decir, el análisis se realiza a partir de una imagen que se extrae del video.

Luego de evaluar las tres posibilidades para este proyecto, se considera más apropiado y sencillo utilizar el video en tiempo real. Esto se debe a que el sistema permanecerá continuamente analizando el video para encontrar una patente. En el caso de las imágenes fijas, primero se debería determinar el momento en que se realizaría la captura de la imagen, lo que implicaría el desarrollo de un código externo para ello. Luego, se la utilizaría para realizar el reconocimiento. Para los videos, la situación es similar. Sin embargo, se debería determinar cuándo comenzar y finalizar el video. Además, este proceso es mucho más costoso computacionalmente y, por ende, se demora más tiempo en obtener una respuesta.

Por último, la principal ventaja de trabajar con video en lugar de imágenes fijas es que al estar continuamente analizando los frames del mismo, el sistema es capaz de entregar la respuesta para una misma patente varias veces. De esta manera, es posible obtener un resultado con mayor nivel de confianza. Para lograr esto con imágenes, se deberían obtener varias por vehículo y analizar cada una de ellas. Esto aumentaría los recursos computacionales requeridos.

Por estos motivos, se determinó que la mejor opción es trabajar a partir del video en tiempo real. Para ello, el sistema cuenta con un modo de funcionamiento denominado `alprd` (`alpr daemon`) el cual funciona en segundo plano y permite entregarle al sistema un stream de vídeo en tiempo real.

Cuando el sistema detecta una matrícula en el vídeo, procesa el frame. A partir del mismo se obtiene el número de la patente junto con su confianza, y algunas características adicionales como el tiempo de procesamiento, la ubicación de la patente en la imagen, entre otras.

Una vez obtenida dicha información, el sistema tiene dos formas de seguir adelante. En la primera de ellas, esta es enviada a una cola de trabajo y luego a un servidor HTTP, del cual se pueden extraer los datos que contiene. La segunda forma, que es la implementada en este proyecto, es utilizando únicamente la cola de trabajo. En este caso, la misma funciona bajo el protocolo Beanstalk. Este es un protocolo que corre sobre TCP y se encarga de crear un servidor web al que el sistema sube el resultado como un nuevo trabajo o Job [38].

Para recuperar la información contenida dentro de los trabajos, es necesario extraer a estos últimos de la cola. Para ello, se usa un consumidor de cola. En este caso, para realizar esto, se desarrolló un código en C++ (podría ser en otro lenguaje) al que se añadió la librería Beanstalk-client para utilizar algunas de sus funciones con el objetivo de conectarse a este servidor, seleccionar la cola a usar, reservar un trabajo, entre otras [39].

Finalmente, la información recuperada se encuentra en formato “.Json”. De esta manera, a partir de un código se puede acceder fácilmente a las diferentes secciones de la misma, pero al extraer los trabajos mediante el mismo, esta pierde el formato y como resultado se obtiene una cadena de caracteres. Por lo tanto, para evitar tener que buscar la información deseada (como la patente obtenida o la confianza) en dicho texto sin formato, se dota al código de C++ de otra librería. La misma permite llevar todo ese texto al formato original (“. Json”) y trabajar con el mismo en C++ [40].

De esta manera, el sistema es capaz de obtener la información de la patente del vehículo que se ha posicionado delante de la cámara [41].

3.4.2.2. OpenCV 3 License Plate Recognition

El mismo es un programa de código abierto desarrollado por Chris Damhs, el cual está implementado sobre el lenguaje C++ y está basado en la utilización de

la librería OpenCV [30].

Este sistema tiene una estructura tipo pipeline, ya que se atraviesan diferentes etapas para lograr obtener la patente del vehículo deseado.

Respecto a la etapa general inicial de los sistemas ALPR, este software se diferencia en que, en lugar de buscar la patente en la imagen, busca primero posibles caracteres de esta para luego ubicarla. Para realizar esto, aplica en primera instancia un pre-procesamiento sobre la imagen en el que se la transforma a escala de grises y luego se la binariza mediante la función `AdaptiveThresh`. La misma usa binarización Gaussiana. Los resultados parciales de este proceso pueden observarse en las figuras 3.20a, 3.20b y 3.20c.



(a) Imagen original.



(b) Imagen en escala de grises.



(c) Imagen binarizada.

Figura 3.20: Resultados parciales de la etapa inicial del software basado en OpenCV.

Una vez que la imagen se encuentra binarizada se procede a buscar todos los

contornos dentro de la misma. Luego, se eliminan aquellos que no coincidan con las dimensiones establecidas para los caracteres. Esto se puede observar en las figuras 3.21a y 3.21b.



(a) Contornos dentro de la imagen binarizada.

(b) Caracteres que superaron el proceso de eliminación.

Figura 3.21: Proceso de búsqueda de contornos y eliminación de aquellos que no poseen dimensiones de caracteres.

Con estos contornos, el sistema procede a generar diferentes grupos de caracteres, los cuales poseen características similares entre sí como el tamaño y la ubicación en las fotos. Si el grupo generado no posee un mínimo de caracteres preestablecido, queda descartado. Este resultado se observa en la figura 3.22.



Figura 3.22: Grupos de caracteres coincidentes.

Por último, basándose en la ubicación de los caracteres, el sistema los ordena y realiza un recorte de la región donde se encuentra el grupo en la imagen original. Por lo tanto, en la etapa de detección de patente, este software no entrega la ubicación de la matrícula en la imagen, sino que entrega un conjunto de posibles patentes, a las cuales se les aplica el mismo pre-procesamiento que a la imagen original, es decir, se lleva a escala de grises y se las binariza. Esto se ve en las figuras 3.23a, 3.23b y 3.23c.

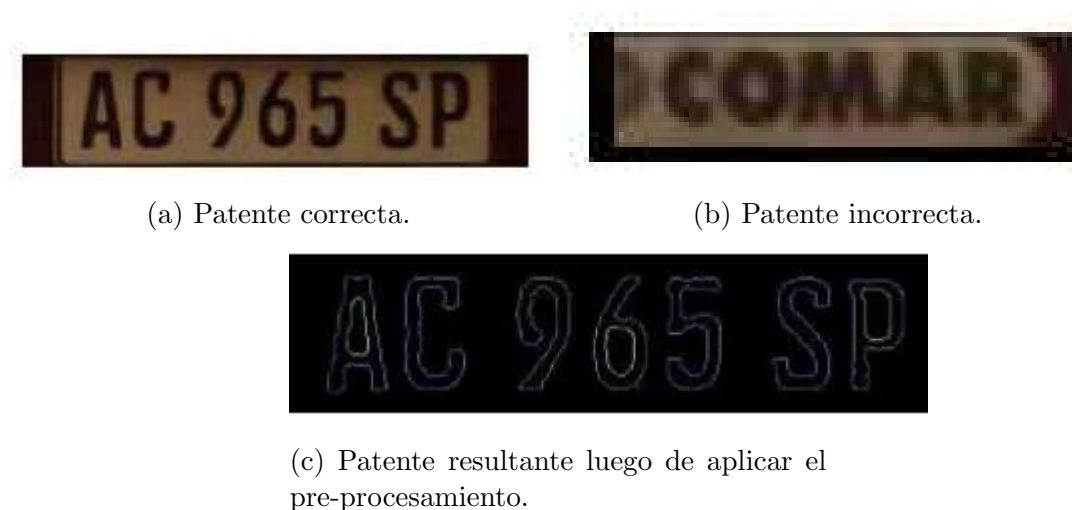


Figura 3.23: Proceso desarrollado en la etapa de detección de patente.

Siguiendo con el esquema general de los sistemas ALPR, esta herramienta busca segmentar los caracteres de las posibles patentes que se hayan encontrado. Para esto, realiza el mismo procedimiento utilizado para encontrar caracteres en la imagen original, pero ahora sobre la imagen de la posible patente. Una vez encontrados, mediante una comparación de tamaños y ubicación, se remueven los caracteres internos (círculo interior dentro del “0” o la “O”) o superpuestos, que pueden ser considerados como un carácter diferente por sus dimensiones. De esta manera, se evita incluir dos veces el mismo carácter o caracteres extra. La figura

3.24 muestra el resultado de esta etapa.



Figura 3.24: Resultado de la remoción de caracteres internos.

En la fase de reconocimiento de caracteres (tercera etapa general de los sistemas ALPR) este software presenta falencias ya que, en primer lugar, solo aplica el reconocimiento a la posible patente que contenga un mayor número de caracteres. Esto lo hace muy sensible a variaciones en el entorno de la imagen. En segundo lugar, utiliza el algoritmo K-NN (K Nearest Neighbours) [42] el cual, si bien es capaz de entregar una respuesta, no es un motor de OCR. Este es uno de los algoritmos supervisados más simples de Machine Learning, el cual se usa mayormente para la clasificación. Básicamente determina a que grupo pertenece un nuevo elemento dependiendo de cómo están clasificados sus vecinos más cercanos. Es decir, almacena todos los casos conocidos que se poseen y los utiliza para clasificar nuevos casos dependiendo de la similitud que tenga con las características determinadas [18].

Por último, se encuentra la fase de post-procesamiento. En esta, el sistema toma el resultado obtenido para cada uno de los caracteres, los cuales fueron ordenados previamente, e informa el resultado marcando con un recuadro rojo la patente seleccionada. Junto a ella, coloca los valores de los caracteres. Este último proceso se puede observar en la figura 3.25.



Figura 3.25: Resultado entregado por la fase de post-procesamiento.

3.4.3. Experimentación y puesta a punto de los sistemas

En esta sección se desarrollarán las modificaciones realizadas sobre cada uno de los sistemas trabajados, con el objetivo de mejorar los resultados devueltos por ellos. Además, podrá observarse el impacto provocado por las mismas, al visualizar las respuestas obtenidas antes y después de modificar dichos sistemas.

Para evaluar el funcionamiento de estos sistemas, se utilizó un ordenador con sistema operativo Ubuntu 18.04 LTS, un procesador Intel Core I5 de segunda generación con cuatro núcleos y 4GB de memoria RAM.

3.4.3.1. OpenALPR

Inicialmente, se utilizaron los sets de 20 imágenes, descritos en la sección 3.3.2, para evaluar el sistema con sus parámetros por defecto. Los resultados de esta prueba se muestran en la tabla 3.1, donde en la columna “=” se muestran las patentes reconocidas correctamente, en la columna “X” aquellas que fueron detectadas por el sistema con algún error en uno o más de sus caracteres y, por último, en la columna “NF”, se encuentran las imágenes en las que el sistema no

detectó ninguna patente. Esta simbología se mantendrá a lo largo de la sección de experimentación.

	Set de 20 imágenes		
MERCOSUR	75 % (15)	10 % (2)	15 % (3)
Antiguas	40 % (8)	5 % (1)	55 % (11)

Tabla 3.1: Resultados de la detección con el entrenamiento original.

Para mejorar el rendimiento de este software, fue necesario adaptarlo a las patentes argentinas. Para ello, se crearon diferentes archivos, los cuales se describen a continuación:

1. Archivos de configuración de las patentes:

A partir de lo expuesto en la sección 3.3.1, se crearon dos archivos de configuración (archivos de extensión “.conf”), uno para las patentes del MERCOSUR y otro para el modelo antiguo. Dentro de estos archivos, se configuraron las dimensiones de cada una de las placas: alto, ancho, tamaño de los caracteres, entre otros. Además, se estableció la cantidad máxima y mínima de caracteres que debe poseer cada patente: 7 para modelo MERCOSUR y 6 para el antiguo. Por otra parte, el software está preparado por defecto para leer letras negras sobre un fondo blanco. Sin embargo, trabajando sobre estos archivos es posible invertir los colores de la patente para poder leer letras blancas sobre un fondo negro. Esto último se realizó únicamente en el archivo de las patentes antiguas. Adicionalmente, en estos archivos se debe configurar el entrenamiento que se aplica al motor de OCR, y el archivo del detector a utilizar.

Luego, debido a las similitudes que presentan las dimensiones de las patentes

del MERCOSUR con respecto a las de las matriculas europeas, se decidió establecer estos parámetros en forma coincidente con los correspondientes a esta región, los cuales ya forman parte de la versión de código abierto de este software. En el caso del modelo antiguo ocurre algo similar con respecto a las matrículas estadounidenses, por lo que se procede de la misma forma que en el caso anterior. Por último, cabe destacar que el resto de los parámetros que se deben establecer mantuvieron los valores correspondientes a los de las matriculas europeas y estadounidenses, respectivamente.

2. Archivos de Post-Procesamiento:

Como se mencionó en la sección 3.4.2, el sistema es capaz de validar el formato de las combinaciones que se forman. Para realizar esto, el software utiliza, para cada estilo de patente que posee, un archivo de extensión “.patterns”, en el cual se indica el o los formatos posibles de cada patente. Por lo tanto, se creó un archivo para las patentes del MERCOSUR donde se estableció el formato @@ ### @@, donde @ representa las letras y # los números.

Por otra parte, se debe destacar que en este archivo se pueden agregar los formatos del resto de las patentes del MERCOSUR, de forma de poder reconocerlas, ya que las dimensiones son las mismas. Finalmente, para las patentes argentinas antiguas se creó otro archivo en el que se estableció el formato @@@ ###.

3. Openalpr.defaults.conf:

Es el archivo de configuración general del sistema. Contiene los parámetros por defecto establecidos en el sistema. Con el objetivo de mejorar el funcionamiento del mismo en el caso de las patentes argentinas se modificaron

algunos de estos valores. Entre ellos, podemos destacar los siguientes:

- **detection_iteration_increase**

Por defecto, este valor está establecido en 1.1. Sin embargo, por los resultados obtenidos en la experimentación, se considera que el valor 1.07 es el que otorga mejores resultados. Cabe destacar, que este parámetro es el porcentaje de incremento del cuadro del algoritmo LBP para cada iteración, donde cuanto más bajo es su valor, más lento es el sistema.

- **must_match_pattern**

Este valor por defecto es 0. Su función al tomar el valor uno es informarle al sistema que las respuestas que otorgue deben coincidir con los formatos establecidos en los archivos de post-procesamiento y, en caso contrario, descartarla. Se lo seteó en 1.

- **postprocess_min_confidence**

Por defecto, este parámetro está establecido en 65. Este valor determina el mínimo de confianza que debe poseer cada símbolo luego de pasar por el OCR para ser considerado. Se decidió disminuirlo a 55, debido a que en este valor se obtenían una mayor cantidad de respuestas correctas que con el valor por defecto.

- **detection_mask_image**

Por defecto este parámetro está vacío, lo que significa que el sistema debe analizar toda la imagen en busca de la patente. Al establecer una máscara, como se observa en la figura 3.26, el sistema solo analizará las partes blancas de la imagen e ignorará las partes negras, generando un resultado más veloz. De esta forma, se logra reducir el tiempo de procesamiento del algoritmo LBP, disminuyendo el tiempo total de procesamiento en aproximadamente 30 % con una máscara como la de

la figura 3.26. Cabe aclarar que el tiempo que se reduzca depende de la forma que posea la máscara.



Figura 3.26: Ejemplo de máscara que puede ser seteada.

- **Prewarp**

Este parámetro permite adicionarle al software la configuración realizada sobre la cámara de video. De esta manera es posible reconocer más eficientemente las matriculas de los vehículos que no se encuentran de frente a la cámara. A pesar de que actualmente no se lo está utilizando, a futuro, este parámetro será modificado.

- **max_plate_width_percent y max_plate_height_percent**

Por defecto, estos valores son 100. Se los seteó en 30 a ambos. De esta manera, se le informa al sistema que las posibles patentes no pueden ocupar más de un 30 % de la imagen. Por lo tanto, descartarán las posibles patentes que posean un valor superior al establecido.

4. Archivos para el funcionamiento del ALPRD:

Para poder ejecutar este modo de funcionamiento y reconocer las patentes argentinas, se debió modificar el archivo “alprd.conf”. En el mismo, se le

provee al sistema del URL que genera el streaming realizado por la cámara, en los parámetros “country” y “pattern” se establecen los archivos correspondientes creados para las patentes argentinas y, por último, se configura el parámetro “topn” en 1, ya que solo interesa obtener la mejor patente y no una lista de posibles matrículas. Además, se le configuró un ID a la cámara para poder distinguirla de las demás.

Por otro lado, como el estacionamiento considerado posee una vía de entrada y otra de salida, para que las cámaras situadas en las vías no suban sus trabajos a la misma cola de trabajo, se modificó el archivo “daemon.cpp” de manera que el sistema cree dos colas, una para la entrada y otra para la salida. Cabe destacar que una vez que se modifica este archivo se debe volver a compilar el software de manera que estos cambios tengan efecto sobre el funcionamiento del sistema.

Finalmente, para poder diferenciar entre las colas de entrada y salida, se debe crear un nuevo archivo “alprd.conf” con el ID de la cámara y el URL, referidos a las cámaras de salida. Para poder tener funcionando simultáneamente el sistema de entrada y el de salida, se debe tener una copia de los archivos “alprd.conf” y “openalpr.conf” (archivo que posee los mismos valores que openalpr.defaults.conf) para la entrada y, en una carpeta diferente, otra copia destinada a la salida, donde el “alprd.conf” debe estar seteado con los valores correspondientes a la salida. Por lo tanto, se deben ejecutar ambos archivos “alprd” en forma independiente, indicando la ubicación del archivo en el caso de que no se encuentre en la carpeta por defecto. De esta forma, las cámaras que se encuentran en la vía de entrada subirán sus trabajos a la cola de entrada y las de salida a su respectiva cola.

Una vez realizadas estas modificaciones, se volvió a evaluar el sistema utilizando los mismos sets de pruebas que al inicio, lo que permitió verificar las mejoras logradas. Los resultados se observan en la tabla 3.2. Cabe destacar que el parámetro `detection_iteration.increase` es el que produce el mayor impacto sobre el sistema, por lo que los nuevos resultados se presentan para tres diferentes valores de este parámetro.

	1.1			1.07			1.04		
	=	X	NF	=	X	NF	=	X	NF
MERCOSUR	80 %	0 %	20 %	90 %	0 %	10 %	90 %	0 %	10 %
Antiguas	70 %	0 %	30 %	85 %	0 %	15 %	95 %	0 %	5 %

Tabla 3.2: Variación de los resultados de los set de 20 imágenes.

Por último, se utilizaron los sets de 165 imágenes para obtener un resultado que sea más significativo. Dichos resultados se muestran en la tabla 3.3.

	1.1			1.07			1.04		
	=	X	NF	=	X	NF	=	X	NF
MERCOSUR	92.73 %	3.03 %	4.24 %	95.76 %	0 %	4.24 %	95.76 %	0 %	4.24 %
Antiguas	88.48 %	0.61 %	10.91 %	93.94 %	0 %	6.06 %	94.55 %	1.21 %	4.24 %

Tabla 3.3: Resultados obtenidos para los sets de 165 imágenes.

Utilizando una muestra significativa de imágenes, se pudo verificar que al setear el parámetro mencionado anteriormente en 1.04 en lugar de 1.07 no se producen variaciones significativas en el reconocimiento. Mientras que en las patentes del MERCOSUR los resultados fueron los mismos que con el valor original del parámetro, en las antiguas solo hubo una mejoría de 0.61 %. Considerando que el tiempo de procesamiento es un factor muy importante y que el mismo

debe ser el menor posible, se establece 1.07 como valor para el parámetro `detection_iteration_increase`.

Cabe destacar que la experimentación fue realizada a partir de imágenes fijas. Sin embargo, el resultado de la misma es igualmente válido para el modo de funcionamiento a partir de un video en tiempo real debido a que procesar un frame del video es lo mismo que procesar una imagen fija.

Luego de analizar el comportamiento del sistema con patentes de automóviles y camionetas se comprobó el funcionamiento del mismo con patentes de motocicletas. Este análisis se llevó a cabo sobre un número reducido de imágenes debido a que solo se buscó analizar su factibilidad, dejando a futuro el desarrollo de un conjunto de prueba mayor. Se construyeron dos sets de diez imágenes cada uno, uno con patentes del MERCOSUR y el otro con antiguas. Debe destacarse que se crearon los mismos archivos que en el caso de los otros vehículos, a excepción del archivo de configuración para las patentes del MERCOSUR. Esto se debió a que el sistema ya proveía el archivo correspondiente a las matrículas del MERCOSUR de las motos de Brasil, teniendo en cuenta que todas las matrículas del MERCOSUR son muy similares, como se mencionó anteriormente. Por último, debe mencionarse que las pruebas fueron realizadas con los parámetros de los demás archivos ya modificados, y no con los que se encontraban seteados por defecto. Los resultados obtenidos se pueden observar en la tabla 3.4.

	=	X	NF
MERCOSUR	100 %	0 %	0 %
Antiguas	90 %	0 %	10 %

Tabla 3.4: Resultados obtenidos para el análisis de los sets con imágenes de motocicletas.

3.4.3.2. OpenCV 3 License Plate Recognition

Al igual que para el otro sistema, se realizaron las pruebas iniciales con la configuración por defecto sobre los mismos sets de 20 imágenes. En este caso, los resultados obtenidos son los observados en la tabla 3.5.

	Set de 20 imágenes		
	=	X	NF
MERCOSUR	25 %	70 %	5 %
Antiguas	0 %	45 %	55 %

Tabla 3.5: Resultados del reconocimiento desarrollado con la configuración por defecto.

Este software fue diseñado para el reconocimiento de matrículas estadounidenses, por lo que fue necesario modificarle las dimensiones de los caracteres para adaptarlos a los de las patentes argentinas.

Por otra parte, como se mencionó previamente, para el reconocimiento de caracteres se recurre al algoritmo K-NN, el cual utiliza dos archivos: “classification.xml” e “images.xml”. Los mismos son usados para entregar una respuesta para los caracteres que son analizados por el sistema. Por lo tanto, si se observan estos archivos, se puede verificar que fueron realizados para un formato de letra que no es el propio de las patentes argentina. Esto puede producir errores en el reconocimiento. En consecuencia, se desarrollaron dos nuevos archivos de cada tipo. El primer par se generó mediante una imagen que contenía todas las letras en mayúscula y todos los números con la fuente de las patentes del MERCOSUR (FEEngschrift). El segundo par se generó de la misma forma, pero con la fuente del modelo antiguo de patentes argentinas (LicensePlate).

Para crear dichos archivos, se utilizó el código “OpenCV 3 KNN Character

Recognition”, basado en la librería OpenCV, desarrollado por Chris Damhs [43]. Mediante el archivo “GenData.cpp” se pueden generar fácilmente los archivos mencionados para luego utilizarlos en el código descripto anteriormente.

Finalmente, se estableció la cantidad mínima y máxima de caracteres que debe poseer la matrícula.

Luego de aplicar estas modificaciones, los resultados que se obtuvieron para los mismos sets de 20 imágenes y para los nuevos de 165 son los que se muestran en la tabla 3.6.

	Set de 20 imágenes			Set de 165 imágenes		
	=	X	NF	=	X	NF
MERCOSUR	60 %	35 %	5 %	37.58 %	55.55 %	7.87 %
Antiguas	0 %	45 %	55 %	0 %	68.48 %	31.52 %

Tabla 3.6: Resultados de la detección con el entrenamiento desarrollado.

Cabe destacar que no se efectuó la experimentación con patentes de motocicletas con este software debido a que no posee la capacidad de entregar resultados de patentes multilinea.

3.4.4. Comparación de los modelos

Si bien ambos sistemas son capaces de reconocer la patente que se observa en la imagen analizada, con el software OpenALPR se logra obtener un porcentaje de éxito mayor. Esto se puede confirmar al ver los resultados obtenidos por ambos sistemas para los sets de 165 imágenes. Por un lado, en el caso de las matriculas del MERCOSUR, con el OpenALPR se obtiene una mejora del 60 % aproximadamente en el reconocimiento correcto respecto del OpenCV. Por otro lado, en el caso del modelo antiguo, se obtuvo una mejora del 95 %.

Luego de analizar el comportamiento de los sistemas, se determinó que la única ventaja que posee el código basado en OpenCV es que, al determinar primero la ubicación de los caracteres y luego la patente, este es capaz de detectar todo tipo de patentes independientemente de su tamaño. En este aspecto el software OpenALPR requiere un conjunto de archivos específicos para cada formato de matrícula.

Por otra parte, OpenALPR presenta varias ventajas sobre el otro sistema que son, junto con la efectividad, las razones por las que se decidió trabajar con este software. Entre estas ventajas podemos destacar:

- Es más veloz
- Al detectar la ubicación de la patente y sus bordes, puede descartar con mayor facilidad posibles patentes
- Otorga una lista de posibilidades sobre la patente con un porcentaje de confianza, lo que permite detectar errores del OCR
- Posee la opción de invertir los colores de las patentes de manera de analizar cualquier formato bajo los mismos colores. De esta manera se evita tener que hacer un procesamiento previo de la imagen
- Brinda la posibilidad de validar las combinaciones de caracteres obtenidas según el formato que tengan
- Permite leer patentes multilínea, como es el caso de las matrículas de las motocicletas
- Permite el tratamiento de video y video en tiempo real, que fue la opción elegida para este proyecto

En lo que respecta a los errores, en el caso del código de OpenCV, se debe mencionar que el sistema es muy sensible al entorno de la imagen. Cualquier cartel o conjunto de letras concatenadas que supere la longitud de caracteres de la patente va a ser considerado como la patente, ya que el sistema analiza una única posible matrícula y es la que contenga la mayor cantidad de caracteres. También es posible que dentro del entorno se consideren como posibles caracteres partes del fondo de la imagen, debido a que el sistema trabaja con contornos y sus dimensiones para detectar los caracteres. Además, al no utilizar un motor de OCR propio, se producen confusiones entre los caracteres que son similares entre sí, como en el caso del “0”, “O” y “Q” o bien “B” y “8”, entre otros.

Por otra parte, deben considerarse los posibles errores que se producen al trabajar con OpenALPR. En cuanto a las matrículas del MERCOSUR, se pudo identificar que los caracteres poseen escritos a lo ancho de ellos las palabras “MERCOSUR” y “Argentina” en color gris. Esto puede presentar grandes inconvenientes debido a que, si se logran apreciar en la imagen, al momento de reconocer los caracteres el sistema falla y no logra entregar una respuesta. Un ejemplo de este caso se presenta en las figuras 3.27 y 3.28.



Figura 3.27: Patente con palabras internas apreciables.



Figura 3.28: Reconocimiento de caracteres en patente con palabras internas apreciables.

Como se puede observar en la figura 3.27, la patente de la imagen resulta muy brillante por la forma en que la luz impacta sobre ella. Debe destacarse, en base las pruebas realizadas, que otro factor que puede influir para que esto suceda o no es el ángulo con el que se toma la imagen. Realizando variaciones sobre el mismo, se observó que el problema puede aparecer incluso si la patente no es tan brillante como se observa en el ejemplo anterior.

La solución a este problema es tomar la fotografía a una distancia entre 2.5 y 3 metros aproximadamente, dependiendo de qué tan iluminada esté la patente y el ángulo que posea la cámara. A estas distancias, si bien se siguen apreciando las palabras dentro de los caracteres, al segmentarlos e intentar reconocerlos no aparecen tan manchados como en el caso de la figura 3.28, donde la imagen fue tomada a 1 metro de distancia.

Un problema que se produjo con ambos tipos de patente es la visualización de una sombra sobre la misma. Esto genera que a la hora de segmentar los caracteres aparezcan cortados y no se obtengan resultados. Cabe destacar que, trabajando a una distancia como la ya mencionada, el sistema igualmente puede fallar pero se vuelve mucho más robusto ante circunstancias como estas.

Por último, en cuanto al modelo antiguo, surgió una mayor cantidad de problemas en relación al ángulo de la cámara. Esto hace referencia a que rotaciones

pequeñas pueden generar conflictos al reconocer caracteres como la “O” y la “Q”, debido a que en estas matrículas los mismos son muy similares. Si bien este problema también se da en las patentes del MERCOSUR con letras como la “V”, la cual se confunde con la “W”, y la “M” que se confunde con la “H”, el mismo tiene una ocurrencia mucho menor que con el otro modelo.

3.5. Conclusiones

A lo largo de este capítulo se han introducido los sistemas ALPR, pasando por sus diferentes aplicaciones y empresas que los desarrollan y/o utilizan a nivel mundial. Además, se ha explicado el funcionamiento de un sistema ALPR en forma general y se ha hecho hincapié en el funcionamiento y la experimentación de dos software con los que se trabajó. A partir de estos últimos, se pudo observar que no todos los sistemas ofrecen los mismos servicios y calidad, ya que presentan variaciones a la hora de llevar a cabo las etapas generales, dependiendo de la visión adoptada por la empresa o persona que los haya desarrollado. Por estos motivos, a la hora de elegir un sistema es de vital importancia analizar la aplicación en la que se lo va a utilizar, de manera de determinar el tipo de software que se requiere (pago, gratuito o de código abierto). Además, debe considerarse la forma en la que los mismos funcionan, de manera de determinar cuál de estos se adapta mejor a las necesidades.

Se puede observar que la adaptación de un sistema ALPR para modelos de patentes que no vienen incorporadas en el sistema no es una tarea trivial, ya que se requiere la modificación y ajuste de una gran cantidad de parámetros. Además, se debe tener en cuenta la posibilidad de que el mismo no incluya los archivos de entrenamiento para el motor de OCR y el detector para el nuevo tipo de patentes. En este caso, sería necesario desarrollar dichos entrenamientos, lo

cual es una tarea muy compleja. En el caso particular de este proyecto, debido a la similitud de las patentes argentinas con otros tipos de patentes, se pudieron reutilizar esos archivos entrenamientos a partir de los destinados a otros países. Sin embargo, en un futuro se debería realizar un entrenamiento propio para lograr mejores resultados para todos los modelos de patentes argentinas.

Capítulo 4

Segundo núcleo: Placa I/O

FALTAN COMPLETAR ALGUNAS COSAS

En el presente capítulo se tratarán los aspectos relacionados con el desarrollo del segundo gran núcleo del proyecto. El mismo consiste en el diseño y construcción de la placa encargada de controlar los equipos a ubicados en las zonas de ingreso y egreso del establecimiento. Se presentan los esquemas circuitales correspondientes a la fuente de alimentación y a la parte lógica de la placa. Finalmente, se analizan los diversos periféricos comerciales que se encuentran implementados en el sistema, y que son administrados por ella.

4.1. Componentes principales

En esta primera sección se realiza un breve análisis acerca de los dos principales elementos que constituyen la placa de control. Los mismos son el microcontrolador y el módulo de comunicación WiFi.

4.1.1. Microcontrolador

Para el reconocimiento del estado de los periféricos del sistema y el uso de dicha información para controlar la ejecución de diversas acciones se escogió el microcontrolador de 8 bits Atmega328P. El mismo es fabricado por la empresa Atmel. El microcontrolador cuenta con 28 pines, de los cuales 23 son líneas de entrada-salida programables. De ellas, 14 son digitales. De estas últimas, dos pertenecen a un puerto serial USART, utilizado para la transmisión de datos con el módulo de comunicación WiFi. Adicionalmente, se utilizan 11 pines digitales para el sensado del estado de los periféricos y la distribución de las tareas a realizar por los mismos, el control del encendido del módulo WiFi y la muestra de señales luminosas de alarma en caso de fallas [44]. Finalmente, el microcontrolador requiere 5V de tensión de alimentación para funcionar correctamente. Este componente se ilustra en la imagen 4.1.

Ver si falta mencionar alguna info importante del micro.



Figura 4.1: Microcontrolador Atmega328P utilizado.

4.1.2. Módulo de comunicación WiFi

El módulo utilizado es el ESP-01, que es fabricado por Ai-Thinker y se basa en la familia de módulos WiFi “ESP8266” de Espressif Systems. El mismo se observa en la figura 4.2. Este pequeño módulo puede utilizarse como adaptador Wi-Fi,

que es la función que cumple en este proyecto. Mediante el mismo, se le agregó acceso inalámbrico a Internet al microcontrolador que maneja la placa, utilizando comunicación serie (USART). El módulo posee las siguientes características [45]:

- Funciona en la banda de 2.4GHz
- Soporta WPA/ WPA2
- Rango de tensión de operación: 3 - 3.6V
- Consumo máximo de corriente: 170mA
- Puede funcionar como Estación, como Access Point (AP) o como ambas simultáneamente

Finalmente, el mismo posee ocho pines. Entre los mismos se encuentran un reset, un pin de habilitación, dos pines de alimentación y cuatro pines digitales, donde dos de ellos corresponden a una UART. En este proyecto, la UART se utiliza para la comunicación con el microcontrolador, mientras que uno de los pines digitales permite implementar una señal luminosa en caso de falla.

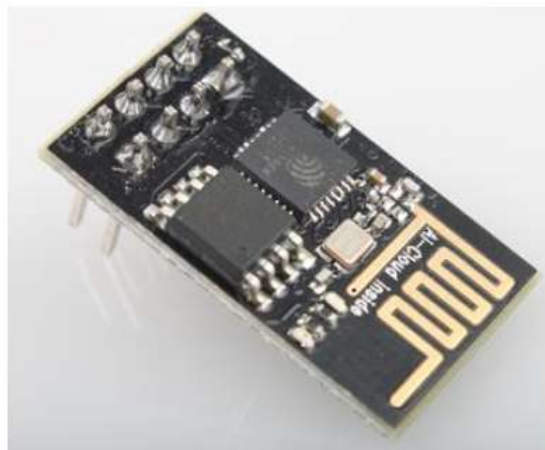


Figura 4.2: Módulo de comunicación WiFi ESP-01 utilizado.

ENCONTRAR Y VER SI SE ESCRIBE JUSTIFICACION SOBRE EL USO DE WIFI EN LUGAR DE OTRO PROTOCOLO. SI SE COMPLICA, SPEECH SOBRE POR QUE COSA PUEDE CAMBIARSE A FUTURO

4.2. Fuente de alimentación

Este proyecto contempla el diseño y construcción de una fuente de tensión regulada. La misma se encarga de alimentar las barreras infrarrojas, el detector magnético, el microcontrolador y el módulo de comunicación WiFi. Para ello, cuenta con salidas de 12V, 5V y 3.3V de tensión continua. Fue implementada a partir de un transformador de 220Vac/12Vac. Su corriente de carga efectiva es X (medirla con el sistema completo conectado y funcionando). El diagrama circuital de la fuente se observa en la figura 4.3.

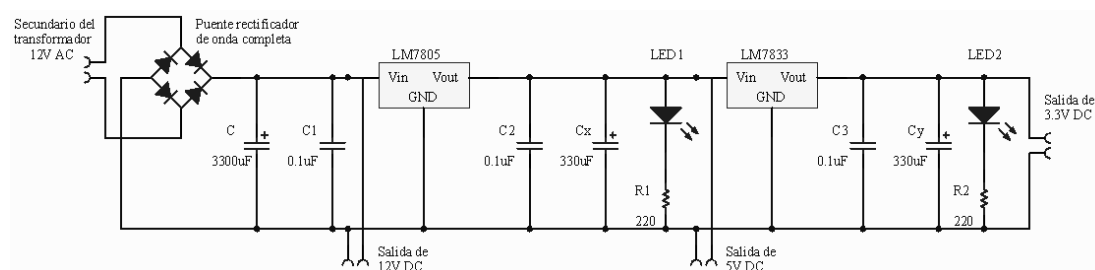


Figura 4.3: Fuente de alimentación de tensión continua del sistema.

La fuente además cuenta con capacitores cerámicos y electrolíticos para filtrar los ruidos de alta y de baja frecuencia, respectivamente, que puedan ingresar al sistema a través de la misma, dado que se la utiliza para alimentar elementos externos, como es el caso de los periféricos (los cables actúan como antenas).

Adicionalmente, se realizó el cálculo del disipador asociado al regulador de tensión LM7805 para que su temperatura no supere los 42°C.

Finalmente, se cuenta con una protección frente a sobretensiones en el lado primario del transformador montada sobre la caja contenedora de la placa.

4.3. Diseño de la etapa lógica y de comunicación

El esquema circuital correspondiente a esta etapa se ilustra en la figura 4.4. Para garantizar el correcto funcionamiento del circuito básico integrado por el microcontrolador y el módulo de comunicación WiFi es necesario realizar ciertas conexiones en torno a cada uno de estos componentes. Respecto al primero, se deben conectar las alimentaciones, un pulsador de reset y un oscilador de cristal. Este último está compuesto por un cristal de 16MHz y dos capacitores cerámicos de 22pF, como se indica en la tabla 6-3 de la hoja de datos del microcontrolador [44]. El mismo es necesario para proveer al microcontrolador de una señal de reloj. En cuanto al módulo de comunicación WiFi, se debe realizar la conexión de la alimentación, el reset, el pin de habilitación y las líneas de transmisión y recepción.

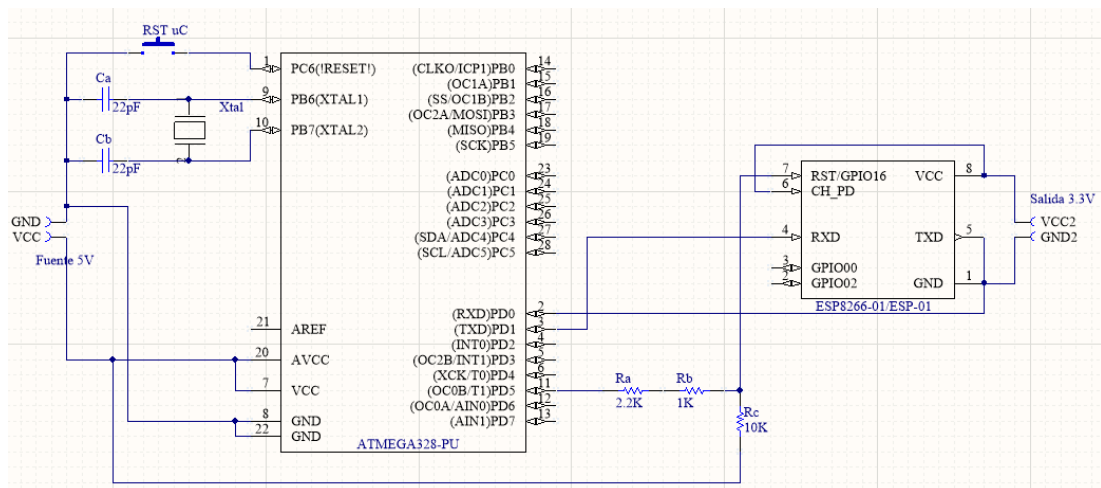


Figura 4.4: Esquema circuital de la etapa de lógica y comunicación.

En lo que respecta al conexionado de los periféricos que forman parte del sistema, se debe considerar el hecho de que algunos de ellos han sido simulados debido a una cuestión de costos o a que han quedado fuera del alcance del proyecto.

En cuanto a aquellos que se encuentran simulados, como es el caso de uno de los detectores magnéticos, las barreras vehiculares de entrada y salida y la expendedora de tickets del ingreso, la conexión con el microcontrolador es directa, mediante sus pines digitales de entrada. Mientras que la elevación de cada una de las barreras y la impresión del ticket se visualizan mediante un led, el segundo detector magnético y el retiro del ticket se simulan con pulsadores.

Entre los que se encuentran verdaderamente implementados, que son las barreras infrarrojas que forman parte del sistema de detección de tamaño y uno de los sensores magnéticos de presencia, la conexión con el microcontrolador se realiza a través de una adaptación de niveles de tensión, como la de la figura 4.5, para evitar dañar al mismo.

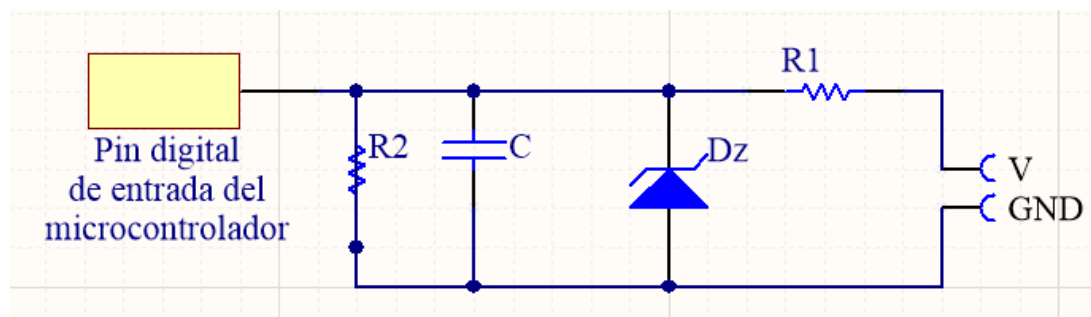


Figura 4.5: Adaptación de niveles de tensión entre los periféricos y el microcontrolador.

La función del diodo Zener es limitar la tensión máxima de entrada al pin digital a 5V, dado que en la hoja de datos se especifica que los mismos no admiten un valor superior a 5.5V. Esto se implementa debido a que los periféricos cuentan

con contactos de relé que, en caso de cerrarse, envían al pin del microcontrolador los 12V con que son alimentados. Además, en caso de una inversión de la tensión de entrada, el Zener protege al microcontrolador ya que solo le entrega su tensión en directa, que son 0.6V aproximadamente.

Por otra parte, R1 se encarga de limitar la corriente que circulará por la adaptación, de manera de proteger al diodo Zener. El capacitor cerámico C, junto con R1 y R2, conforman un filtro que minimiza los ruidos provenientes desde el exterior del sistema, incluyendo los rebotes provocados por los relés de los periféricos. La resistencia R2 se coloca en paralelo a la impedancia de entrada del pin del microcontrolador por dos razones. Primero, para obtener una impedancia de entrada fija y aproximadamente del valor de R2. Segundo, para disminuir el ruido que ingresa al circuito integrado, dado que la tensión de ruido depende de la magnitud de la impedancia de entrada. Entonces, al reducir esta última colocándole una resistencia en paralelo, se disminuye el ruido de entrada.

Finalmente, la placa además cuenta con leds indicadores que se encienden en caso de producirse alguna falla en el microcontrolador o el módulo de comunicación WiFi.

4.4. Diseño e implementación del PCB

En esta sección se hace alusión a aquellas cuestiones relacionadas con el diseño y la implementación del circuito impreso. Para ello, fue utilizada la herramienta Altium Designer 19.0.15 con una licencia gratuita para estudiantes, la cual tiene validez por seis meses.

Inicialmente, la Placa I/O fue implementada mediante una placa Arduino Mega. Esto se debió a que, en conjunto con docentes del Laboratorio de Comunicaciones (LAC), en el cual se desarrolló el proyecto, se determinó que la

misma es una herramienta adecuada para llevar a cabo el diseño del prototipo del sistema. Las razones que motivaron esta elección son la disponibilidad de una gran cantidad de información acerca del tema, tanto en libros como en la web, en forma gratuita, y a que en el laboratorio se cuenta con personal que posee experiencia trabajando con este modelo de placa. Por otra parte, el microcontrolador ATmega2560 que la misma posee permite implementar el sistema planteado.

El uso de esta placa de desarrollo permitió ensayar el funcionamiento de los códigos que se fueron realizando al inicio del proyecto. Posteriormente, la misma fue reemplazada por un circuito construido utilizando protoboards. El mismo fue implementado con el microcontrolador ATmega328P. Esto se debió a que, a pesar de haber estado trabajando con el ATmega2560 que es superior, éste resulta adecuado para desempeñar las funciones requeridas por el prototipo que se obtendrá al final de este proyecto.

Una vez que se verificó que el circuito desarrollado funcionaba en forma correcta, se procedió a realizar el diseño del PCB correspondiente. Durante el mismo, se creó la mayor parte de los encapsulados utilizados y, además, se recurrió a librerías disponibles en la página web oficial de Altium. Inicialmente se desarrolló un modelo esquemático, a partir del cual se obtuvo el circuito impreso del sistema. Con respecto a este último, fue necesario modificar algunas de las reglas de diseño establecidas por defecto, como es el caso del ancho de las pistas, la distancia entre ellas, y a distancia entre pistas y el plano de masa.

Con el objetivo de verificar si el ancho de las pistas elegido era adecuado, se recurrió a calculadoras de ancho de pista en función de la corriente y la sobreelevación de temperatura permitida. Debido a que la corriente de carga del sistema es baja, el ancho de pista de 0.7mm elegido resultó correcto.

Finalmente, la placa fue elaborada mediante el método de insolado. En un principio, se intentó fabricarla con la insoladora proporcionada por el LAC pero,

luego de varios intentos fallidos debidos a la baja calidad de impresión del fotolito y del film fotosensible disponible, se decidió enviarla a un productor de PCB local.

La placa construida se observa en la figura 4.6.

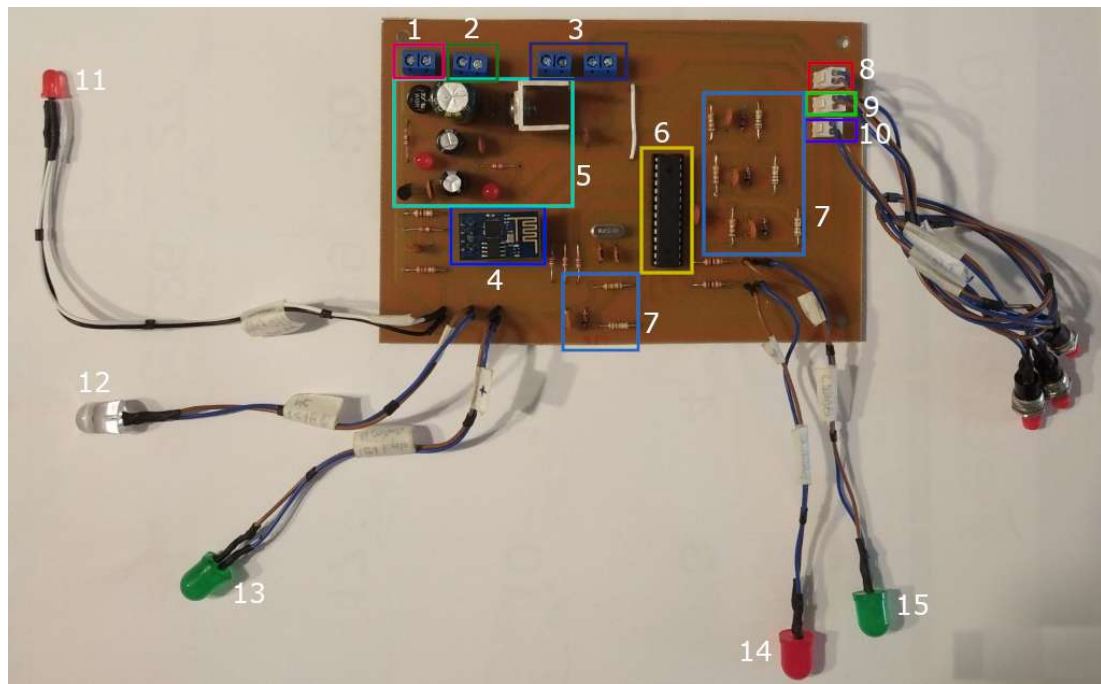


Figura 4.6: PCB generado a partir del diseño final.

Sobre la misma se ha indicado con recuadros de diversos colores las diferentes partes que la componen. Estas se detallan a continuación:

1. Entrada de 12V de alterna, proveniente desde el bobinado secundario del transformador.
2. Salida de 12V de continua destinada a la alimentación de los periféricos: las tres barreras infrarrojas y el detector magnético.
3. Entradas provenientes desde los periféricos. Observando desde la izquierda se tiene: barrera infrarroja 3, barrera infrarroja 2, detector magnético y

- barrera infrarroja 1.
- 4. Módulo de comunicación WiFi.
- 5. Fuente regulada de tensión continua.
- 6. Microcontrolador ATmega328P
- 7. Adaptaciones de nivel de tensión entre los periféricos y los pines del microcontrolador.
- 8. Pulsador de reset del microcontrolador.
- 9. Pulsador para simulación del detector magnético de salida.
- 10. Pulsador para simulación del retiro del ticket a la entrada.
- 11. Led indicador de falla del Módulo WiFi.
- 12. Led indicador de falla del Microcontrolador.
- 13. Led para simulación de impresión del ticket.
- 14. Led para simulación de barrera de entrada.
- 15. Led para simulación de barrera de salida.

4.5. Periféricos

En la presente sección, se realiza una breve descripción de los periféricos que forman parte del sistema, que fueron proporcionados por la empresa interesada en el desarrollo de este proyecto.

4.5.1. Barreras Infrarrojas

El sistema posee tres barreras infrarrojas que componen el sistema de detección de tamaño de los vehículos que desean ingresar a los establecimientos. Uno de esos equipos se observa en la figura 4.7.



Figura 4.7: Barrera infrarroja utilizada: receptor (izquierda) y transmisor (derecha).

Las características más relevantes de este modelo de barrera son las siguientes:

- Alcance $\leq 15\text{m}$
- Tensión de operación: DC/AC 12-24V
- Consumo de corriente del receptor: 15mA
- Consumo de corriente del transmisor: 30mA

Por otra parte, tanto el equipo transmisor como el receptor poseen dos jumpers J2 y J3. Estos se utilizan para determinar la frecuencia de trabajo de la barrera. A partir de los mismos, existen cuatro combinaciones correspondientes a cuatro frecuencias distintas. Esto permite que barreras cercanas trabajen a diferentes frecuencias y se evita que se produzcan interferencias entre ellas.

Adicionalmente, el receptor posee un jumper adicional J1 que se utiliza para determinar si el contacto disponible de su relé interno es el normal cerrado o el normal abierto.

En la figura 4.7, se observa el diagrama de conexiones del receptor (izquierda) y del transmisor (derecha). Mientras el segundo únicamente requiere tensión de alimentación, el primero también posee el común y la salida del relé interno. En el sistema, el común se encuentra conectado a la tensión de alimentación para que, en caso de activarse la barrera, a la salida del relé se tengan los 12V de entrada. Estos 12V son acondicionados mediante la adaptación de niveles para que al microcontrolador lleguen 5V.

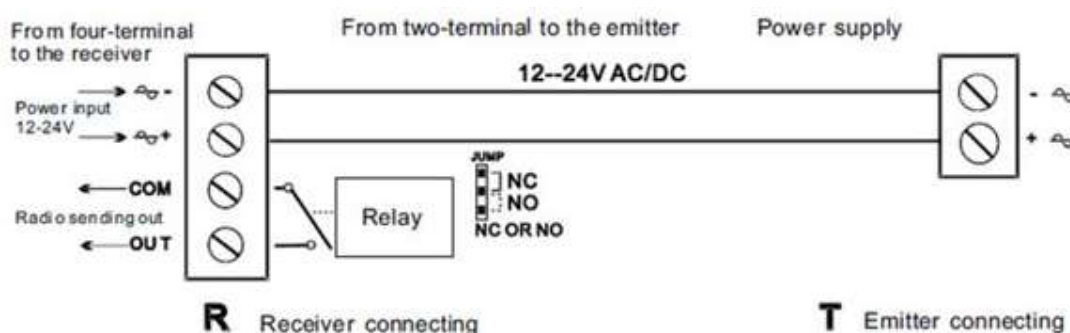


Figura 4.8: Conexiones de la barrera infrarroja: receptor (izquierda) y transmisor (derecha).

Finalmente, en lo que respecta al aseguramiento del correcto funcionamiento de este equipo, el mismo debe ser colocado al menos a 20cm de altura y con una distancia mayor a 1m entre receptor y transmisor.

4.5.2. Detector de presencia magnético

El sistema posee dos detectores de presencia magnéticos, aunque uno de ellos se encuentra simulado. Mientras que uno de ellos se ubica en el ingreso al establecimiento, el otro se encuentra en el egreso del mismo. Se lo puede observar en la figura 4.9.



Figura 4.9: Detector magnético utilizado.

Respecto al funcionamiento, al momento de darle alimentación, el detector se sintoniza automáticamente con el lazo inductivo que tiene conectado. El equipo requiere una tensión de alimentación de 12V DC/AC y **tiene un consumo de corriente de X (verificarlo al probarlo)**.

Por otra parte, el equipo posee dos indicadores leds que se encenderán según el estado en el que el mismo se encuentra. Cuando está sintonizado, se encienden el led rojo (power) y el verde (detect). Luego de dos segundos, el verde se apaga. Por su parte, mientras el equipo se encuentre alimentado, el led rojo se encuentra siempre encendido. El verde, va a parpadear en caso de falla o va a quedar encendido cuando se detecte un vehículo atravesando el lazo.

El diagrama de conexiones se muestra en la figura 4.10. Los pines 1 y 2 están destinados a la alimentación del detector. Los pines 5, 6 y 10 forman parte del relé de presencia del sistema. Entre 5 y 6 se tiene el contacto normal abierto y entre 6 y 10 el normal cerrado. Los pines 3, 4 y 11 forman parte del relé de salida pulsada. Entre 3 y 4 se tiene el contacto normal abierto y entre 4 y 11 el normal cerrado. Al momento de detectar la presencia de un vehículo, puede utilizarse tanto la salida del relé de presencia como el de pulso. Mientras que el primero es energizado mientras el vehículo se encuentra sobre el lazo, el segundo lo hace cuando el vehículo ingresa al lazo o cuando sale de él. Esto último es configurable. Finalmente, entre los pines 7 y 8 se conecta el lazo inductivo.

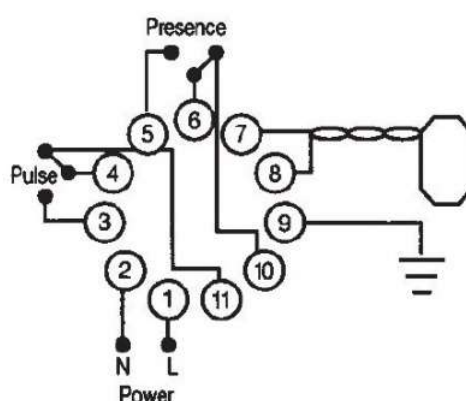


Figura 4.10: Conexiones de la barrera infrarroja: receptor (izquierda) y transmisor (derecha).

Como se observa en la figura 4.9, en el frente del detector hay 8 selectores que permiten configurar el equipo:

- **Ajuste de frecuencia:** corresponde a los primeros dos selectores. Hay cuatro combinaciones o frecuencias posibles. Esto permite que detectores que están ubicados cerca operen a diferentes frecuencias y no interfieran

entre sí.

- **Ajuste de sensibilidad:** corresponde a los siguientes tres selectores (3, 4 y 5). Hay ocho posibilidades, que hacen que el sensor sea más o menos selectivo al momento de realizar la detección.
- **Filtro:** permite eliminar las interferencias del entorno. Se lo activa seteando en ON el selector 6. Cuando se encuentra activado, el tiempo de reacción del detector aumenta y la sensibilidad se ve reducida. Es por esto que, a menos que el entorno lo requiera, no debe activarse.
- **Salida de relé:** corresponde al selector 7. Solo afecta a la salida del relé de pulso. Si se lo setea en OFF, el relé se energiza cuando un vehículo ingresa al lazo. En caso contrario, el mismo es energizado cuando el vehículo egresa del lazo.
- **Tiempo de presencia:** corresponde al selector 8. Permite determinar si el relé de presencia mantiene la presencia detectada hasta que el vehículo deja el lazo o si lo hace solo por diez minutos.

En caso de hacer alguna modificación en la configuración es necesario volver a sintonizar el detector. Para ello, puede utilizarse el botón de reset.

Respecto al lazo inductivo, el mismo debe realizarse con cable de $1,5mm^2$ de área y debe tener forma rectangular. La cantidad de vueltas que posee depende de las dimensiones con las que se lo construya. Basándose en el manual de usuario y en el tamaño de las vías de acceso de los establecimientos relevados, en este proyecto se consideró el uso de un lazo de 2.5m de largo por 1m de ancho, al que le corresponden cuatro vueltas de cable. Adicionalmente, los extremos del cable del lazo debe trenzarse, con al menos veinte vueltas por metro. Esto último

tiene por objetivo que el cable de conexión del lazo no forme parte del área de detección, evitando que participe de la misma.

En caso de que el detector no funcione correctamente, se debe proceder de la siguiente manera:

- Verificar el estado del lazo inductivo y del cableado
- Modificar la sensibilidad y/o la frecuencia
- Setear el filtro en ON

Capítulo 5

Prototipo desarrollado

5.1. Introducción

Como se mencionó previamente, el proyecto se llevó a cabo en forma incremental. Esta modalidad consiste en desarrollar núcleos pequeños que, una vez completados, evolucionan mediante el agregado de nuevas funcionalidades y el mejoramiento de aquellas que ya poseían.

Se realizaron dos prototipos. El primero de ellos se enfocó en lograr la lectura de la patente, la detección de la presencia del vehículo y la manipulación de una barrera, considerando una sola vía de acceso.

Una vez que este primer módulo se puso a punto, se mejoraron algunas de sus funcionalidades y se le añadieron nuevas, dando lugar al segundo prototipo.

5.2. Prototipo inicial

En la etapa inicial del proyecto se desarrolló un primer prototipo cuyo objetivo principal fue lograr la correcta obtención las patentes de los vehículos detectados. Por este motivo, en esta fase no se tuvieron en cuenta varios aspectos que se

presentan en un escenario real, como por ejemplo, la diferenciación entre vía de ingreso y de egreso.

Durante la realización de este prototipo se desarrollaron tres códigos en lenguaje C++, uno para la UCC, otro para el módulo WiFi y un tercero para la Placa I/O. El primero de ellos fue diseñado en el entorno de desarrollo Eclipse [46], el cual es una plataforma de software compuesto por un conjunto de herramientas de programación de código abierto y multiplataforma. Por otra parte, los otros dos códigos fueron desarrollados mediante el IDE (Integrated Development Environment) de Arduino [47], debido a que tanto el microcontrolador como el módulo WiFi utilizados son compatibles con el software que provee esta compañía.

A continuación, se detalla el funcionamiento de este primer prototipo. En primer lugar, se lleva a cabo una etapa de conexión entre el módulo WiFi, la Placa I/O y la UCC. En esta, al encender el sistema, la Placa I/O pone en alto el pin de reset del módulo WiFi produciendo el arranque del mismo.

Una vez encendido, el módulo procede a conectarse a la red WiFi local creada mediante un router y, luego, genera un servidor web en su dirección IP. A partir de este momento, la placa y el módulo quedan a la espera de que la UCC se conecte a dicho servidor. La primera acción que esta última lleva a cabo es la creación un socket TCP/IP, el cual utiliza para conectarse al servidor mencionado anteriormente. De esta manera, se concluye la etapa de conexión y el sistema queda listo para funcionar en modo continuo.

Cuando se presiona alguno de los dos pulsadores, acción que representa la detección de un vehículo mediante uno de los sensores magnéticos, el microcontrolador de la Placa I/O recibe dicha información y se la envía por puerto serie al módulo WiFi. Este último, comunica la información sobre el estado de los magnéticos a la UCC vía WiFi, únicamente si alguno de ellos fue activado.

Entonces, cuando a la UCC le llega esta información, a partir del procesamiento de los trabajos generados por el OpenALPR, se obtiene la patente del vehículo detectado.

En esta etapa, la activación de cualquiera de los pulsadores por separado, o de ambos simultáneamente, desemboca en la misma condición: la activación del sistema de reconocimiento de patentes sin discriminación entre entrada o salida. Esto se debe a que aún no se desarrolló esta distinción, ya que el principal objetivo de este primer prototipo es la verificación del correcto funcionamiento del reconocimiento. Luego de que se obtuvo la patente, la Unidad Central se la envía a la Placa I/O, de manera de indicarle que levante la barrera.

Cabe destacar que los tres códigos fueron desarrollados de manera secuencial. Esto implica que, si por ejemplo se activa algún detector magnético mientras se lleva a cabo un proceso de ingreso o egreso anterior, el sistema no podrá responder a dicho evento.

5.3. Prototipo final

En esta sección, se desarrollará el funcionamiento del prototipo final del proyecto. En este, la principal modificación fue la coexistencia de dos vías, una de ingreso y otra de egreso, funcionando simultáneamente. Para realizar esto se utilizó el software YAKINDU Statecharts Tools [48] con una licencia académica, la cual brinda acceso a todas sus funcionalidades seis meses. Además, su entorno de desarrollo está basado en Eclipse, lo que facilitó su uso, ya que dicho software fue implementado al realizar el prototipo inicial.

Este framework es capaz de generar bloques de código en diferentes lenguajes tales como C, C++ o Java, a partir de una máquina de estado conformada mediante diagramas UML (Unified Modeling Language). En la figura 5.1 se presenta

una tabla comparativa entre las máquinas de estado UML y otros tipos, donde se observan las ventajas que las primeras poseen sobre las demás.

	Mealy	Moore	Harel	UML
States and transitions	✓	✓	✓	✓
Transitions produce output	✓		✓	✓
States produce output		✓	✓	✓
Depth (hierarchies, composite states)			✓	✓
Orthogonality (parallel substatemachines)			✓	✓
Broadcast communication (events)			✓	✓
History, actions, delays, timeouts, conditions			✓	✓

Figura 5.1: Comparación entre diferentes máquinas de estado [49].

Los sistemas diseñados a partir de esta herramienta tienen dos formas de funcionamiento: basadas en ciclos o basadas en eventos [50]. Para cualquiera de los dos casos, un ciclo es el procedimiento mediante el cual el sistema analiza si debe pasar o no al siguiente estado, en cada una de las regiones que posea. La diferencia entre éstas dos formas de funcionamiento radica en el disparador del siguiente ciclo. En el primer caso, el sistema ejecuta su siguiente ciclo cada un cierto periodo de tiempo fijo, mientras que en la segunda forma, se espera a que algún evento haya ocurrido para ejecutar el ciclo. Debido a que en este proyecto se requiere que determinadas acciones sucedan a lo largo del ingreso o egreso para poder avanzar, se ha diseñado un sistema basado en eventos.

Por otra parte, mediante esta herramienta es posible generar sistemas basados en la técnica de la multiprogramación, es decir, la ejecución de tareas de manera concurrente por el procesador a tal velocidad que causa la impresión de ser en paralelo. Esto es de vital importancia para poder considerar a la entrada y

la salida como dos vías independientes, permitiendo efectuar ambos procesos simultáneamente. Además, al igual que el entorno de Eclipse, este software permite el desarrollo de proyectos de Arduino, que pueden ser cargados sobre las placas de dicha compañía. Debido a estas razones, tanto el código de la UCC como el de la Placa I/O fueron desarrollados mediante esta herramienta. Cabe destacar que el código correspondiente al módulo WiFi se generó con el IDE de Arduino, ya que no presentó gran complejidad debido a que únicamente es un intermediario entre la UCC y la placa. En la figura 5.2 se puede observar un diagrama en bloques representativo de la comunicación dentro del sistema.

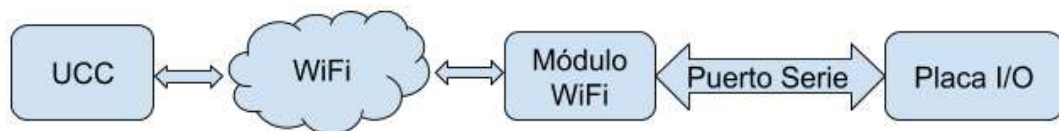


Figura 5.2: Comunicación entre UCC, Módulo y Placa.

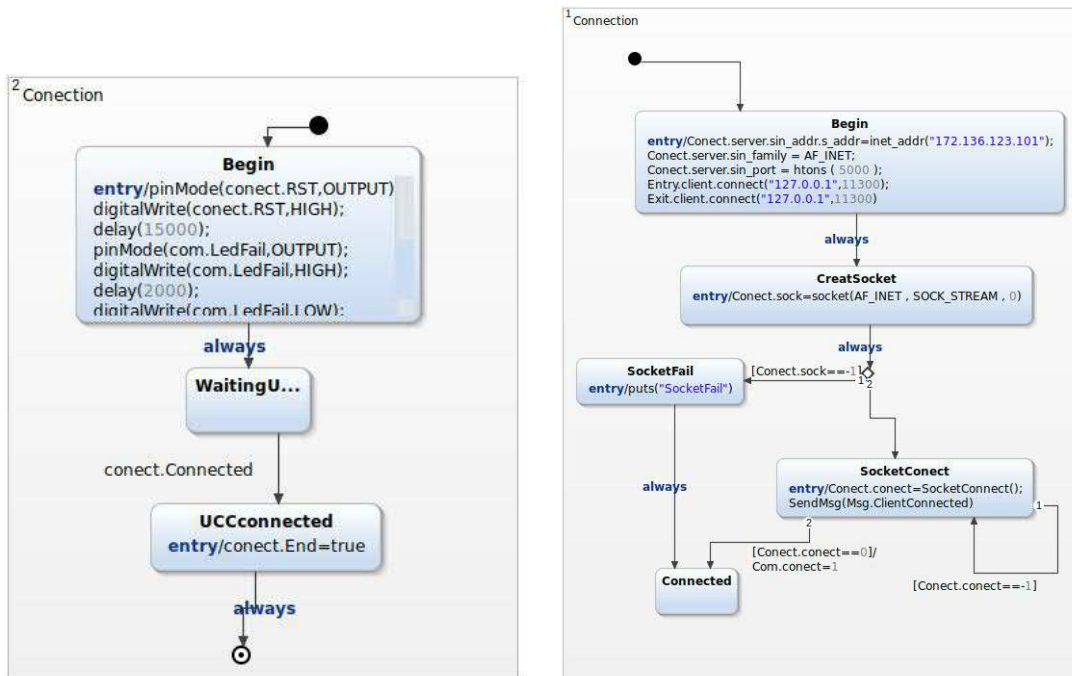
En las siguientes secciones de este capítulo se desarrollará el funcionamiento de cada una de las etapas del sistema. Se destaca que, a pesar de que su funcionamiento se explique en forma individual, las mismas trabajan en forma conjunta, a excepción de la fase de conexión.

5.3.1. Fase de conexión

En esta primera etapa, se busca establecer la conexión entre la UCC, el módulo WiFi y la Placa I/O. Esta fase ocurre solamente al iniciarlo el sistema.

Cada uno de los códigos desarrollados con el software YAKINDU posee una región específica para esta etapa, y las mismas se pueden observar en las figuras 5.3a y 5.3b. Por otra parte, en el módulo WiFi, esto sucede dentro del setup de

Arduino, como se ve en la figura 5.3c.



(a) Región de conexión de Placa I/O.

(b) Región de conexión de UCC.

```

48 void setup() {
49   Serial.begin(9600);
50   pinMode(LedFail, OUTPUT);
51   digitalWrite(LedFail, LOW);
52   pinMode(2, OUTPUT);
53   digitalWrite(2, LOW);
54   WiFi.mode(WIFI_STA);
55   WiFi.config(ip, gateway, subnet);
56   WiFi.begin(ssid, password);
57   unsigned long t = millis();
58   while (WiFi.status() != WL_CONNECTED) {
59     if ((millis() - t) > 10000) {
60       digitalWrite(LedFail, HIGH);
61       return;
62     }
63     delay(100);
64   }
65   server.begin();
66   Connect();
67   while (client.available() <= 0);
68   while (client.available() > 0) {
69     int msg = client.read();
70     NewMsg(msg);
71   }
72   timer = millis();
73 }

```

(c) Conexión del módulo WiFi.

Figura 5.3: Proceso de establecimiento de la conexión entre UCC, módulo WiFi y Placa I/O.

Al encender la Placa I/O, el microcontrolador pone en alto uno de sus pines digitales, el cual se encuentra conectado al pin de reset del módulo WiFi. De esta manera, el funcionamiento de este último queda controlado por la placa. Debido a que al iniciar el módulo el mismo escribe caracteres residuales en su puerto serie, la placa tiene un tiempo de espera de 15 segundos antes de inicializar dicha comunicación. De esta manera, se evita que estos caracteres afecten el funcionamiento normal de la placa. Luego de este período, esta última permanece a la espera de que se establezca la conexión por parte de la UCC.

Como se observa en la figura 5.3c, luego de iniciar, el módulo WiFi se conecta a la red WiFi que se le indica, crea un servidor web y queda a la espera de que la UCC se conecte al mismo. En caso de no poder conectarse a la red correspondiente, el módulo encenderá un led que indica la existencia de una falla.

Por su parte, la UCC debe conectarse al servidor creado. Para esto, luego de inicializar las variables necesarias, procede a crear un socket TCP/IP y conectarse a través de este al servidor. Es preciso mencionar que, en caso de falla en la creación o en la conexión del socket, el sistema informa dichos errores en pantalla.

Una vez establecida la conexión, la UCC le envía un mensaje al módulo WiFi indicándole este suceso, terminando su parte en la etapa de conexión. Al recibirlo, el módulo lo reenvía a la placa, poniendo fin a la etapa de conexión en su totalidad. De esta manera, el sistema queda preparado para su funcionamiento en forma continua. Se debe tener en cuenta que si se inicializa el código de la UCC en simultáneo con la placa, este mensaje que se envía se pierde. Esto se debe a los 15 segundos de demora establecidos en el microcontrolador para iniciar la comunicación serie. Por tal motivo, luego de este tiempo, la placa enciende durante dos segundos su led de falla, indicando que está preparada para la recepción de mensajes. Entonces, el código de la UCC debe ejecutarse luego de la ocurrencia de este evento.

5.3.2. Envío y recepción de mensajes

En los tres sistemas, al recibir un mensaje, este es contestado por el receptor con un mensaje de ACK para confirmar su llegada. Debe aclararse que los mensajes de ACK recibidos no se contestan. Por otra parte, al enviar un mensaje diferente de ACK, todos los sistemas deben recibir su respectivo mensaje de confirmación. De esta manera, se puede comprobar el arribo de todos los mensajes que se envían de un punto a otro. En caso de que un mensaje de ACK no se reciba en el plazo de 10 segundos establecido, se enciende un led de falla o se muestra un mensaje de error en pantalla, dependiendo de qué sistema no lo recibió. Ante una falla de este tipo, se determinó que el sistema deje de funcionar. A futuro, se implementará una solución que no implique reiniciar el sistema manualmente.

El envío y recepción de mensajes es la función principal del módulo WiFi, ya que se encarga de tomar los mensajes provenientes de un extremo y enviarlos al otro. Por lo tanto, el módulo no es más que un intermediario entre la placa y la UCC.

Por su parte, la Unidad Central posee una región especial que se encarga de comprobar si todos los ACK fueron recibidos correctamente. La misma se activa cada vez que en alguna parte del código se ejecuta la función de envío de mensajes. Esto puede observarse en la figura [5.4](#).

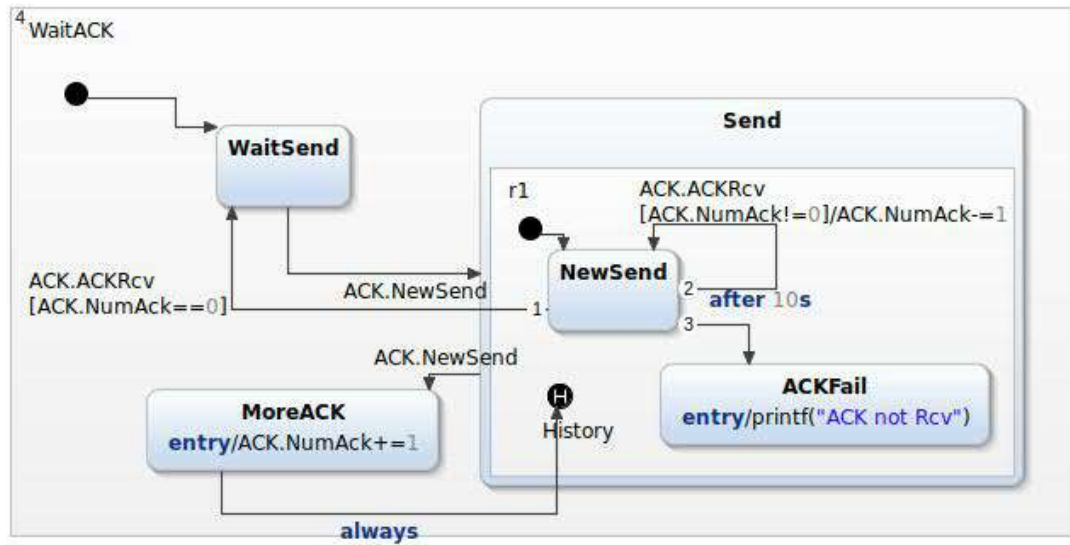


Figura 5.4: Control de ACK de la UCC.

En cuanto a la recepción de mensajes, la UCC se encuentra continuamente verificando si llegó alguno nuevo. En caso de que esto ocurra, informa en pantalla lo que recibió y procede a ejecutar las acciones que correspondan, ya sea modificar algún valor, levantar algún evento, entre otras.

Por otra parte, la Placa I/O posee dos regiones, una destinada a la recepción de mensajes y otra para el envío de los mismos. Estas pueden observarse en las figuras 5.5 y 5.6, respectivamente.

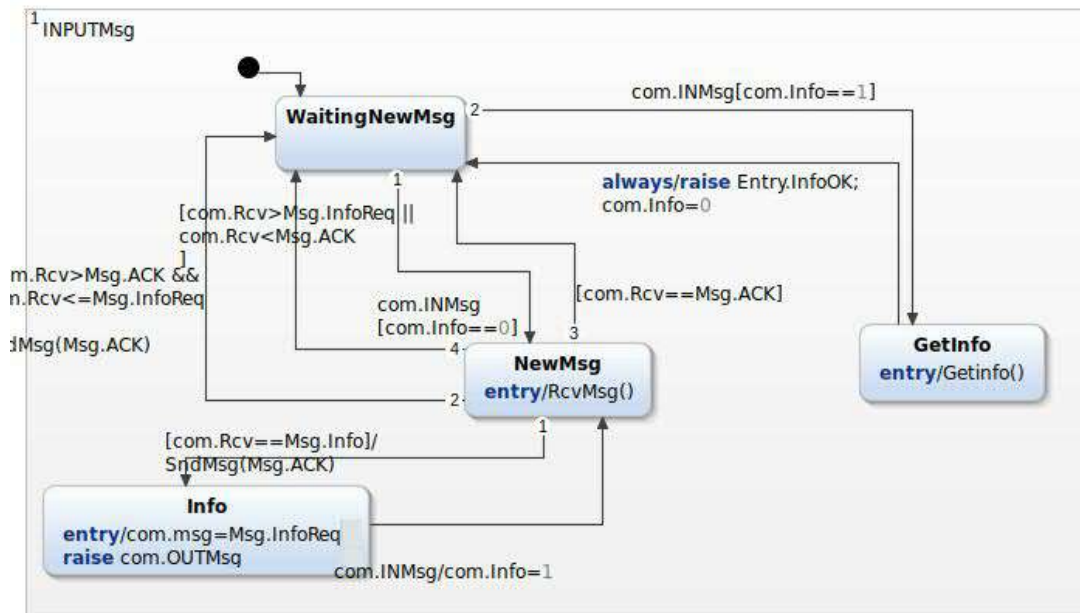


Figura 5.5: Recepción de mensajes desde Placa I/O.

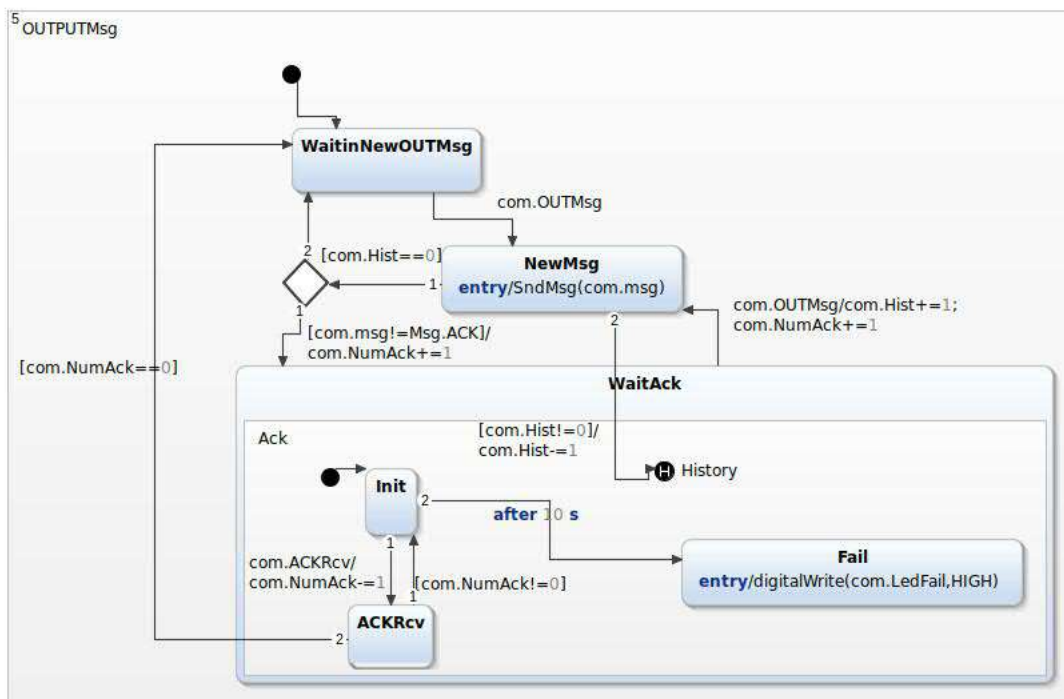


Figura 5.6: Envío de mensajes desde Placa I/O.

En la figura 5.5, se pueden observar las diferentes reacciones del sistema ante la recepción de un ACK, de la información del vehículo o cualquiera de los otros posibles mensajes. Por otra parte, en la región de envío de mensajes, se puede visualizar el estado WaitACK, donde el sistema comprueba el arribo de ACKs para los mensajes que haya enviado.

5.3.3. Vía de ingreso

Tanto la UCC como la Placa I/O disponen de una región destinada al funcionamiento de la vía de ingreso al establecimiento. Éstas dos se pueden observar en las figuras 5.7 y 5.8, respectivamente. Por su parte, durante esta etapa, el módulo WiFi solo envía los mensajes que recibe, de un extremo al otro.

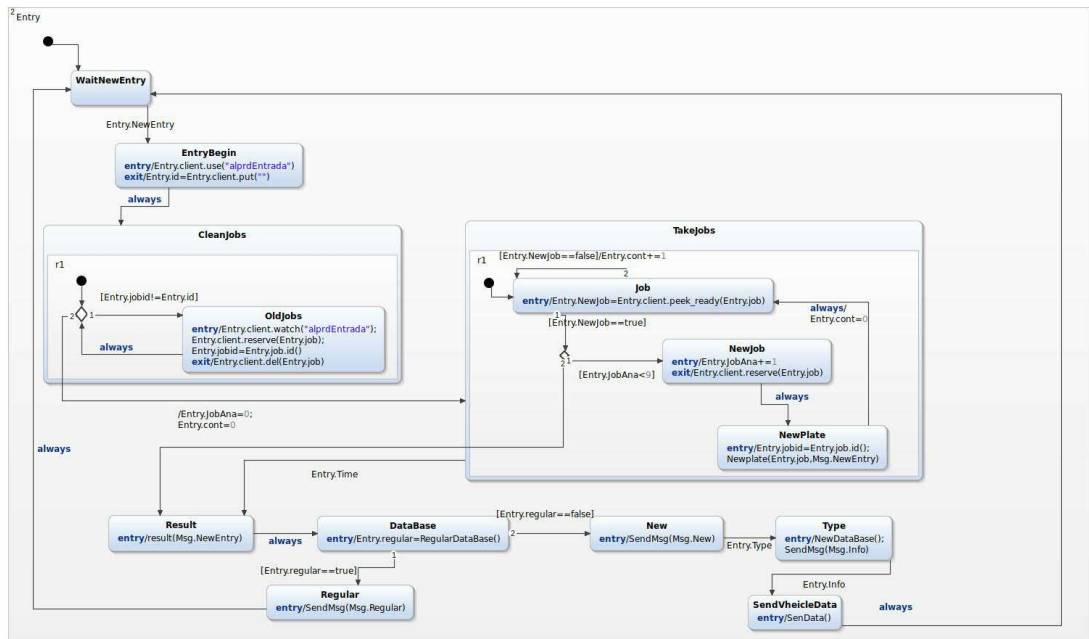


Figura 5.7: Procedimiento sobre la Vía de ingreso de la UCC.

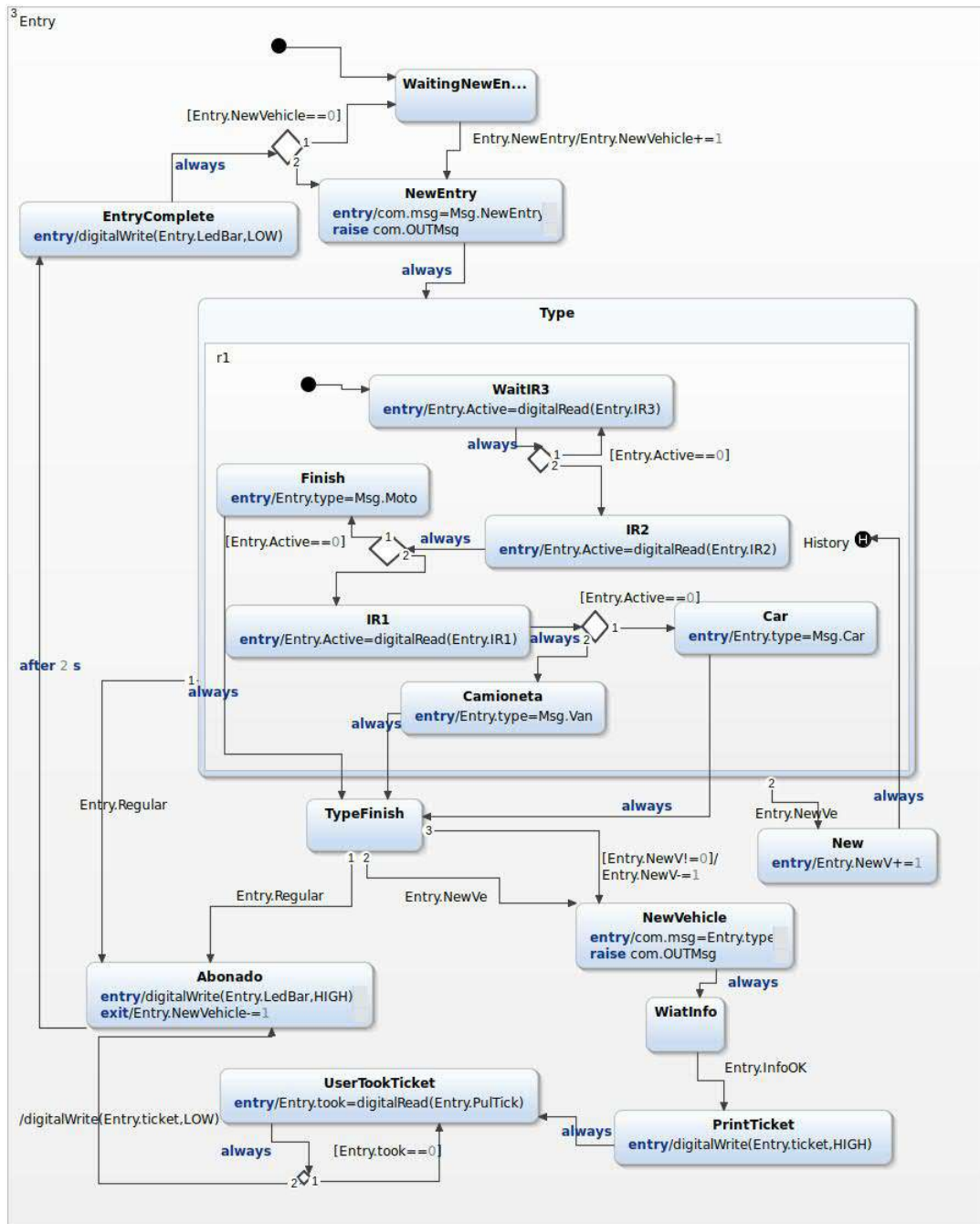


Figura 5.8: Procedimiento sobre la Vía de ingreso de la Placa I/O.

En la vía de ingreso, la placa se encuentra verificando continuamente el estado del detector magnético. Una vez que este se activa, comienza el funcionamiento de

la región mostrada en la figura 5.8. La primera acción de la misma es comunicarle a la UCC que un nuevo vehículo ha ingresado al estacionamiento. Al recibir esta información, la UCC levanta el evento “NewEntry” y procede con la fase de obtención de la patente.

Para obtener la matrícula, en primera instancia, la UCC inserta un trabajo bandera del cual se guarda el número de ID. Este se utiliza como guía para eliminar todos los trabajos anteriores acumulados en la cola, que poseen un número de ID inferior al mismo, evitando así que se efectúe una lectura errónea. Una vez que estos son eliminados, se procede a analizar los nuevos trabajos que el sistema va agregando a la cola hasta alcanzar una cifra de nueve, o bien, hasta que transcurran dos segundos. Luego, dado que se obtiene un resultado para la patente por cada trabajo analizado, la respuesta a devolver corresponderá a aquel que más veces se presente. Cabe destacar que, de esta manera, el sistema puede obtener diferentes posibles resultados para la misma patente. Por lo tanto, en caso de que se produzca un empate entre la cantidad de ocurrencias de dos (o más) resultados, se elige aquel que presente un valor de confianza mayor. Por otra parte, en caso de que no se logre analizar ningún trabajo, el sistema entrega como respuesta un número de siete cifras que representa un patente provisoria.

Con el resultado de la patente obtenido, la UCC debe verificar si el vehículo que desea ingresar al establecimiento es un cliente nuevo o un abonado. Para realizar esto se implementó una base de datos sencilla basada en MySQL. Este es un sistema de gestión de bases de datos desarrollado en C/C++ bajo licencia dual: Licencia pública general/Licencia comercial, por Oracle Corporation, y está considerada como la base de datos de código abierto más popular del mundo [51].

La base de datos creada consta de tres tablas independientes. En la primera de ellas, denominada “Regular”, se registran todos los vehículos abonados, es decir, aquellos que pagan una tarifa mensual. En la segunda, llamada “Parking”,

se almacenan todos los automóviles que hayan ingresado al establecimiento y aún no se hayan retirado. Esto implica que, con esta tabla, se puede saber la cantidad de vehículos estacionados dentro del estacionamiento. Por último, la tabla denominada “History”, lleva un historial de todos los vehículos que han ingresado y salido del establecimiento.

Entonces, para determinar si un vehículo es abonado o no, la UCC verifica si su patente se encuentra en la tabla “Regular”. La misma se observa en la figura 5.9. Luego, se ingresa la información de la patente, la fecha y hora de ingreso a la tabla “Parking”, la cual se ve en la [figura x10 \(poner antes de que se actualice el tipo\)](#). Por último, se informa a la placa de qué tipo de cliente se trata. En el caso de un abonado, además de la información mencionada, también se añade el tipo de vehículo y el estado del pago a dicha tabla.

Plate	Type	Pay	LastPay
NVZ087	Car	Yes	2020-02-10
SEW899	Van	Yes	2020-02-10
AA245UI	Car	Yes	2020-02-10

Figura 5.9: Tabla “Regular” de la base de datos desarrollada.

Mientras la placa espera que la UCC envíe la información acerca de la clase de cliente, calcula el tipo de vehículo que intenta ingresar: auto, moto o camioneta. Para ello, se implementa un sistema constituido por tres barreras infrarrojas, las cuales se encuentran dispuestas cada 2.5m. Se determinó que esta debía ser la distancia, a raíz de una investigación realizada sobre el tamaño máximo y mínimo que pueden tener los tres tipos de vehículos a detectar. En cuanto a su funcionamiento, el sistema analiza cuántas de las tres barreras son cortadas

simultáneamente por el vehículo que ingresa, cuando la más cercana de ellas a la barrera es activada. En caso de que los tres haces infrarrojos sean interrumpidos, se trata de una camioneta. Por otra parte, si solo los 2 sensores más próximos a la barrera se encuentran activados, es un auto. Finalmente, si no se activa más de una barrera a la vez, el vehículo es una motocicleta. De esta forma, se determina la tarifa que debe abonarse, ya que en muchos estacionamientos del país se cobra un monto diferenciado según el tipo de automóvil.

En el caso de un cliente abonado, luego de enviar la información mencionada anteriormente, la UCC vuelve a quedar a la espera de un nuevo ingreso. Por otra parte, la placa al recibir esta información, descarta el cálculo del tipo y el retiro del ticket, y le permite el ingreso al estacionamiento levantando la barrera. Esto se puede visualizar con el encendido de un led, durante dos segundos, que representa la barrera.

En el caso de un cliente no abonado, luego de informar su clase, la UCC espera recibir el tipo de vehículo para actualizar la tabla de la base de datos. Esto se observa en la **figura x11**.

Al recibir que se trata de un no abonado, la placa le informa a la UCC el “tipo” que determinó, y queda a la espera de la patente y la hora de ingreso. Estos datos se presentarían en el ticket que se le entrega a los usuarios, pero, en este caso, la impresión del mismo se encuentra simulada por el encendido de un led. Por lo tanto, al recibir la información del tipo, la UCC responde con estos datos, terminando su participación en el proceso de entrada. Al recibir dicha información, la placa enciende el led que indica que el ticket fue impreso.

Por último, para permitir el acceso al establecimiento, el usuario debe retirar el ticket. En este caso, esta acción se encuentra representada por la activación de un pulsador. Entonces, al presionarlo, se lleva a cabo el retiro del ticket, se apaga el led de ticket impreso y se enciende, durante dos segundos, el led que simula la

barrera. De esta manera, se concluye el ingreso del vehículo.

Debe tenerse en cuenta que, si se activa el detector magnético mientras está ingresando otro vehículo, esta condición se guarda. Por lo tanto, luego de que se apague el led de barrera, la placa no vuelve a su estado de espera, sino que comienza directamente con el proceso de ingreso pendiente.

Finalmente, en el caso de vehículos que no posean patentes, o que la misma no se haya podido obtener, el sistema igualmente permite el ingreso y se establece una patente provisoria, como se mencionó anteriormente.

5.3.4. Vía de egreso

De manera similar al caso de la vía de ingreso, tanto la Placa I/O como la UCC tienen una región destinada a la salida del vehículo, mientras que el módulo WiFi solo retransmite los mensajes. Ambas regiones pueden visualizarse en las figuras [5.10](#) y [5.11](#), respectivamente.

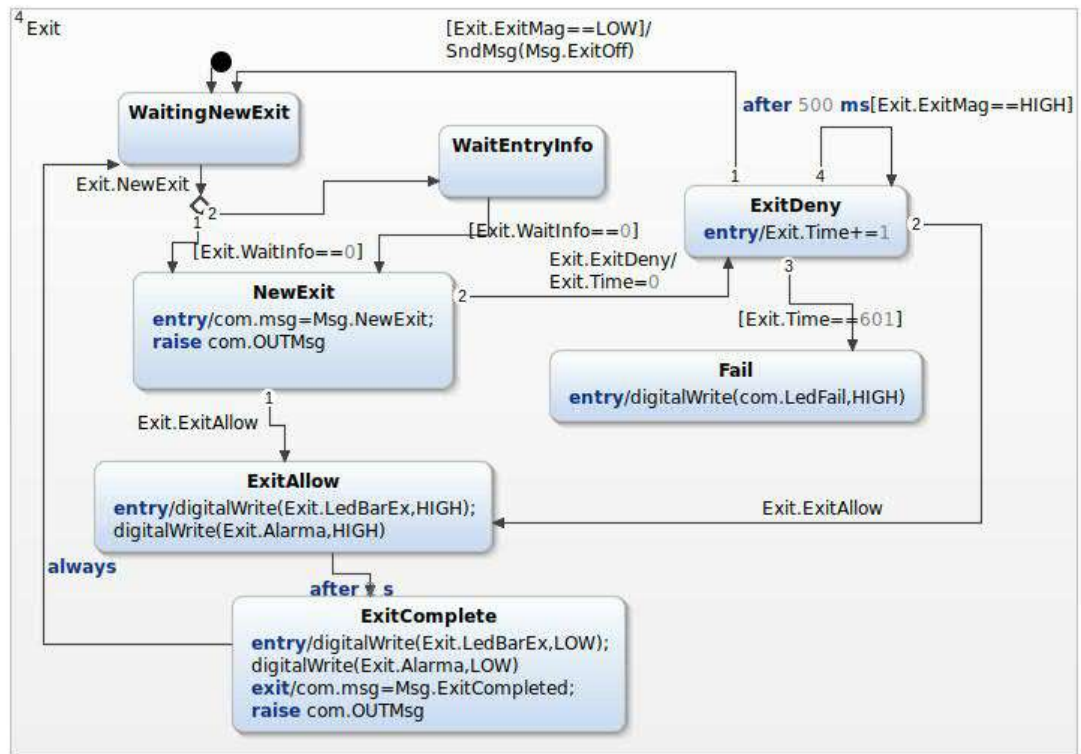


Figura 5.10: Etapa de egreso de la Placa I/O.

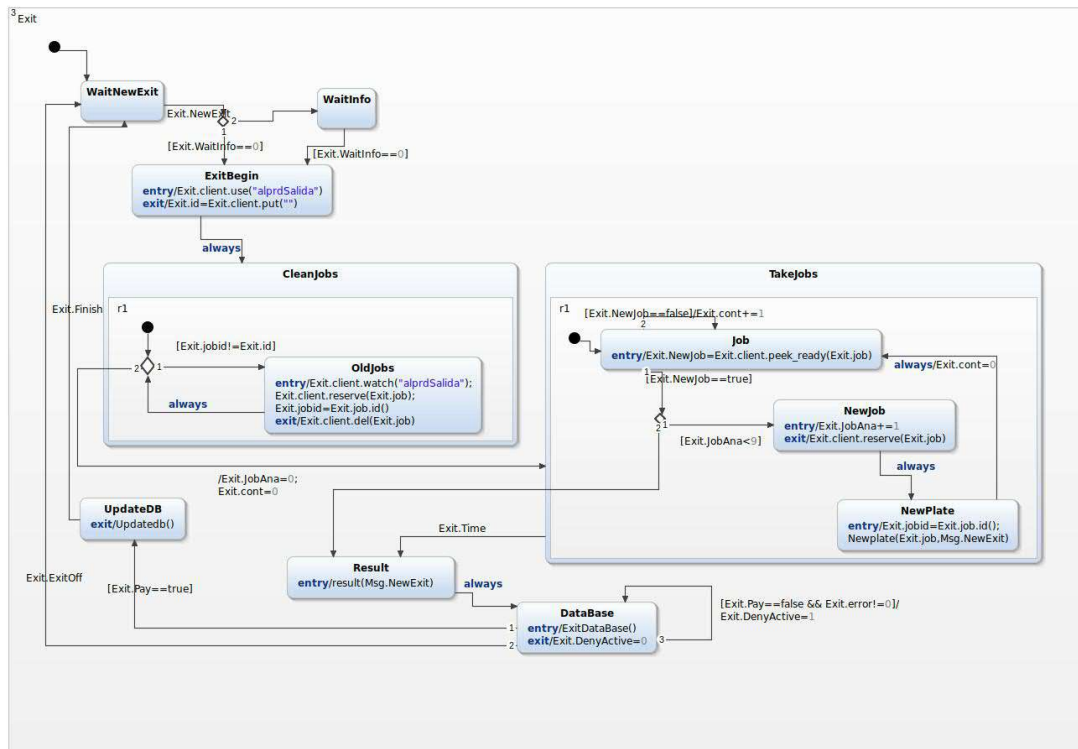


Figura 5.11: Etapa de egreso de la UCC.

Al igual que en la vía de ingreso, el sistema se encuentra a la espera de la activación del detector magnético. Cuando la Placa I/O verifica que un nuevo vehículo desea retirarse del establecimiento, lo informa a la UCC y se mantiene a la espera de la respuesta de ésta.

La única diferencia con la entrada es que en caso de no detectar ninguna patente, se establece el valor “NoPlate” como resultado.

Al recibir esta información, el proceso seguido por la UCC para obtener la patente del vehículo que activó el sistema es idéntico al de la entrada. La UCC debe verificar que la matrícula detectada se encuentre en la tabla “Parking”, es decir, que se comprueba que el vehículo haya ingresado. Por lo tanto, en este punto tenemos dos posibilidades:

1. **La patente se encuentra en la base de datos.** En este caso, se verifica

si el cliente ya realizó el pago de la tarifa. Con la finalidad de ensayar el prototipo, se diseñó un código que lo simula. En este, se le pide al operario que ingrese la patente, la tarifa y si el pago fue efectuado. De esta manera, se actualiza la información de la tarifa, el estado del pago y la fecha y hora de salida en la tabla “Parking”, de la base de datos. En la **figura x11**, se puede observar que dichos campos se encontraban vacíos, mientras que en la **figura x14** se ve que fueron actualizados. Por lo tanto, la UCC avisa a la placa que el egreso es permitido o denegado dependiendo de esta condición.

2. **La patente no se encuentra en la base de datos.** En este caso, se informa por pantalla al operario de lo sucedido, y este debe verificar si fue un error en la detección de la patente o no, visualizando el vehículo. En el primer caso, se le pide que ingrese la patente correcta y el sistema la vuelve a buscar en la base de datos, llevando a cualquiera de éstas posibilidades nuevamente. Se destaca que, en el caso de no poder detectar correctamente la patente, si a la entrada se asignó una provisoria, se debe ingresar ese valor. En el segundo caso, se avisa a la placa que el egreso fue denegado.

Por lo tanto, luego de verificar en la base de datos, la UCC informa a la placa sobre si la salida es autorizada o no. Esto lleva a dos casos posibles:

1. **Salida autorizada.** En este caso, la placa enciende durante dos segundos el led que simula la barrera de salida, permitiendo el egreso del vehículo. Una vez que se completa el egreso, la placa se lo informa a la UCC, terminando su parte en el proceso y quedando a la espera de un nuevo egreso. De esta manera, al recibir esta información, la UCC elimina al vehículo de la tabla “Parking” y lo agrega a la tabla “History”. Esto se puede observar en las **figuras x15 y x16**, respectivamente. Luego de esta actualización, el

proceso actual finaliza y la UCC queda preparada para el arribo de un nuevo vehículo.

2. **Salida Denegada.** Luego de enviar esta información, la UCC se mantiene comprobando de manera continua el estado del pago de la tarifa. Esto se debe a que en el momento en que el cliente realice esta acción, se le autorizará el egreso. En caso de que la salida haya sido denegada por no encontrarse la patente en la base de datos, el sistema se queda a la espera de que el vehículo se retire de la vía de egreso, retrocediendo hacia el interior del establecimiento. Simultáneamente, la placa se encarga de analizar si el vehículo sigue posicionado sobre el detector magnético. En caso afirmativo, el cliente tiene cinco minutos para realizar el pago, si aún no lo hizo, o para retirar el vehículo de la vía de egreso. En la primera situación, al actualizarse el pago, la UCC lo verifica y se procede como en el caso de la salida autorizada. Por otra parte, si el cliente retira el vehículo de la salida, esto se informa a la UCC y el sistema finaliza el proceso, quedando preparado para recibir un nuevo vehículo.

Cabe destacar que, en este caso, no existe la posibilidad de que un segundo vehículo active el detector magnético mientras otro realiza el ingreso. Esto se debe a que el mismo se ubica muy próximo a la barrera, por lo cual va a mantenerse activado por el mismo vehículo hasta el momento en que se retire.

Capítulo 6

Conclusiones y trabajos futuros

6.1. Conclusiones

A lo largo de este documento se ha explicado la investigación, desarrollo e implementación del sistema SAE para automatizar playas de estacionamiento. Se decidió trabajar de manera incremental. Por lo tanto, se desarrolló un prototipo inicial, el cual fue progresando a medida que se avanzaba en el proyecto, debido al agregado de nuevas funcionalidades y la mejora de las ya existentes, hasta obtener un prototipo final. Mientras en el inicial se hizo hincapié en el funcionamiento de algunas etapas de gran importancia y complejidad, como es el caso de la detección de patentes, el final se enfocó en la parte de hardware.

En el primer capítulo, se desarrolló el sistema como un producto comercial, tal como fue pensado en conjunto con la empresa que apoya este proyecto. Luego, se explicó su funcionamiento, tanto para el caso del ingreso como para el del egreso. Además, junto con los directores del trabajo final, con el objetivo de limitar el tiempo de desarrollo del proyecto y su nivel de complejidad, se estableció el alcance del mismo. De esta manera, se dejaron fuera del mismo la entrega de los tickets al ingreso y la implementación de una pantalla para la asignación de lote,

entre otras. Esto se trata con más detalle en la sección siguiente.

Por otra parte, para determinar la ubicación más adecuada de los periféricos dentro de un posible estacionamiento, se llevó a cabo un relevamiento. El mismo se abocó a las características de los establecimientos que se encuentran en la zona céntrica de la ciudad de Mar del Plata. Una posible distribución de los mismos se observa en el esquema de la figura 2.1.

Debido a que el reconocimiento de las patentes es un punto crítico de este proyecto, se ha llevado a cabo una investigación exhaustiva en torno al mismo. Mediante esta, se pudieron observar los diferentes tipos de sistemas ALPR existentes, su funcionamiento general y las ventajas y desventajas que presentan. Al tratarse de una etapa crucial, se tomó la decisión de analizar dos alternativas diferentes (OpenALPR y OpenCV 3 license plate recognition), para luego optar por una de ellas. Para ambos sistemas, se llevó a cabo un análisis en detalle de su funcionamiento, se experimentó con ellos y se realizó su ajuste en base a dicha experimentación. Para ello, fue necesaria la construcción de diversos conjuntos de imágenes. A partir de los resultados obtenidos, que fueron presentados en el Capítulo 3, se decidió continuar trabajando con el software OpenALPR. Esto se debe, principalmente, a que con esta herramienta, al reconocer la matrícula, se obtiene un porcentaje de éxito mucho mayor y en menor tiempo que con la otra alternativa. Además, en este capítulo se muestra el relevamiento hecho sobre las características de las matrículas argentinas, considerando el modelo antiguo y el del MERCOSUR.

El otro punto crucial del sistema es el desarrollo del hardware necesario. En el Capítulo 4 se presentó el diseño y construcción de la placa encargada de controlar los equipos que se ubicarían en las zonas de ingreso y egreso. Se realizó la muestra de los esquemas circuitales correspondientes a la fuente de alimentación del

sistema y la parte lógica de la placa, junto con la justificación de las diversas decisiones que se tomaron durante el diseño. Además, se realizó una breve descripción de los elementos principales del PCB, que son el microcontrolador ATmega328P y el módulo de comunicación WiFi ESP-01, y de los periféricos que integran el sistema, que son el detector vehicular magnético y las barreras infrarrojas. Finalmente, se presentó el desarrollo e implementación del circuito impreso. Para la realización del mismo, fue necesario instruirse en el uso de Altium Designer y en el método de insolado para la fabricación de placas.

El funcionamiento del sistema se basa en tres códigos diferentes, que fueron desarrollados completamente, y se encuentran distribuidos en la UCC, la Placa I/O y el Módulo WiFi. Estos fueron diseñados con el objetivo de implementar la multiprogramación. Mediante esta técnica, la ejecución de tareas por parte del procesador se da a una velocidad que causa la impresión de que ocurren en paralelo. De esta manera, se le permitió al sistema SAE poder diferenciar entre las vías de ingreso y egreso, es decir, que puede realizar el proceso de entrada y de salida en "simultáneo". Por otra parte, se añadieron al sistema funcionalidades para responder ante algunos casos particulares, como por ejemplo, un vehículo que pretende retirarse sin efectuar el pago, uno que se retire de la vía de egreso hacia el interior del establecimiento, la obtención de la matrícula de un vehículo que desea egresar y que no se encuentra en la base de datos, entre otros.

Por último, deben destacarse las características distintivas que posee el prototipo obtenido. Las mismas se enumeran a continuación:

- Los sistemas de automatización investigados, tanto los implementados en los establecimientos de nuestra ciudad como los ofrecidos por empresas radicadas en la ciudad de Buenos Aires, no poseen un sistema de reconocimiento de patentes ni un sistema de detección de tamaño estandarizado.

- No se encontraron fabricantes de este tipo de sistemas en Mar del Plata. El desarrollo de un sistema en la ciudad permite abastecer a los establecimientos locales y proporcionarles un servicio de mantenimiento (actualmente dependen de empresas radicadas en Bs.As.).
- El prototipo se encuentra desarrollado completamente mediante software libre. Esto evita el pago de licencias y, en el caso del software de reconocimiento de patentes, el pago de un servicio mensual.

6.2. Trabajos futuros

El objetivo de esta sección es dejar establecidas aquellas ideas o mejoras que pueden ser implementadas en un futuro en el sistema desarrollado. Esto se debe a que, al tratarse de un proyecto de finalización de una carrera universitaria, el mismo tiene un período acotado de duración, el cual hace que muchas de estas ideas deban mantenerse fuera del mismo.

6.2.1. Mejoras

- Realizar el entrenamiento del motor de OCR específico para las matrículas argentinas
- Desarrollo de una base de datos más compleja, que ofrezca más opciones al operador del sistema
- Diseño de un sistema de reseteo remoto en caso de falla en el sistema
- Implementación de un sistema de respaldo en caso de corte de luz
- Diseño y construcción de una terminal que integre la UCC, en formato All in One, junto con el lector de código de barra y la impresora de tickets

fiscales

6.2.2. Nuevas funcionalidades

- Agregado de impresora de tickets internos al estacionamiento, con información de hora de ingreso, número de patente y tarifa, entre otras, con su correspondiente lector de código de barra
- Agregado de impresora de tickets fiscales
- Desarrollo de una interfaz de usuario que permita el ingreso de información por parte del operario en caso de que ocurra algún inconveniente
- Desarrollo de un algoritmo de asignación de ubicación dentro del establecimiento y verificación del correcto estacionamiento
- Agregado de pantalla para mostrar el resultado de la asignación de ubicación al cliente
- Automatización del cobro de la tarifa a los clientes
- Adaptación del sistema para su aplicación en cocheras de edificios comerciales y privados, control de ingreso y egreso a barrios cerrados, etc
- Detección de rostros al ingreso y egreso del establecimiento, de manera de incrementar la seguridad en cuanto al retiro de los vehículos

Capítulo 7

COSAS QUE LUEGO DEBEN SACARSE DEL ESCRITO

7.1. Cosas a CORREGIR del Escrito 22-02-2020

- En cada capítulo poner al principio un resumen de lo que se va a tratar en el mismo. Poner algo similar a lo que se escribió sobre dicho capítulo al final de la Introducción general del trabajo.
- Poner texto más explicativo en las figuras y las tablas. Que leyendo el epígrafe se pueda entender de qué se trata la foto. Por ejemplo, “imagen original” o “imagen binarizada” dicen poco o nada.
- Verificar que las figuras y tablas tengan todos nombres diferentes, no repetir.
- Inventar un nombre al sistema con una sigla asociada, para que sea más fácil referirse al mismo, sin tener que llamarlo siempre “Sistema de estacionamiento automatizado” (Sistema Automatizado de Estacionamiento -SAE-).

- Al esquema que hice en 3D del estacionamiento (está en el cap uno de estructura) agregar algún grafico similar al de la página web china (q muestra entrada y salida, en la Solution 1)
- Ver si usamos la palabra “Núcleo” para referirnos a las 2 partes más importantes de la tesis, o buscamos una mejor.
- Cambiar nombre a la placa (Placa I/O es muy cabeza).
- ¿AGREGAR?: El proyecto se escribió utilizando las herramientas TexStudio versión 2.12.14 (editor de Latex) y la distribución de Latex MiKTeX 2.9.

7.2. Aspectos aprendidos durante el desarrollo del Proyecto Final -no solo poner cosas técnicas

- Cómo documentar
- Interacción con personal, al realizar el relevamientos de los estacionamientos
- Manejo y configuración de un sistema tipo caja negra
- A trabajar en equipo
- Primeros pasos en C++
- Altium
- Latex
- Etc

Bibliografía

- [1] Mauro Osorio. La flota circulante de argentina creció 30 % en 6 años. url <http://www.carsdrive.com.ar/noticias/novedades/la-flota-circulante-de-argentina-crecio-30-en-6-anos/>, 2018. Online; último acceso: 06 Agosto 2019.
- [2] Leandro Murciego. Cocheras: todo lo que hay que saber antes de invertir. url <https://www.lanacion.com.ar/propiedades/cocheras-todo-lo-hay-saber-antes-invertir-nid2224897>, 2019. Online; último acceso: 06 Agosto 2019.
- [3] KinouWell. Off street parking solution. url <https://www.kinouwellparking.com/solutions/off-street-parking-system/>. Online; último acceso: 22 febrero 2020.
- [4] Eugenia Bahit. Scrum y extreme programming para programadores, 2011.
- [5] Wingu. Manual de metodologías Ágiles. url <https://www.winguweb.org/capacitacion/manual-de-metodologias-agiles>, 2016. Online; último acceso: 06 Agosto 2019.
- [6] Ignacio Moretti, Javier Jorge, José Amado, Cristian Caniglia, y Daniel Puntillo. Software libre para reconocimiento automático de las nuevas patentes del mercosur. 08 2016.

- [7] Chirag Patel, D. Shah, y Atul Patel. Automatic number plate recognition system (anpr): A survey. *International Journal of Computer Applications (IJCA)*, 69, 05 2013. doi: 10.5120/11871-7665.
- [8] G. De Giovanni J.C. Cuttitta y A.N. Campos. Reconocimiento automático de número de patente, 2011.
- [9] S. Du, M. Ibrahim, M. Shehata, y W. Badawy. Automatic license plate recognition (alpr): A state-of-the-art review. *IEEE Transactions on Circuits and Systems for Video Technology*, 23(2):311–325, Feb 2013. ISSN 1558-2205. doi: 10.1109/TCSVT.2012.2203741.
- [10] Andrew Agbemenu, Jephthah Yankey, y Ernest O. An automatic number plate recognition system using opencv and tesseract ocr engine. *International Journal of Computer Applications*, 180:1–5, 05 2018. doi: 10.5120/ijca2018917150.
- [11] Pips technology alpr. url <http://www.pipstechnology.com/es/soluciones-alpr/>. Online; último acceso: 09 marzo 2020.
- [12] Genetec alpr. url <https://www.genetec.com/es/soluciones/productos/autovu>. Online; último acceso: 09 marzo 2020.
- [13] Vpar – librería de reconocimiento de patentes. url http://www.neurallabs.net/es_AR/sistemas-ocr/lector-matriculas. Online; último acceso: 09 marzo 2020.
- [14] Especificaciones lpr visart-visiondepates. url <http://visart.com.ar/especificaciones-lpr-anpr/>. Online; último acceso: 09 marzo 2020.

- [15] Bastián Nicolás Carvajal Ahumada. Reconocimiento fotográfico de patentes. 03 2018.
- [16] Christian Wolf y Jean-Michel Jolion. Extraction and recognition of artificial text in multimedia documents. *Formal Pattern Analysis & Applications*, 6: 309–326, 02 2004. doi: 10.1007/s10044-003-0197-7.
- [17] Jamileh Yousefi. Image binarization using otsu thresholding algorithm, 05 2015.
- [18] Jorge Navacerrada. Sistema de detección de matrículas con open cv. 10 2017.
- [19] Tesseract ocr. url <https://opensource.google/projects/tesseract>. Online; último acceso: 07 febrero 2020.
- [20] Reglamentacion patente unica del mercosur, 08 2014.
- [21] Las patentes se renuevan. url <https://tinyurl.com/rqau287>, . Online; último acceso: 02 octubre 2019.
- [22] 1995-2016: todas iguales. url <https://tinyurl.com/v9zqcsz>, . Online; último acceso: 02 octubre 2019.
- [23] Detectan fallas de fabricación en las nuevas patentes y recomiendan cambiarlas. url <https://tinyurl.com/w4e8479>. Online; último acceso: 08 febrero 2020.
- [24] ANYLINE. Your smartphone as license plate scanner. url <https://anyline.com/products/scan-license-plates/>. Online; último acceso: 02 febrero 2020.

- [25] PLATE RECONGNIZER. Accurate, fast, developer-friendly anpr. url <https://platerecognizer.com/>. Online; último acceso: 02 febrero 2020.
- [26] OPENALPR. Automatic license plate recognition. url <http://www.openalpr.com/>. Online; último acceso: 02 febrero 2020.
- [27] Chris Dhams. Opencv 3 license plate recognition. url <https://tinyurl.com/qplgw4q>, . Online; último acceso: 02 febrero 2020.
- [28] Open alpr. url <https://github.com/openalpr/openalpr>, . Online; último acceso: 02 febrero 2020.
- [29] Openalpr documentation. url <http://doc.openalpr.com/index.html>, . Online; último acceso: 07 febrero 2020.
- [30] Open alpr. url <https://opencv.org/>. Online; último acceso: 07 febrero 2020.
- [31] Developers guide. url <https://tinyurl.com/rsrlmc5>. Online; último acceso: 07 febrero 2020.
- [32] Enrique Ramón-Balmaseda. Transformaciones basadas en el algoritmo local binary pattern de imágenes capturadas con la kinect para clasificación facial. 12 2011.
- [33] Lbp y ulbp – local binary patterns y uniform local binary patterns. url <https://tinyurl.com/txfflkq>. Online; último acceso: 09 marzo 2020.
- [34] Jaakko Sauvola, Tapio Seppänen, Sami Haapakoski, y Matti Pietikäinen. Adaptive document binarization. volume 33, pages 147–152 vol.1, 09 1997. ISBN 0-8186-7898-4. doi: 10.1109/ICDAR.1997.619831.
- [35] Edgar Molina, Julia Diaz, Hugo Hidalgo-Silva, y Edgar Chávez. Algoritmos de binarización robusta de imágenes con iluminación no uniforme. *Revista*

- Iberoamericana de Automática e Informática industrial*, 11 2017. doi: 10.4995/riai.2017.8847.
- [36] Rafael Muñoz Magro. Binarización de imágenes digitales y su algoritmia como herramienta aplicada a la ilustración entomológica. 2013.
- [37] Valentín Tomás Rubio. Sistema de reconocimiento de matrículas y gestión de acceso en una plataforma embebida. 06 2015.
- [38] Beanstalk protocol. url [https://github.com/beanstalkd/beanstalkd/blob/master/doc/prot](https://github.com/beanstalkd/beanstalkd/blob/master/doc/protocol.txt). Online; último acceso: 10 marzo 2020.
- [39] Beanstalk client. url <https://github.com/deepfried/beanstalk-client>, . Online; último acceso: 10 marzo 2020.
- [40] Jsoncons. url <https://github.com/danielaparker/jsoncons/blob/master/README.md>. Online; último acceso: 10 marzo 2020.
- [41] Openalpr daemon (alprd). url [https://github.com/openalpr/openalpr/wiki/OpenALPR-Daemon-\(alprd\)](https://github.com/openalpr/openalpr/wiki/OpenALPR-Daemon-(alprd)). Online; último acceso: 10 marzo 2020.
- [42] Alberto Arce Arroyo. Visión artificial de documentos., 07 2016.
- [43] Chris Dham. Opencv 3 knn character recognition. url <https://tinyurl.com/tgzd2ww>, . Online; último acceso: 07 febrero 2020.
- [44] Atmega 328 p. url <https://www.alldatasheet.com/datasheet-pdf/pdf/313656/ATMEL/ATmega328P.html>. Online; último acceso: 10 febrero 2020.
- [45] Esp-01 wifi module. url <http://www.microchip.ua/wireless/esp01.pdf>. Online; último acceso: 10 febrero 2020.

- [46] The platform for open innovation and collaboration. url <https://www.eclipse.org/>. Online; último acceso: 18 marzo 2020.
- [47] Arduino. url <https://www.arduino.cc/>. Online; último acceso: 18 marzo 2020.
- [48] Yakindu statechart tools use the power of state machines. url <https://www.itemis.com/en/yakindu/state-machine/>, . Online; último acceso: 18 marzo 2020.
- [49] What is a state machine? url <https://tinyurl.com/wjwo2ro>, . Online; último acceso: 18 marzo 2020.
- [50] Event-driven vs. cycle-based execution. url <https://tinyurl.com/wlt6qrh>, . Online; último acceso: 18 marzo 2020.
- [51] Oracle mysql: The most popular open source database. url <http://www.oracle.com/us/products/mysql/overview/index.html>. Online; último acceso: 20 marzo 2020.