



**UNIVERSIDAD DEL AZUAY**  
**FACULTAD DE CIENCIAS DE LA ADMINISTRACIÓN**  
**ESCUELA DE INGENIERÍA DE SISTEMAS**

**Sistema para registro de ingreso de vehículos a la Universidad del Azuay  
mediante el reconocimiento de placas**

**Tesis previa a la obtención del título de:  
Ingeniero de Sistemas**

**Autores:**

**Fabián Mauricio Atariguana Quezada  
Pedro Andrés Ávila Regalado**

**Director**

**Ing. Marcos Orellana Cordero**

**Cuenca, Ecuador  
2015**

## **DEDICATORIA**

Dedico esta tesis a mis padres, que siempre estuvieron a mi lado apoyándome en momentos buenos y malos, quienes se esforzaron para darme una buena educación y permitirme ser un buen profesional.

Pedro Andrés Avila Regalado

## DEDICATORIA

A Dios que me bendice y me permite lograr mis metas dirigiendo mi progreso día a día.

A mis padres, que son una bendición para mí, siempre incondicionales, que con su sabiduría y fortaleza supieron aportar a mi desarrollo.

A toda mi familia que de una u otra forma siempre estaban aportando a mi futuro.

A mi novia, Elizabeth por ser una de las impulsadoras de mi futuro y mi inspiración en todo momento.

Fabián Mauricio Atariguana Quezada

## AGRADECIMIENTO

Agradezco primeramente a Dios, por ser siempre la guía en vida, mostrándome siempre el camino que debo seguir para ser una persona de bien.

A mis padres que fueron el soporte para formarme como un buen profesional, brindándome sus enseñanzas y conocimientos.

A mis familiares quienes me otorgaron su apoyo y sus consejos durante mi vida académica.

A Dios que me guía y bendice cada día.

A mis padres cuyo valor es invaluable y su aporte incondicional para mí, a pesar de los momentos difíciles, supieron luchar contra todo y ser pilares fundamentales en mí crecer como persona y como profesional.

A toda mi familia que supo extender su mano y brindarme su apoyo incondicional.

Pedro Andrés Avila Regalado

Fabián Mauricio Atariguana Quezada

También queremos agradecer a la Universidad del Azuay por permitirnos demostrar nuestros conocimientos y brindarnos la oportunidad de desarrollarnos profesionalmente.

Queremos agradecer especialmente al Ing. Marcos Orellana, nuestro director de tesis y profesor en los años de nuestra carrera, que con su ayuda y paciencia, contribuyera con sus conocimientos al progreso de este proyecto de investigación.

## ÍNDICE DE CONTENIDOS

INTRODUCCIÓN .....	2
Capítulo 1 .....	3
Visión Artificial .....	3
1.1.    Conceptos básicos .....	3
1.1.1.    Definición.....	3
1.1.2.    Clasificación.....	5
1.1.2.1.    Área espacial .....	5
1.1.2.2.    Área industrial.....	5
1.1.2.3.    Área de seguridad.....	5
1.1.2.4.    Área de robótica .....	6
1.1.2.5.    Área de biología .....	6
1.1.2.6.    Área de la medicina.....	6
1.1.2.7.    Área biomedicina .....	7
1.1.3.    Aplicaciones .....	7
1.1.3.1.    Aplicaciones industriales .....	7
1.1.3.2.    Aplicaciones para el área espacial.....	8
1.1.3.3.    Aplicaciones para la robótica .....	9
1.1.3.4.    Aplicaciones para la biomedicina .....	9
1.1.3.5.    Aplicaciones para la seguridad.....	10
1.2.    Funcionamiento.....	12
1.2.1.    Elementos requeridos .....	12
1.2.1.1.    Cámara .....	12
1.2.1.2.    Dispositivo de interfaz .....	13
1.2.1.3.    Computador.....	13
1.2.1.3.1.    Componentes de visión por computador .....	14
1.2.1.3.2.    Componentes de Reconocimiento Óptico de Caracteres ...	14
1.2.1.3.3.    Entorno de desarrollo .....	14
1.2.2.    Estructura general.....	15
1.2.2.1.    Detección y captura de objetos: .....	15
1.2.2.1.1.    Captura de imagen.....	15
1.2.2.1.2.    Iluminación.....	16
1.2.2.1.3.    Pre-procesamiento: .....	16

1.2.2.2.	Extracción de características:.....	16
1.2.2.2.1.	Reconocimiento de patrones.....	16
1.2.2.2.2.	Procesamiento de imágenes.....	17
1.2.3.	Entrenamiento y detección.....	17
1.2.3.1.	Boosting .....	17
1.2.3.2.	Entrenamiento .....	17
1.2.3.2.1.	Detección.....	18
1.3.	Futuro en el ámbito de procesamiento de patrones.....	20
Capítulo 2.....		23
Análisis de la situación actual .....		23
2.1.	Estudio de la problemática actual. ....	23
2.1.1.	Causas del problema .....	24
2.1.2.	Efectos del problema.....	25
2.1.2.1.	Principales afectados.....	26
2.2.	Delimitación del problema.....	27
Capítulo 3.....		29
Estudio de herramientas de visión artificial.....		29
3.1.	Soluciones propuestas a los problemas mencionados.....	29
3.2.	Herramientas seleccionadas para el sistema. ....	31
3.2.1.	Herramientas de reconocimiento de patrones. ....	31
3.2.1.1.	Características.....	33
3.2.1.2.	Aplicación en el sistema.....	35
3.2.1.3.	Requerimientos. ....	36
3.2.1.3.1.	Software.....	36
3.2.1.3.2.	Hardware .....	36
3.2.2.	Herramienta de digitalización de caracteres (OCR).....	37
3.2.2.1.	Características. ....	38
3.2.2.2.	Aplicación en el sistema.....	41
3.2.2.3.	Requerimientos. ....	41
3.2.2.3.1.	Elemento de detección.....	42
3.2.2.3.2.	Software.....	42
3.2.2.3.3.	Hardware .....	42
3.2.3.	Herramienta de procesamiento de imágenes.....	42
3.2.3.1.	Características.....	43

3.2.3.2.	Aplicación en el sistema.....	49
3.2.3.3.	Requerimientos. ....	50
Capítulo 4.....		52
Análisis y diseño del sistema. ....		52
4.1.	Recolección de información para el sistema.....	52
4.1.1.	Investigación de características de un sistema de seguridad.....	52
4.1.1.1.	Estudios de sistemas existentes.....	52
4.1.1.2.	Investigación de características a implementar.....	54
4.1.2.	Obtención de elementos requeridos de cada herramienta definida	56
4.1.2.1.	Obtención de imágenes. ....	56
4.1.2.2.	Definición de la estructura de las imágenes. ....	58
4.1.2.2.1.	Clasificación.....	59
4.1.2.2.2.	Numeración.....	60
4.1.2.2.3.	Posición interna.....	60
4.1.2.2.4.	Ubicación.....	61
4.2.	Determinación de los requerimientos básicos de hardware.....	62
4.2.1.	Equipos necesarios para desarrollo y pruebas.....	62
4.3.	Identificación de los procesos que intervienen en el sistema.....	64
4.3.1.	Diagrama de casos de uso.....	64
4.3.2.	Diagrama de actividades.....	66
4.3.3.	Diagrama de clases.....	70
4.4.	Determinación de los flujos de datos.....	70
4.5.	Determinación de la información para los almacenes de datos.....	72
4.6.	Diseño de la base de datos.....	73
4.7.	Definición de procedimientos en cada herramienta.....	75
4.7.1.	Elementos de entrada.....	76
4.7.2.	Procedimientos a realizar.....	79
4.7.2.1.	Enumeración de pasos a seguir.....	79
4.7.2.2.	Observaciones.....	88
4.7.2.3.	Resultados esperados.....	90
Capítulo 5.....		92
Desarrollo del sistema.....		92
5.1.	Herramientas de programación.....	92
5.1.1.	Gestor de bases de datos.....	92

5.1.2.	Lenguajes de programación.....	94
5.2.	Evaluación y selección de componentes.....	96
5.2.1.	Componentes externos necesarios.....	96
5.2.2.	Evaluación de componentes.....	98
5.3.	Desarrollo del sistema.....	100
5.3.1.	Fase de codificación.....	100
5.3.1.1.	Entrenamiento.....	100
5.3.1.2.	Sistema (Aplicativo).....	109
5.3.1.2.1.	Preparación de librerías para el sistema.....	109
5.3.1.2.2.	Creación del sistema.....	113
5.3.2.	Desarrollo y monitoreo de progresos en el sistema.....	125
5.3.3.	Integración de resultados.....	130
Capítulo 6.....		134
Pruebas y simulaciones.....		134
6.1.	Pruebas de detección.....	134
6.2.	Pruebas de rendimiento.....	141
6.3.	Pruebas de interfaz.....	149
Capítulo 7.....		154
Manual de usuario.....		154
7.1.	Manual para el usuario supervisor.....	154
7.1.1.	Menú del usuario supervisor.....	155
7.1.2.	Detección de placa.....	155
7.1.2.1.	Acceso manual.....	156
7.1.2.2.	Acceso eventual como invitado.....	156
7.1.2.3.	Acceso de vehículo con registro nuevo.....	158
7.1.2.4.	Denegar acceso a vehículo no registrado.....	161
7.1.3.	Consultas.....	161
7.1.3.1.	Consulta de registro de acceso.....	161
7.1.3.1.1.	Búsqueda de registros.....	162
7.1.3.2.	Consulta de empleado.....	163
7.1.3.2.1.	Búsqueda de empleados.....	164
7.2.	Manual de administrador.....	164
7.2.1.	Menú de administrador.....	164
7.2.2.	Mantenimientos.....	165

7.2.2.1. Usuarios .....	165
7.2.2.1.1. Ingreso de usuarios .....	165
7.2.2.1.2. Modificación de usuarios .....	166
7.2.2.1.3. Eliminación de usuarios .....	167
7.2.2.2. Empleados .....	168
7.2.2.2.1. Ingreso de empleados .....	168
7.2.2.2.2. Modificación de empleados.....	170
7.2.2.2.3. Eliminación de empleados.....	171
7.2.2.3. Cargos .....	171
7.2.2.3.1. Ingreso de cargos .....	172
7.2.2.3.2. Modificación de cargos .....	173
7.2.2.3.3. Eliminación de cargos .....	173
7.2.2.4. Vehículos .....	174
7.2.2.4.1. Ingreso de vehículos .....	175
7.2.2.4.2. Modificación de vehículos .....	176
7.2.2.4.3. Eliminación de vehículos .....	177
CONCLUSIONES .....	179
RECOMENDACIONES .....	182
BIOBLOGRAFIA.....	184

## ÍNDICE DE IMÁGENES

Figura 1.1 Procesos de la visión artificial. ....	4
Figura 1.2 Brazo robótico implementando visión artificial. (Technologies, 2008).....	6
Figura 1.3 Vision Hawk cámara de control de calidad. (Microscan, 2013).....	8
Figura 1.4 ThermalSentry detección de calor corporal para enfermedades.....	10
Figura 1.5 SmartLoop sistema de detección de tráfico y velocidad .....	11
Figura 1.6 Cámara Web. ....	13
Figura 1.7 Cálculo de detección dos rectángulos.....	19
Figura 1.8 Cálculo de detección tres rectángulos.....	19
Figura 1.9 Cálculo de detección cuatro rectángulos .....	19
Figura 3.10 Detección de líneas base en texto inclinado .....	39
Figura 3.11 Detección de caracteres en palabras unidas.....	39
Figura 3.12 Problemas de espaciado en palabras.....	39
Figura 3.13 Histograma de una imagen .....	44
Figura 3.14 Detección de contornos de un objeto dentro de una imagen. ....	45
Figura 3.15 Segmentación de un objeto.....	45
Figura 4.16 Placa tomada para entrenamiento .....	56
Figura 4.17 Imágenes tomadas de los caracteres. ....	57
Figura 4.18 Caracteres clasificados y binarizados. ....	57
Figura 4.19 Imagen capturada al detectar placa.....	58
Figura 4.20 Imagen recortada. ....	58
Figura 4.21 Imagen positiva.....	59
Figura 4.22 Imagen negativa.....	60
Figura 4.23 Ejemplo de numeración. ....	60
Figura 4.24 Ejemplo de coordenadas.....	61

Figura 4.25 Ejemplo de archivo box con QT Box Editor. ....	62
Figura 4.26 Captura de placas para entrenamiento .....	80
Figura 4.27 Ejemplo de imágenes negativas.....	80
Figura 4.28 Archivos de direcciones para imágenes positivas y negativas .....	81
Figura 4.29 Programa de comprobación de resultados .....	82
Figura 4.30 Extracción de caracteres para entrenamiento OCR .....	82
Figura 4.31 Ejemplo de archivo TIF de lenguaje de entrenamiento .....	83
Figura 4.32 Herramienta “QT Box Editor” .....	83
Figura 4.33 Archivo box para entrenamiento .....	84
Figura 4.34 Archivo Font_properties .....	85
Figura 4.35 Contenido de la carpeta para entrenamiento.....	85
Figura 4.36 Ejemplo imagen almacenada en proceso de captura. ....	86
Figura 4.37 Ejemplo de placa recortada.....	87
Figura 4.39 Ejemplo de detección de caracteres .....	88
Figura 5.40 Interfaz Mysql Workbench .....	94
Figura 5.41 Interfaz Netbeans. ....	95
Figura 5.42 Generación del vector de muestras Open CV.....	102
Figura 5.43 Generación archivo de entrenamiento Open CV .....	103
Figura 5.44 Cascade.xml generado .....	104
Cuadro 5.10 Elementos para entrenamiento Tesseract .....	104
Figura 5.45 Depuración de los caracteres para entrenamiento. ....	105
Figura 5.46 Archivo para entrenamiento de tesseract.....	105
Figura 5.47 QT Box generador de coordenadas de caracteres.....	106
Figura 5.48 Archivo box generado.....	106
Figura 5.49 Generación del archivo training.....	107
Figura 5.50 Generación de la Shapetable.....	108

Figura 5.51 Archivos renombrados para entrenamiento .....	108
Figura 5.52 Generación del archivo final traindata.....	109
Figura 5.53 Librerías externas JAVA .....	110
Figura 5.54 Librerías Hibernate .....	110
Figura 5.55 Cadena de conexión de la base de datos.....	111
Figura 5.56 Resumen de configuración de la base de datos .....	111
Figura 5.57 Xml de estructura de las tablas mapeadas por Hibernate .....	112
Figura 5.58 Estructura de librerías de Tess4J .....	112
Figura 5.59 Login del sistema.....	113
Figura 5.60 Opciones del menú del sistema.....	113
Figura 5.61 Interfaz detección de la placa .....	114
Figura 5.62 Imagen de la placa binarizada.....	115
Figura 5.63 Información de placa detectada .....	116
Figura 5.64 Ubicación de las imágenes almacenadas en la carpeta imgplaca. ....	117
Figura 5.65 Mensaje de confirmación de acceso .....	117
Figura 5.66 Formulario de ingreso de vehículo .....	118
Figura 5.67 Semáforo en rojo representa un acceso denegado .....	118
Figura 5.68 Semáforo en amarillo representa el sistema en espera de detección ....	119
Figura 5.69 Botón de acceso manual .....	119
Figura 5.70 Cuadro de información del sistema .....	119
Figura 5.71 Ventana de mantenimiento de vehículos .....	120
Figura 5.72 Campo de búsqueda del sistema .....	121
Figura 5.73 Ejemplo de criterio de búsqueda.....	121
Figura 5.74 Ventana de información para editar los registros .....	122
Figura 5.75 Botones que realizarán las acciones en el formulario.....	122
Figura 5.76 Consulta de empleados .....	123

Figura 5.77 Búsqueda de empleados.....	123
Figura 5.78 Búsqueda general de registros de acceso.....	124
Figura 5.79 Búsqueda avanzada de registros de acceso.....	124
Figura 5.80 Consulta de registro de acceso.....	125
Figura 5.81 Prueba de detección de rostros. ....	125
Figura 5.82 Interfaz de pruebas de detección.....	126
Figura 5.83 Sección de pruebas de OCR .....	127
Figura 5.84 Interfaz de integración con la base de datos. ....	127
Figura 5.85 Interfaz final de detección de placas.....	128
Figura 5.86 Interfaz de registro de accesos .....	128
Figura 5.87 Interfaz final de registro de accesos.....	129
Figura 5.88 Interfaz de mantenimiento de empleados .....	129
Figura 5.89 Interfaz final de mantenimiento de empleados.....	130
Figura 5.90 Cascade XML de detección de placas .....	131
Figura 5.91 Idioma de entrenamiento OCR .....	132
Figura 6.92 Placa en buen estado .....	134
Figura 6.93 Placa en buen estado binarizada. ....	135
Figura 6.94 Resultado de la placa en buen estado binarizada.....	135
Figura 6.95 Placa en estado regular. ....	136
Figura 6.96 Placa en estado regular binarizada.....	136
Figura 6.97 Resultados de la placa en estado regular. ....	136
Figura 6.98 Placa en mal estado.....	137
Figura 6.99 Placa en mal estado binarizada. ....	137
Figura 6.100 Resultados de la placa en mal estado.....	137
Figura 6.101 Placa en un ambiente oscuro.....	138
Figura 6.102 Placa binarizada en un ambiente oscuro.....	138

Figura 6.103 Resultados de la placa en un ambiente oscuro.....	138
Figura 6.104 Placa en ambientes claros. ....	139
Figura 6.105 Placa en ambientes claros binarizada. ....	139
Figura 6.106 Resultados de la placa en ambientes claros. ....	139
Figura 6.107 Placa con diferente tipografía. ....	140
Figura 6.108 Placa binarizada con diferente tipografía. ....	140
Figura 6.109 Resultado de placa con diferente tipografía.....	140
Figura 6.110 Captura con cámara VGA, distancia corta. ....	141
Figura 6.111 Binarización con cámara VGA, distancia corta.....	141
Figura 6.112 Resultado con cámara VGA distancia corta. ....	141
Figura 6.113 Captura con cámara VGA, distancia larga. ....	142
Figura 6.114 Resultados con cámara VGA, distancia larga.....	142
Figura 6.115 Captura con cámara HD, distancia corta. ....	143
Figura 6.116 Resultado con cámara HD, distancia corta. ....	143
Figura 6.117 Captura de cámara HD, distancia media.....	144
Figura 6.118 Resultado de cámara HD, distancia media. ....	144
Figura 6.119 Captura de cámara HD, distancia larga. ....	145
Figura 6.120 Binarización de cámara HD, distancia larga.....	145
Figura 6.121 Resultados de cámara HD, distancia larga. ....	145
Figura 6.122 Tiempo de inicialización del sistema, pc de gama alta.....	146
Figura 6.123 Tiempo de detección del sistema, pc de gama alta. ....	147
Figura 6.124 Tiempo de consulta del sistema, pc de gama alta. ....	147
Figura 6.125 Tiempo de inicialización del sistema, pc de gama media.....	148
Figura 6.126 Tiempo de detección del sistema, pc de gama media. ....	148
Figura 6.127 Tiempo de consulta del sistema, pc de gama media. ....	149
Figura 6.128 Interfaz de acceso correcto. ....	150

Figura 6.129 Interfaz de placa no registrada.....	150
Figura 6.130 Interfaz de acceso denegado. ....	151
Figura 6.131 Interfaz de mantenimiento. ....	152
Figura 6.132 Interfaz de consulta de registros de acceso. ....	153

## **RESUMEN**

Actualmente la tecnología presenta múltiples herramientas para optimizar y simular procesos que requieren la intervención humana. Entre ellas destaca la visión artificial, que al no ser perfecta como la visión humana tiene la capacidad de automatizar ciertos procesos, reduciendo significativamente la dependencia directa del individuo.

En la presente investigación se empleará esta tecnología sobre un problema real, como el ingreso automatizado de vehículos. Este proceso permitirá mayor rapidez en el ingreso vehicular además de incrementar los niveles de seguridad y organización.

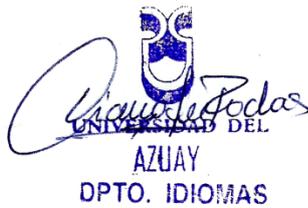
Para esta implementación se utilizará herramientas de reconocimiento de patrones y OCR permitiendo obtener un mejor rendimiento y precisión.

## ABSTRACT

Nowadays technology offers many tools to optimize and simulate processes that require human intervention. These include the artificial vision which, even though it is not as perfect as human vision, has the ability to automate certain processes, significantly reducing the direct dependence from the person.

During the present investigation, we will use this technology on a real problem such as the automated entry of vehicles. This process will allow faster vehicular entrance in addition to increasing the safety and organizational levels.

For its implementation, we will use tools for pattern recognition and OCR (Optical character recognition) so as to allow better performance and accuracy.



  
Translated by,  
Lic. Lourdes Crespo

## RESPONSABILIDAD

La responsabilidad del contenido de esta tesis de grado con el tema “**Sistema para registro de ingreso de vehículos a la Universidad del Azuay mediante el reconocimiento de placas**”, corresponde exclusivamente a Fabián Mauricio Atariguana Quezada y Pedro Andrés Ávila Regalado.



---

**Pedro Andrés Avila Regalado**



---

**Fabián Mauricio Atariguana Quezada**

## INTRODUCCIÓN

El presente trabajo de investigación establece un enfoque hacia el estudio y el desarrollo de un sistema de visión artificial, para la solución de problemas de ingreso vehicular.

Como principal referencia se estudió y analizó el caso particular del parqueadero de la Universidad del Azuay, debido a que actualmente requiere un mayor control y organización en el ingreso vehicular. La falta de un registro de actividades dificulta saber con exactitud el momento de acceso de un vehículo o su grado de autorización, para ello, se busca establecer una aplicación que gestione todos estos procesos eficientemente.

La aplicación permitirá verificar la placa del vehículo a través de una cámara web con herramientas informáticas de identificación y reconocimiento de patrones, luego realizará un registro de ingreso en una base de datos, lo que permitirá posteriores consultas.

Este sistema aprovechará la capacidad de procesamiento del computador para realizar rápidamente los procesos de detección y comprobación de la placa, esto validará el acceso al parqueadero verificando si el vehículo está autorizado. El sistema no solo automatizará el ingreso vehicular, sino también registrará la información del propietario.

## Capítulo I

### Visión Artificial

#### Introducción

A continuación se especificará todas aquellas definiciones que forman parte de la tecnología de visión artificial, empezando por aclarar los términos que se manejarán a lo largo de la creación del sistema. Estos van desde la definición de conceptos, los ámbitos de desarrollo, la estructura y el funcionamiento de cada componente que forma parte de la misma.

#### 1.1. Conceptos básicos

Como primer punto se definirán los conceptos, clasificación y áreas en las que se desarrolla actualmente la tecnología de visión artificial.

##### 1.1.1. Definición

“Inteligencia Artificial” es el campo de las ciencias de la computación, que consiste en el desarrollo de algoritmos que permiten la toma de decisiones de organismos que realizan procesos mecánicos. Esta ciencia tiene un gran número de ramas, de la cual se deriva la visión artificial. (Stanford, 2003)

“Visión Artificial” o “Visión por computador” no es más que la extracción de información de un ambiente o entorno, presentado o capturado mediante imágenes de una cámara web, que permiten obtener una percepción del mundo real, y procesado mediante técnicas de análisis de información especial. (Centro Integrado Politécnico, 2010).

Esta tecnología está conformada por cinco procesos básicos para la realización del reconocimiento y análisis de las imágenes. Dentro de los cuales se definen los siguientes:

- Captación y digitalización de imágenes
- Pre procesamiento
- Segmentación

- Representación y descripción de resultados
- Reconocimiento e interpretación de resultados.

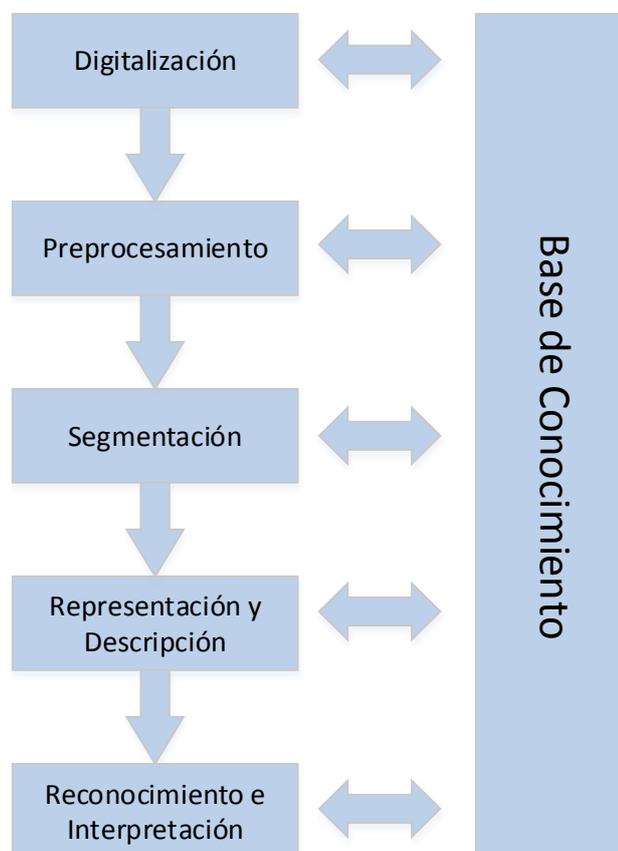


Figura 1.1 Procesos de la visión artificial.

La “Visión Artificial” tiene como objetivo principal, entender la imagen que percibe el computador, es decir, identificar sus objetos, sus relaciones, sus posiciones y su importancia; para luego realizar tareas sobre ella, ya sea, interactuar sobre características específicas de la imagen o del ambiente capturado, validar acciones a realizar en una situación específica, etc. (Quintero, 2010).

Es importante señalar que la “Visión Artificial” en comparación con la “Visión Humana” difiere mucho en cuanto a capacidades y desempeño. La visión artificial como ciencia no se encuentra desarrollada en su totalidad, es por eso que la misma no puede superar a la capacidad de reconocimiento de la visión humana, ya que la visión humana mantiene una mayor adaptabilidad a situaciones imprevistas o entornos cambiantes, puesto que consta de un conocimiento previo. (Daniel Bescansa, 2006)

Sin embargo, considerando los antecedentes anteriormente especificados la Visión Artificial resulta una excelente herramienta para lograr la automatización y un buen desempeño de tareas rutinarias, fácilmente reemplazables por la tecnología.

### **1.1.2. Clasificación**

La “Visión Artificial” en el transcurso de su crecimiento se ha venido acoplado a las diferentes áreas de la ciencia según sus necesidades, ya sea para solucionar ciertos problemas o para mejorar actividades rutinarias. Debido a ello se ha obtenido la siguiente clasificación (CEDIBA, 2006):

#### **1.1.2.1. Área espacial**

Esta área consiste en la utilización de sensores o cámaras en satélites y robots, cuyo fin es analizar el terreno, tomar muestras y procesarlas, estos dispositivos generalmente son comandados desde la tierra teleméricamente. Algunos sistemas también son programados para que tomen sus propias decisiones, como tareas de navegación (esquivar objetos, trazar rutas, etc.), buscar indicios de vida, tomar muestras, detectar movimiento, etc.

#### **1.1.2.2. Área industrial**

Está conformado por brazos robóticos que se encargan de realizar tareas programadas. Presentan cámaras y sensores para tener un control visual sobre el objeto y determinar las acciones que van a realizar conforme se avanza con el procesamiento del mismo. Entre las tareas que realizan se identifican: ensamblaje de piezas, perforaciones, cortes, etc.

#### **1.1.2.3. Área de seguridad**

Establecida por cámaras de seguridad y sensores de movimiento que detectan el acceso de un individuo o de un vehículo, y lo validan mediante algoritmos de detección de características como: el rostro, la voz, una identificación, etc. También permiten registrar toda la información del acceso, o realizar seguimientos y no perder al objetivo de vista.

#### 1.1.2.4. Área de robótica

En este ámbito la visión artificial se implementa a los diferentes robots que poseen inteligencia artificial. Permiten el modelado 3D del escenario de navegación para que el robot pueda disponer de este a su antojo, además de detectar señales de movimiento, realizar acciones complejas con interacciones, ya sea con otros robots o con presencias humanas.



Figura 1.2 Brazo robótico implementando visión artificial. (Technologies, 2008)

#### 1.1.2.5. Área de biología

Esta área permite, mediante cámaras de alta resolución la detección de organismos microscópicos, y mediante técnicas de segmentación el aislamiento según las diferentes características del elemento, además permite realizar un conteo de grandes cantidades de organismos en un tejido e identificar el estado de una planta o un animal por sus características como textura, color, brillo, etc.

#### 1.1.2.6. Área de la medicina

En esta área, la visión artificial tiene una amplia gama de aplicaciones orientadas hacia las dolencias o las enfermedades. Estas utilizan técnicas como “Image Registration”, que consiste en definir estrategias para encontrar las

características comunes entre un modelo y una muestra, y así determinar el grado de semejanza para ser catalogado. Esta técnica permite aprovechar la tecnología de visión artificial para realizar análisis de radiografías, resonancias magnéticas, tomografías, etc.

#### **1.1.2.7. Área biomedicina**

Estos sistemas se encuentran compuestos por dispositivos externos, como cámaras o sensores, que captan las imágenes del ambiente y las procesan, para luego ser transmitidas por elementos internos vía estímulos eléctricos enviados hacia el cerebro, estas señales codificadas crean la percepción de letras, objetos geométricos, figuras, etc. Estos implantes serán ubicados en distintas partes del cuerpo humano conectadas al cerebro para estimular las neuronas y dar sensaciones de visión y detección del mundo real.

#### **1.1.3. Aplicaciones**

Se debe considerar que la Visión Artificial es una ciencia aun en desarrollo, por lo que la misma puede tener ciertas brechas o limitaciones, es por ello que en ciertos casos la solución más óptima sigue siendo la visión humana para la identificación de situaciones especiales; sin embargo con el pasar de los días la evolución de esta tecnología permiten brindar soluciones mediante numerosas aplicaciones. Entre las aplicaciones más comunes realizadas mediante visión artificial se definen: (INFAIMON, 2013):

##### **1.1.3.1. Aplicaciones industriales**

Como principal referencia se identifican las aplicaciones de control de calidad, ya que estos procesos rutinarios mantienen ciertos parámetros con pequeñas probabilidades de cambio, estos son evaluados fácilmente al momento de verificar un mismo producto.

Mediante estas soluciones se alcanza un mejor nivel de producción así como se mantiene un mejor control de calidad, tanto de productos como procesos, ayudando a minimizar los tiempos muertos y representando un incremento de fiabilidad dentro del área industrial con una reducción de costes significativa.



Figura 1.3 Vision Hawk cámara de control de calidad. (Microscan, 2013)

Entre las aplicaciones más destacadas de este ámbito se puede mencionar:

- Control de calidad ya sean de productos terminados o partes de un proceso para la obtención de un producto final.
- Control e inspección de los procesos de fabricación de los productos.

#### **1.1.3.2. Aplicaciones para el área espacial**

La visión artificial para esta área comprende la aplicación de reconocimiento a distancia para la obtención de información sin necesidad de contacto físico.

Entre sus principales métodos se encuentra el tratamiento de imágenes, la misma que es analizada y procesada para la obtención de información aplicable a cualquiera de las áreas anteriormente explicadas; se mencionara los campos que implementan esta categoría:

- Clasificación de cultivos
- Asistencia de cartografía
- Control de asentamientos urbanísticos
- Monitorización de recursos naturales
- Control de incendio e inundaciones, etc.

Es importante destacar que dentro de esta área existe una gran combinación de tecnologías, con la parte de hardware más orientada a la robótica, ya que en

ciertos procesos se maneja con artefactos como: brazos electrónicos, vehículos guiados y equipos de tratamiento de imágenes automáticos.

### **1.1.3.3. Aplicaciones para la robótica**

En el área de la robótica las aplicaciones de visión artificial son más numerosas y evidentes, dentro de las cuales pueden considerarse aquellas como el control e inspección, automatización de procesos, guía de objetos, etc. Muchas de estas aplicaciones permiten una integración de tecnologías como: aquellas orientadas al seguimiento visual de vehículos, sistemas de seguridad, misiles, aplicaciones espaciales, etc. Se destaca el caso del reconocimiento de objetos para evitar obstáculos que mezcla diferentes tipos de visión artificial. Esto consiste en:

- Una construcción de modelos de objetos.
- Localización y reconocimiento de objetos.
- Determinar posición actual y ruta de movimiento.

### **1.1.3.4. Aplicaciones para la biomedicina**

Una de las áreas donde se requiere de mayor precisión y en donde las soluciones presentadas por la visión artificial constituyen un amplio porcentaje de desarrollo, es en la biomedicina. Dentro de esta área las aplicaciones son orientadas a la detección e interpretación de radiografías, tomografías, etc.

Un enfoque adicional de esta tecnología permite el tratamiento de dolencias presentadas por los pacientes, destinado al uso de aplicaciones robóticas en operaciones asistidas.

Como otro punto se destaca la detección y diferenciación de tejidos, este proceso consiste en especificar todas aquellas características que determinan si el problema es una simple infección o son lesiones que ocasionan problemas más serios, mediante la extracción de bordes o patrones que identifican las variaciones en los tejidos.



Figura 1.4 ThermalSentry detección de calor corporal para enfermedades. (Infaimon, 2012)

#### 1.1.3.5. Aplicaciones para la seguridad

En el área de seguridad; la visión artificial permite mejorar el control de acceso automatizado a una empresa ya que códigos o tarjetas no son lo suficientemente precisos y pueden ser mecanismos vulnerables a infiltraciones o fallos, esta tecnología permitirá captar características específicas tanto de la persona como del objeto a validar.

Entre los más utilizados se destaca la identificación de huellas digitales, que consiste en la captura y comparación de varias muestras identificando los puntos relevantes de la misma, esta técnica es aplicada en diversas actividades como: marcación de personal, acceso a descuentos, créditos para clientes, etc.

Una de las aplicaciones que usa algoritmos más complejos es el reconocimiento del iris, que emplea la iluminación infrarroja sobre el ojo para reducir su reflejo y así tomar una muestra completa del iris validando el acceso del individuo con el uso de algoritmos matemáticos. Estas aplicaciones han sido utilizadas en algunos países como Holanda, Canadá y Estados Unidos con la finalidad de brindar un rápido acceso dentro del aeropuerto y el control de fronteras.

Otra de las aplicaciones más usadas dentro de la seguridad es el reconocimiento facial, este consiste en tomar la imagen del rostro de una persona y

determinar sus fotogramas en un vector de características, este se encargará de compararlas con un conjunto de imágenes de una base de conocimiento. Esta tecnología es usada en la detección de sospechosos en robos, identificación de personas en lugares de acceso masivo, en el registro de un historial de visitas, identificación dentro de cajeros automáticos y también es aplicable a medios de entretenimiento como dispositivos móviles, aplicaciones web, videojuegos, etc.

Un enfoque que actualmente se está adoptando en la seguridad es la detección de vehículos infractores principalmente usado para el control policial, este consiste en identificar la placa de todos aquellos automóviles que superan el límite de velocidad.



Figura 1.5 SmartLoop sistema de detección de tráfico y velocidad. (Quercus Technologies, 2010)

Adicionalmente, una nueva forma de control que empieza a tener acogida tanto en instituciones privadas como públicas, es el acceso vehicular automatizado, este consiste en la detección de la matrícula, la interpretación de cada uno de sus caracteres se comparan en una base de datos y permiten identificar toda la información del conductor que accede a la institución.

En esta última aplicación se basará el presente proyecto como una solución al control vehicular y será una de las primeras en implementarse en la ciudad impulsando el desarrollo de esta tecnología dentro de la seguridad.

## **1.2. Funcionamiento**

Una vez definida la tecnología de visión artificial, es indispensable especificar su estructura y funcionalidad con el fin de conocer a detalle cada elemento que la conforma (Druzella, 2010):

### **1.2.1. Elementos requeridos**

En cuanto a hardware y software se tienen en cuenta los siguientes elementos indispensables en el desarrollo de la técnica de visión artificial:

#### **1.2.1.1. Cámara**

Entre los diferentes tipos de cámaras que contienen características relevantes se detallan las siguientes:

- Cámaras matriciales.- Estas pueden captar imágenes en 2 dimensiones.
- Cámaras lineales.- A diferencia de las matriciales estas captan una sola línea de imagen.
- Cámaras inteligentes.- Llevan el procesador de imágenes en la misma carcasa de la cámara realizando la mayoría de operaciones instantáneamente pero su velocidad, su procesamiento es más lento que una pc normal.
- Cámaras convencionales.- Cuya única función es captar las imágenes y enviarlas a una fuente de procesamiento externa.
- Cámaras infrarrojas/térmicas.- Cámaras que permiten la detección de imágenes más allá del espectro luminoso, es decir, permiten captar el nivel de radiación infrarroja que emite un cuerpo.

Se debe tener en consideración que mientras menores características presente una cámara, el sistema realizara mayor procesamiento de la imagen aumentando el tiempo de reconocimiento.

La cámara a ser utilizada para este proyecto será una cámara web de captación lineal. Más adelante se evaluará una cámara adecuada para el sistema que permita cumplir los objetivos de detección.



Figura 1.6 Cámara Web.

### **1.2.1.2. Dispositivo de interfaz**

Este se encargará de recibir las señales eléctricas enviadas por la cámara y transformarlas en una señal digital para que pueda ser asimilada por el computador y puedan realizarse las acciones de procesamiento. Generalmente las cámaras web se conectan directamente al computador que hace de dispositivo de interfaz por lo que no se requiere un componente específico.

### **1.2.1.3. Computador**

Contiene el software que se encargará de realizar el procesamiento y el control de las imágenes adquiridas por la cámara y recibidas por el dispositivo de interfaz, que formará parte de este mismo computador. Sus requerimientos dependerán de la velocidad con la que se pretende tratar la información y la cantidad de datos que vaya a manejar el sistema. Donde se verá afectado el desempeño del ordenador, será en el tiempo en que realice el aprendizaje de objetos en el procesamiento y depuración de la imagen.

Es recomendable realizar el aprendizaje en máquinas potentes con procesadores rápidos para que el proceso de entrenamiento se realice en menor tiempo y luego de esto, el sistema podrá ser ejecutado en ordenadores con requerimientos menores.

Luego de haber captado la imagen, es necesario presentar ciertos componentes de software que se encargarán de extraer la información que contiene esta imagen (texturas, formas, colores, posición de objetos. etc.).

#### **1.2.1.3.1. Componentes de visión por computador**

En este punto se requiere de una herramienta que permita realizar la operación de visión artificial para el proceso de detección del objeto. Esta herramienta contendrá ciertas características que permitan identificar y recortar una placa.

Toda la funcionalidad y las herramientas seleccionadas se especificarán a detalle en los siguientes capítulos.

#### **1.2.1.3.2. Componentes de Reconocimiento Óptico de Caracteres (OCR)**

“El reconocimiento óptico de caracteres u OCR (Optical Character Recognition), consiste en la identificación automatizada de símbolos o caracteres pertenecientes a un alfabeto, a partir de una imagen recogida mediante lectura óptica de un texto grabado en un apoyo real”. (*Arlandis, 2013*).

Este alfabeto deberá ser almacenado dentro de una base de conocimientos con la cual se podrá interactuar posteriormente.

El OCR identifica inicialmente una imagen binaria, es decir de dos colores (blanco y negro), la misma que es comparada entre los caracteres obtenidos en la imagen con el alfabeto almacenado en una base de conocimiento. Cabe destacar que para un buen funcionamiento dependerá de una buena definición de los caracteres dentro de la base de conocimiento. (Javier Cano, 2010)

#### **1.2.1.3.3. Entorno de desarrollo**

Es la combinación de herramientas que automatizan o facilitan las fases de desarrollo como son: análisis, diseño, codificación y pruebas.

El entorno de desarrollo o programación, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. Estos pueden ser aplicaciones por sí solas o formar parte de aplicaciones existentes. (LSIIS, 2004)

### **1.2.2. Estructura general**

El sistema de visión artificial presenta cuatro pasos bien definidos, que representan a manera general su comportamiento. Dentro de cada paso se tendrá en cuenta dos enfoques; el primero orientado al reconocimiento de la placa como un solo objeto, y el segundo que será el reconocimiento de cada carácter de esta placa.

En términos de funcionalidad, la visión artificial pretende realizar un procesamiento de imágenes basadas en dos propiedades fundamentales que son:

- Propiedades geométricas: Estas pueden constar de características como el tamaño, la forma o la localización.
- Propiedades de composición de los materiales: Es decir características de iluminación, color, textura, etc.

La visión artificial posee 3 principales procesos ordenados que definen su estructura:

#### **1.2.2.1. Detección y captura de objetos**

Es el paso inicial para la estructura de la visión artificial, donde constará de los elementos de captura que permitirán la obtención de la imagen y los factores externos que influirán en la detección del objeto.

##### **1.2.2.1.1. Captura de imagen**

Las cámaras son las encargadas de captar las imágenes para su posterior procesamiento en el sistema. Existen varios tipos de cámara y dependiendo de las características de cada una, mejorarán la calidad de la captura, permitiendo diferenciar claramente las características del objeto a detectarse. Para una imagen óptima, sin tomar en cuenta la cámara, se debe analizar factores externos como: ángulo de visión, enfoque, calidad de la imagen, etc.

#### **1.2.2.1.2. Iluminación**

La iluminación presenta uno de los factores importantes de esta tecnología ya que influirá directamente en la imagen que será captada, es decir, una buena iluminación permitirá que la imagen sea tomada con más claridad para su procesamiento, mientras una mala iluminación hará que el sistema trabaje más para su detección e inclusive podrá causar que no sea identificada por el sistema.

Este conjunto de componentes trabajarán de manera óptima para dar un mejor resultado y de ellos dependerá la calidad de la imagen que va a ser procesada.

#### **1.2.2.1.3. Pre-procesamiento**

Este proceso no obligatorio aunque efectivo, puede ser previamente utilizado para mejorar la imagen que se va a analizar. Este se encarga de resaltar las características de una imagen aumentando su calidad, aplicando técnicas como el cambio de brillo, contraste, la eliminación del ruido de la imagen, cambio de colores, etc.

Este proceso puede ser usado tanto en el entrenamiento como en la detección de la imagen para así mejorar la calidad con la que estas se manejan.

#### **1.2.2.2. Extracción de características**

Este proceso debe determinar cómo se identificará el objeto y como se procesará la información obtenida para llegar a los resultados deseados. Debido a ello se debe localizar dentro de la imagen, aquellas características que delimitan el patrón a ser reconocido. Consta de dos subprocesos que son:

##### **1.2.2.2.1. Reconocimiento de patrones**

Este proceso se encarga de detectar las características del objeto a encontrar, especificando dónde se localizan grupos de píxeles que conforman el objeto, empleando los principios de discontinuidad y similitud.

#### **1.2.2.2. Procesamiento de imágenes**

Este proceso se encarga de recortar o extraer de la imagen el objeto diferenciado de su entorno, para ello se realiza un estudio y se extraen las características constantes que posea sin tomar en consideración rotaciones, cambios de tamaño y otras variaciones ajenas a las características propias del mismo. Dentro de las características se puede mencionar: área perímetro, circularidad, patrones de texturas, rasgos de formas, etc.

#### **1.2.3. Entrenamiento y detección**

Para que el computador permita identificar el objeto a ser detectado, es necesario someterlo a un proceso de entrenamiento. Este proceso consta de una serie de algoritmos que identifican aquellas características importantes de una imagen y las guardan en un archivo de entrenamiento.

Tanto en el entrenamiento como la detección, la mayoría de los programas que permiten el desarrollo de la visión artificial comparten un método muy importante de detección de imagen a manera de reglas denominado boosting.

##### **1.2.3.1. Boosting**

Este método consiste en la agrupación de varias reglas de detección de imágenes llamadas débiles, ya que por sí solas no presentan una posibilidad de detección aceptable, y así al unir las, conseguir un nivel de detección alto logrando construir un sistema de decisión potente. (Helmut Grabner, On-line Boosting and Vision, 2006)

##### **1.2.3.2. Entrenamiento**

Se establecen un conjunto de reglas para ser aprendidas por el sistema. Estas reglas son generadas por algoritmos llamados de “débil aprendizaje” para que luego mediante boosting las compare una por una en un conjunto de imágenes llamadas positivas y negativas; las positivas contendrán el objeto que se desea detectar y las negativas no lo contendrán. En cada comparación se pondrá una etiqueta por cada regla, especificando el porcentaje de acoplamiento, es decir, si la

imagen encaja o no con esta regla. Los resultados serán las imágenes que mejor porcentaje de acoplamiento tengan o que mejor se asemejen al objeto. La calidad de los resultados viene definida por el manejo de muestras variadas, para ello se pueden emplear tomas de diferentes ángulos y capturas de buena calidad, esto con el fin de obtener resultados óptimos mejorando la detección.

Luego de reunir las imágenes, se procede a la construcción del sistema de detección del objeto, este se basará en establecer todas las mejores reglas y unificarlas, para ello es necesario determinar con indicadores que reglas encajan con mayor o menor dificultad en ciertas imágenes. Las muestras que tengan mayor dificultad de encajar con las reglas tendrán pesos mayores y las de menor dificultad tendrán menores pesos.

Y de esta manera el aprendizaje del sistema tomará en mayor consideración las muestras que mayor peso tengan y así lograr una mejor y más robusta detección, ampliando sus características y su rango de percepción.

La parte fundamental del entrenamiento viene dada por la correcta selección de las imágenes de muestra, estas tienen que contener el objeto en las distintas perspectivas en las que se pudiera presentar en la detección. (Navarro Ubeda, 2004)

#### 1.2.3.2.1. Detección

El proceso de detección se basa en el cálculo de las reglas por diferentes métodos, estos son:

**Dos rectángulos:** Esta técnica consiste en colocar 2 rectángulos continuos en una zona aleatoria a la imagen en la que se quiere detectar el objeto y consiste en la resta de todos los valores de los píxeles de la zona del primer rectángulo con todos los valores de la zona del segundo rectángulo.

$$ht = zonaABCD - zonaCDEF$$



Figura 1.7: Cálculo de detección dos rectángulos

Estos valores serán comparados con un umbral. Este umbral será determinado por un valor obtenido en el entrenamiento, y consistirá en las características óptimas que deberá tener la imagen para ser detectada. La comparación consistirá en que si el resultado de la regla sobrepasa el umbral, entonces no es el objeto, si es menor al umbral si será el objeto. Existen variaciones de este método realizando rotaciones de los rectángulos en 90, 180 y 270 grados.

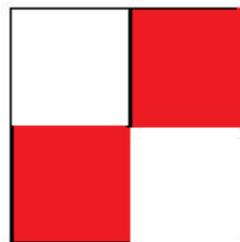
**Tres rectángulos:** Este método a diferencia del anterior se sumarán los 2 rectángulos exteriores y se restará el rectángulo exterior. También se lo puede rotar en 90 grados.



Tres Rectángulos

Figura 1.8: Cálculo de detección tres rectángulos

**Cuatro rectángulos:** En estos el cálculo se realiza sumando los rectángulos de cada diagonal y restándolas entre ellas.



Cuatro Rectángulos

Figura 1.9: Cálculo de detección cuatro rectángulos

Las reglas pueden tener diferentes variaciones como: cambios de ángulos, escalas, etc. Además estas operaciones se dividen en subventanas para disminuir la carga de procesamiento del computador.

### **1.3. Futuro en el ámbito de procesamiento de patrones**

En sus inicios, la tecnología de la Visión Artificial se encontraba de manera primitiva, por lo que su uso era muy limitado y sus aplicaciones no presentaban total confianza en situaciones cambiantes o ambientes no esperados, en la actualidad esta tecnología se encuentra constantemente en desarrollo, en la cual se ha venido puliendo con algoritmos mejor implementados que permiten realizar tareas más complejas y obtener mejores resultados.

Conforme avanza esta tecnología cada día más las empresas ven una buena inversión al comprar sistemas de visión artificial, ya que estas presentan mayor rapidez y obtienen mayor precisión al realizar procesos de control de calidad, categorización y ensamblaje de un determinado producto, ahorrando valiosos recursos para ser ejecutados en otras tareas de mayor importancia. Empresas como CVA (Control y Visión Artificial), JasVisio, Infaimon, TechDesign, etc; son empresas dedicadas a la venta de software o productos para la implementación de visión artificial aplicada a la industria y seguridad. Presentan una amplia gama de productos desde cámaras, sensores, Visión 3D, sistemas de control, máquinas de ensamblado, etc. Todas estas adaptables a las necesidades del cliente y completamente automatizadas.

Entre uno de los proyectos que comienzan a fortalecer el ámbito de la robótica se resalta el EYESHOTS (Percepción tridimensional heterogénea entre diversos fragmentos visuales). Este tiene como objetivo principal construir robots capaces de ver y tomar conciencia de su entorno además de recordar los contenidos de las imágenes vistas y actuar en consecuencia. Su funcionamiento consistirá en la representación del entorno, retener las imágenes pertinentes y aprender, apuntando hacia los objetos que puede interactuar, todo este procedimiento será almacenado en memoria para que luego se ejecute sin necesidad de volver a visualizar el entorno.

Esta tecnología no solo se enfoca en el campo industrial o robótico, también en el campo médico se encuentra efectuando estudios para la realización de prótesis, entre una de ellas se encuentra el microchip denominado Argus creado por el Dr. Liu MacKenzie. Esta prótesis se encuentra en proceso experimental y se ha probado ya en pacientes obteniendo resultados óptimos. Consiste en un dispositivo electrónico diseñado para ser implantado en pacientes que han quedado ciegos por culpa de

enfermedades como la retinitis pigmentosa y la degeneración macular relacionada con el envejecimiento. Este se encarga de estimular las neuronas retínales, retransmitiéndoles señales visuales que al microchip le llegan provenientes de una pequeña cámara de video integrada en unas gafas de sol. Se encuentra en constante mejora y ya se ha desarrollado un nuevo modelo mucho más avanzado que es el Argus II y presenta mejoras en cuanto a su tamaño, conducción de datos inalámbricos entre el implante y las gafas, además de estimular cuatro veces más neuronas que el prototipo original.

En el entretenimiento se han desarrollado nuevas tecnologías para mejorar la interacción del producto con el usuario, como en el caso del Smartphone Samsung Galaxy S4 y S5 que implementa la nueva tecnología de “Eye Tracking” que consiste en detectar los ojos del usuario con la cámara frontal del dispositivo ya sea para realizar scroll o apagar la pantalla, además de pausar un video si no se lo está observando. También en el ámbito de entretenimiento se puede destacar un nivel más avanzado en sistemas de detección, que utilizará el nuevo “Kinect 2.0”, detectando hasta 16 usuarios simultáneos, además de reconocer movimiento de dedos y expresiones faciales.

En donde más se ha notado la visión artificial y en la que con mayor facilidad se ha ido acoplado, es en el campo de la seguridad. En ella se ha venido desarrollando maneras de mejorar los sistemas de detección para reconocer con mayor precisión y rapidez. El “BioSurveillance Next” se presenta como el sistema de reconocimiento facial más veloz hasta el momento, es cuarenta veces más rápido que la tecnología de reconocimiento facial basada en CPU, es capaz de procesar en tiempo real video de alta definición grabado o en vivo a velocidades de 150 fps. El sistema es totalmente escalable, usa tecnología GPU (Graphic Processing Unity) y está diseñado para identificar simultáneamente múltiples rostros en movimiento, lo que lo hace ideal para detectar a un determinado individuo en multitudes.

Para finalizar en esta sección la visión artificial en la época actual ha comenzado a presentar uno de los más importantes avances de la tecnología y pretende proyectarse con mayor fuerza en el futuro permitiendo realizar tareas que hubieran sido inimaginables en tiempos pasados. Sus aplicaciones en distintos campos de la ciencia permiten aprovechar las bondades que esta brinda, y usarlo de

la mejor manera para solucionar muchos problemas o reducir tiempos y defectos que ocurrían con la interacción humana. Si bien esta no vence a la visión humana, intenta alcanzarla con cada vez mejores y más eficientes algoritmos que complementan y fortalecen a una inteligencia artificial cada vez mejor implementada.

## **Conclusión**

El capítulo trató los conceptos de visión artificial que se manejarán a lo largo del desarrollo del sistema. Estos conceptos se definirán más detalladamente conforme se desarrollen los siguientes capítulos. Adicionalmente se intenta demostrar que la tecnología de la visión artificial es importante en los diferentes campos de la ciencia y continua desarrollándose progresivamente con el tiempo.

## Capítulo 2

### Análisis de la situación actual

#### Introducción

El objetivo principal de este proyecto de investigación es la solución de todos aquellos problemas que pueden ser causados por un sistema tradicional de ingreso vehicular. Para poder presentar las características del sistema es necesario explicar el motivo por el cual fue creado y así tener una aplicación en el mundo real.

En este capítulo se identificará todos aquellos errores que son considerados de principal importancia y que pueden ser solucionados con un sistema eficiente de ingreso vehicular. El principal ámbito de investigación estará orientado a solucionar los problemas de ingreso en el parqueadero de la Universidad del Azuay donde se analizarán todos aquellos inconvenientes que surgen con una mala gestión identificando sus respectivas causas y efectos.

#### 2.1. Estudio de la problemática actual

La Universidad del Azuay al contar con grandes cantidades de personal y flujo elevado de vehículos llegan a presentar inconvenientes al no mantener un sistema de registro adecuado. Entre los problemas más importantes se describen:

- Ingresos de personal no autorizado.
- Falta de control de acceso.
- Ausencia de historial de ingresos.
- Demoras en acceso.

El inconveniente que mayor atención requiere, es el ingreso de personal no autorizado que influye en la seguridad de todos los usuarios del parqueadero. Este representa un gran problema en los procesos de autorización y denegación de acceso, es decir, existen casos en que usuarios no identificados acceden al parqueadero y usuarios que si están autorizados no son reconocidos para el ingreso. Los vehículos no autorizados, son aquellos que no pertenezcan al personal que labora dentro de la

institución y los vehículos denegados son aquellos que si bien pertenecen al personal de la institución, no se encuentran registrados dentro de la base de datos.

Adicionalmente la ausencia de un registro ordenado implica un escaso control de los vehículos, ya que no se conoce la cantidad de los mismos que pertenecen a cada usuario de la institución, por lo que no pueden ser identificados correctamente por el encargado.

Otro inconveniente de menor brevedad está identificado por demoras en el ingreso de vehículos, el usuario tiene que esperar un determinado tiempo hasta que el encargado finalice sus actividades y permita el ingreso. Este tiempo no es muy prolongado pero causa un retraso y una variación en el ingreso de los usuarios.

Los problemas antes mencionados tienen consecuencias directas sobre una gestión en el ingreso de vehículos ocasionando una desorganización en el sistema de acceso y un inconveniente en términos de seguridad, por lo que será necesario estudiarlos más a fondo realizando una investigación de sus causas y efectos.

### **2.1.1. Causas del problema**

Es necesario identificar la raíz del problema mediante la investigación de los motivos que los ocasionan. Entre las más notables se encuentran:

- **Dependencia de un encargado para el ingreso del vehículo.**- Existe una dependencia constante del supervisor o encargado para realizar un acceso, debido a ello, si el mismo presenta alguna ocupación el proceso de ingreso se detiene hasta que éste vuelva a estar disponible.
- **Ausencia de registros de acceso.**- En la institución no se lleva ningún registro de acceso manual o automático, debido a ello, no se generan reportes ni pruebas tangentes de los ingresos, por lo que no se comprueba ningún registro de los acontecimientos diarios en el parqueadero. Esto puede ser crítico al ocurrir algún incidente dentro de la institución y no tener registro de vehículos que pudieran ser sospechosos.

- **Problemas de identificación.**- Este acontecimiento se deriva de las causas anteriores ya que se permite el acceso a personal no autorizado o al contrario se podría denegar el acceso a personal calificado pero no registrado o reconocido. Este proceso genera errores operativos ya que dependerá exclusivamente del criterio del supervisor en dar acceso o denegarlo.

Todas estas causas generan problemas en seguridad y gestión en el acceso que a largo plazo pueden afectar directamente a la universidad. También se puede observar que la decisión de tener a un encargado que gestione directamente el ingreso no es del todo acertada y más aún si no se tiene un registro organizado, ya que el factor humano en términos de seguridad puede fallar o puede ralentizar el proceso en situaciones claves de ingreso.

### 2.1.2. Efectos del problema

Luego de analizar las causas más comunes que llevan a efectuar los problemas en el ingreso de vehículos, es necesario identificar a donde pueden llevar estos problemas y hasta qué punto pueden afectar al desempeño de la institución. Entre los principales problemas investigados se describen:

- **Problemas de seguridad.**- Los problemas de detección pueden llevar a permitir el acceso de usuarios que no forman parte de la institución. También existe la posibilidad que pueda acceder personas que lleguen a realizar actos perjudiciales en las instalaciones de la universidad.
- **Desconocimiento de actividades en el parqueadero.**- Debido a los turnos rotativos de los encargados, estos desconocen las actividades realizadas anteriormente para estar preparados ante incidencias de seguridad o control de acciones que deben realizarse.
- **Desorganización en el parqueadero.**- Al tener un espacio limitado exclusivamente para el personal docente y administrativo, si uno o varios usuarios no permitidos ingresan al parqueadero, podrían ocupar espacios designados al personal de la universidad. Adicionalmente si no se tiene un

registro de vehículos definido no se podrá administrar correctamente los espacios ni el personal que los ocupará.

- **Identificación limitada**.- Al no tener un patrón fijo de reconocimiento, se limita el acceso al criterio del encargado. Este consiste en identificar un rostro conocido o características de un vehículo frecuente, sin tener ningún método de consulta para verificar al usuario. Esto puede ser perjudicial para usuarios nuevos, o encargados que desconocen al personal.

Es necesario un mayor conocimiento de estos efectos muy importantes dentro del análisis de la problemática, por lo cual se enfocará hacia los usuarios principalmente afectados y su reacción ante tales circunstancias.

#### **2.1.2.1. Principales afectados**

Los problemas ocasionados en el ingreso pueden afectar a todo tipo de usuarios pertenecientes a la universidad. Puntualmente se destacará aquellos que tienen mayor influencia en el uso del parqueadero. A continuación se listará cada usuario identificado:

##### **Docentes y personal administrativo**

Estos son los usuarios que más utilizan el parqueadero ya que requieren un número elevado de accesos diarios. Estos se verán perjudicados directamente por cualquier inconveniente que surja en el ingreso. Los problemas que afecta a los docentes son los siguientes:

- Inseguridad en el parqueadero.
- Problemas de reconocimiento en el ingreso.
- Espera prolongada en identificación.
- Espacios del parqueadero ocupados.

##### **Encargados de seguridad**

Estos usuarios también se ven afectados por inconvenientes en el sistema actual de acceso al parqueadero, al no contar con las herramientas adecuadas se

dificulta su decisión al permitir o no el ingreso de un vehículo. Entre los problemas que afectan al encargado se describen:

- Problemas al identificar usuarios de ingreso.
- Tiempo enfocado al acceso interrumpiendo otras actividades que debe realizar.
- Ausencia de información para consulta de empleados y accesos.

### **La institución**

Todos los inconvenientes anteriormente mencionados tienen incidencia directa con la institución y sus ocupantes, ya que el ambiente del parqueadero se desarrolla dentro de la misma universidad. Estos problemas aunque en menor medida afectan a la organización de la institución y como esta se involucra en temas de seguridad y gestión.

### **2.2. Delimitación del problema**

Luego de determinar y analizar todos aquellos problemas que surgen por una deficiente organización y gestión de accesos, se establece la necesidad de mejorar los procesos de ingreso de vehículos para así poder mitigar dichos errores, optimizando el ingreso de automotores y corrigiendo las fallas de seguridad, que no hacen más que ocasionarle problemas a la institución. Para poder plantear una solución primero se debe enfocar en los aspectos que deben corregirse:

- Permitir identificar el vehículo de una manera más rápida y organizada.
- Reducir al mínimo la intervención del guardia o supervisor.
- Permitir un registro más completo para evitar problemas de seguridad
- Automatizar el sistema de ingreso para aumentar la eficiencia.

## **Conclusión**

Todos los aspectos mencionados ponen en evidencia la necesidad de plantear una solución automatizada en el proceso de ingreso vehicular y así efectuar dicha operación de una manera rápida, eficiente y completa. Estos problemas si no son manejados de la forma correcta seguirán ocasionando inconvenientes a la institución, que como se describió anteriormente afecta a todo su personal.

## Capítulo 3

### Estudio de herramientas de visión artificial.

#### Introducción

En el capítulo anterior se ha detectado todos aquellos inconvenientes que se presentan ante un método de registro vehicular no adecuado, además se ha señalado los de mayor importancia en los que el sistema se debe enfocar. Ahora se establece como principal prioridad listar todas aquellas soluciones a estos inconvenientes y como pueden ser implementadas para que representen una solución viable.

#### 3.1. Soluciones propuestas a los problemas mencionados

Se puede deducir a simple vista, que la solución inmediata y más viable a la problemática mencionada, es realizar un proceso automatizado de ingreso, ya que el objetivo principal es disminuir tiempos, dependencias externas y aumentar la seguridad en el acceso.

Este método automatizado deberá poseer ciertas características para representar una solución viable a cada uno de los errores mencionados en el capítulo anterior. Estas características son:

- **Reconocimiento inmediato del vehículo:** La detección del vehículo debe realizarse en el menor tiempo posible, para lo cual, se deberá utilizar un sistema computacional que permita procesar rápidamente el ingreso basándose en un criterio de reconocimiento. Este criterio corresponde a la placa del vehículo que servirá como identificador único para realizar los procesos de acceso y registro.
- **Validación en tiempo real del vehículo:** La información que permita autorizar el acceso deberá realizarse de una manera rápida y oportuna, obteniendo una respuesta inmediata.
- **Registro completo:** El registro debe contener toda la información pertinente al acceso (fecha de ingreso, nombre del propietario, cargo, características del

vehículo. etc.), para que esta información sea procesada y permita realizar consultas en cualquier momento.

- **Sistema independiente:** El sistema debe requerir la menor interacción externa y debe contribuir a la toma de decisiones el momento de un ingreso. Debe requerir intervención del supervisor solamente en casos excepcionales.

Las 4 características nombradas con anterioridad son fundamentales para construir una solución lógica a la problemática analizada. A continuación se detallará estas características con el fin de indicar las herramientas y procesos que serán necesarios para realizar el sistema de automatización de ingreso:

- **Uso de la tecnología de visión artificial:** Es necesario un método visual que permita detectar de manera rápida un vehículo, para ello se emplea las técnicas de visión artificial que identifican el patrón de la placa dentro del entorno capturado por una cámara. Esta tecnología simula la visión humana permitiendo captar objetos dentro de un entorno para luego ser procesados y analizados.
- **Procesamiento, identificación y validación:** El proceso de verificación debe presentar un mecanismo de reconocimiento sólido y confiable, es decir, debe implementar un conjunto de procesos que permitirán depurar y digitalizar la imagen de la placa para que sea más fácil y rápida identificarla. Se emplearán algoritmos que digitalicen los caracteres de la placa validando que no alteren la estructura de la misma, con el fin de brindar mayor seguridad a la validación.
- **Administrador completo:** Todos los procesos y herramientas serán unificadas en un solo sistema que presentará un administrador con todas las características necesarias para su uso. El sistema empleará módulos de vehículos, módulos de empleados, registros de acceso, etc.
- **Tecnología y herramientas autónomas:** El resultado final del sistema consistirá en enviar una señal de aprobación o denegación al ingreso del vehículo, pero este resultado solo será mostrado si se ha realizado todos los

pasos anteriores en el sistema, por lo que cada paso con cada herramienta enviará sus resultados de manera automática. Desde la toma de la captura hasta el resultado final no requerirá ninguna intervención de factores externos. En situaciones fuera de los parámetros comunes, se implementará un acceso manual y así evitar que el sistema pueda ocasionar problemas en el ingreso.

Con el objetivo de profundizar más sobre la visión artificial y su funcionamiento, es necesario estudiar las herramientas que formarán parte de esta tecnología y seleccionar aquellas que más se adapten a las soluciones mencionadas.

### **3.2. Herramientas seleccionadas para el sistema**

Las herramientas que formarán parte del sistema se dividirán en tres categorías: reconocimiento de patrones, reconocimiento óptico de caracteres (OCR) y procesamiento de imágenes. A continuación se especificará las herramientas más importantes de cada categoría:

#### **3.2.1. Herramientas de reconocimiento de patrones**

Un sistema de reconocimiento de patrones se encarga de recolectar características descriptivas del elemento a ser identificado o reconocido, para que posteriormente aplicando un método de extracción se pueda obtener información para la toma de decisiones. Esto se logra eliminando la información o datos irrelevantes, dejando solo muestras positivas a ser identificadas. Luego de la etapa de reconocimiento, se procede a la toma de decisiones, esta consiste en evaluar las muestras con la captura realizada, y así determinar los patrones reconocidos del objeto. (Carrasco, 2003)

La herramienta más destacada dentro del ámbito de visión artificial se denomina OPEN CV. Esta poderosa herramienta contiene algoritmos avanzados de entrenamiento denominado haartraining que permite establecer parámetros del objeto para ser detectado de una manera rápida y precisa. Debido a que esta herramienta es la base de todas aquellas que emplean la tecnología de visión artificial será sin ninguna duda la empleada en la realización del sistema de reconocimiento.

## OPEN CV

“OpenCV basado en licencia BSD y orientado para uso académico y comercial fue diseñado para la eficiencia computacional y con un fuerte enfoque en las aplicaciones en tiempo real. Escrito y optimizado en C / C ++, la biblioteca puede tomar ventaja de procesamiento multi-core. Habilitado con OpenCL, puede aprovechar la aceleración de hardware de la plataforma de computación. Adoptado en todo el mundo, OpenCV tiene más de 47 mil personas de la comunidad de usuarios y el número estimado de descargas que superen 9.000.000. Su alcance va desde el uso de arte interactivo, a la inspección de minas, mapas de puntos en la web o a través de la robótica avanzada”. (*Open CV, 2013*)

Debido a su estructura modular está completa herramienta incluye una gran cantidad de librerías que permiten realizar tareas como mapeado, análisis y procesamiento de imágenes. Entre las más destacadas se pueden mencionar (Open CV, 2013):

- **Core:** Es un módulo compacto que define las estructuras de datos básicas, incluyendo matrices multidimensionales y las funciones básicas usadas por los demás módulos.
- **Imgproc:** Es un módulo de procesamiento de imágenes que incluye filtrado de imágenes lineales y no lineales, transformaciones geométricas de imágenes (cambio de tamaño, afinación de perspectivas, tabla genérica basada en reasignaciones), conversión del color del espacio, histogramas, etc.
- **Video:** Es un módulo de análisis de vídeo que incluye estimación del movimiento, la sustracción del fondo, y algoritmos de seguimiento de objetos.
- **Calib3d:** Basada en algoritmos geométricos de múltiples vistas, calibración de la cámara, estimación de la posición de objetos, algoritmos de correspondencia, y elementos 3D de reconstrucción.
- **Features2d:** Son detectores de características sobresalientes sobre objetos.
- **Objdetect:** Detección de objetos de una clase predefinida (por ejemplo: ojos, cara, expresiones faciales, gente, carros, etc).
- **Highgui:** Es una interfaz de captura de video, contiene codecs de imágenes y de videos, así como una interfaz de usuario simple y fácil de usar.

- **Gpu (Graphics Processing Unit):** Algoritmos acelerados por GPU de diferentes módulos OpenCV.

Open CV es compatible con varias herramientas de programación como Visual Studio, Java y herramientas derivadas como: processing, openframework, etc. Incluso puede ser utilizada para desarrollo móvil tanto en Android como en IOS.

### 3.2.1.1. Características

Open CV es la herramienta más completa de visión artificial por ello posee un sin número de características que describen su potencial, entre las más importantes se pueden destacar (Open CV, 2013):

- **Manejo global de clases con “CV Namespace”:** Open Cv posee comandos que permiten acceder a todas las clases y funciones de su biblioteca. Solo es necesario usar el especificador “cv:.” o “*namespace cv*” para acceder a todas y cada una de las funcionalidades de esta potente herramienta.
- **Manejo de memoria automática:** El manejo de estructuras de datos realizado por procedimientos y funciones contienen sus propios destructores que se encargan de gestionar y liberar la memoria cuando lo requieren. Estos destructores no siempre liberan los buffers de memoria al finalizar una llamada a los datos, sino que comparten los mismos asociándolas a un contador. Por cada relación con el buffer el contador incrementará. La memoria será liberada solo si su contador asociado llega a cero, esto significará que ninguna estructura de datos adyacente comparte o utiliza el mismo buffer, es decir, cuando se realiza un proceso de copiado con una función, en realidad no se encuentran los datos duplicados sino que su contador de referencia es incrementado indicando que existe otro que usa esa misma información. Con esta función se permite el manejo eficiente de la memoria para realizar procedimientos con mayor rapidez
- **Asignación automática de los datos de salida:** OpenCV libera la memoria de forma automática, así como asigna automáticamente la memoria para los resultados de salida.

- **Saturación aritmética:** Como una biblioteca de visión por computador, OpenCV puede llegar a manejar una gran cantidad de píxeles de la imagen que a menudo están codificados en un formato compacto, 8 o 16 bits por canal, es decir tienen un rango de valor limitado. Por otra parte, ciertas operaciones en las imágenes, como conversiones de espacio de color, brillo / ajustes contraste, la nitidez, la interpolación compleja (bi-cúbico, lanczos), puede producir valores fuera de los rangos válidos. Para resolver este problema, se utiliza la denominada saturación aritmética. Por ejemplo, al almacenar de  $r$ , el resultado de una operación, a una imagen de 8 bits, se encuentra el valor más cercano dentro del rango de 0 – 255, esta se describe en la siguiente formula:

$$I(x, y) = \min(\max(\text{round}(r), 0), 255)$$

- **Uso de plantillas limitadas:** Las plantillas o “Templates” son una gran característica de C que permite la implementación de estructuras de datos y algoritmos muy potentes, eficientes y seguros.

Sin embargo, el uso excesivo de plantillas puede aumentar drásticamente el tiempo de compilación y el tamaño del código, además, es difícil separar una interfaz y la aplicación cuando se usa exclusivamente plantillas. Esto no tendría problema en algoritmos básicos, pero perjudica a las bibliotecas de visión por computador que en un único algoritmo puede abarcar miles de líneas de código. Debido a ello OpenCV se basa en el polimorfismo y el uso limitado de plantillas basado en despacho de tiempo de ejecución. Esto no solo soluciona problemas de codificación sino también permite de una manera más simple el enlace con otros lenguajes.

- **Inputarray y ouyputArray:** Muchas funciones OpenCV procesan densas matrices numéricas bidimensional o multidimensional. Para evitar muchos duplicados en la API (Application Programming Interface), se han introducido clases especiales denominadas "proxy". La clase base "proxy" es InputArray y se la utiliza para el paso de matrices de sólo lectura en una entrada de la función. El derivado de InputArray clase outputArray se utiliza para especificar una matriz de salida para una función. Esta clasificación la hará

automáticamente el sistema para que el usuario no tenga que preocuparse por optimizaciones manuales.

- **Control de errores:** OpenCV utiliza excepciones para señalar los errores críticos. Cuando los datos de entrada tiene un formato correcto y pertenecen a la gama de valor especificado, pero el algoritmo no puede tener éxito por alguna razón (por ejemplo, el algoritmo de optimización no convergen), devuelve un código de error especial. El código es entregado en instancias de una variable que puede ser manejado correctamente a otras bibliotecas externas como C++. Esta variable generada por diferentes funciones comprueba el estado de operación y retorna una excepción cuando no se cumple cierta condición. Si es necesario se podrá implementar una sentencia TRY para capturar la excepción.
- **Multi-threading and re-enterability:** La implementación actual OpenCV es completamente re-editable. Es decir, la misma función, el mismo método constante de una instancia puede llamarse en diferentes hilos de procesamiento, realizando dichas funciones de manera paralela sin ningún inconveniente.

### 3.2.1.2. Aplicación en el sistema

La aplicación de Open CV está determinada por su propia funcionalidad, es decir, al ser una herramienta de visión artificial permitirá realizar las tareas de reconocimiento de la placa vehicular.

La implementación de esta herramienta será uno de los procesos más importantes en el sistema de reconocimiento ya que contribuye con más del 50% de su funcionalidad y es el proceso que ejecutará el resto de etapas en el sistema. Por consiguiente, ésta herramienta correctamente implementada realizará las tareas de identificación, detección y validación de la placa para luego ser procesada y digitalizada por la herramienta de OCR.

Open CV proporcionará una herramienta denominada trainingcascade que permitirá realizar el entrenamiento de la placa, con ello se podrá preparar al sistema para que proporcione la detección del objeto deseado.

### 3.2.1.3. Requerimientos

Es necesario tener correctamente definidos todos los componentes tanto de software como de hardware que requiere Open CV para un funcionamiento óptimo.

#### 3.2.1.3.1. Software

Al ser una herramienta muy versátil y adaptable Open CV está disponible para los sistemas operativos más populares en la actualidad: Windows, Linux y Mac.

Para Linux y MAC se requerirá de la herramienta “CMAKE”, esta es una herramienta de generación o automatización de código que permite construir, probar y empaquetar software.

Open CV contiene sus propias librerías que permiten una gran compatibilidad con la mayoría de sistemas y lenguajes de programación.

#### 3.2.1.3.2. Hardware

Debido a que Open CV presenta 2 etapas en su proceso de detección, es necesario separar sus requerimientos ya que cada etapa difiere en la manera que será utilizada:

- **Para la etapa de entrenamiento:** será necesario un equipo más potente ya que la fase de entrenamiento consume un número elevado de recursos al procesar una gran cantidad de imágenes y estos influirán directamente en el tiempo de finalización del proceso. Para ello es recomendable:
  - Procesador mínimo de 2 núcleos con frecuencia de 2.4Ghz o más.
  - 2GB de memoria RAM.
  - 1GB de espacio en el disco duro.

- **Para la etapa de ejecución:** no influirá demasiado el poder de procesamiento y los recursos deberán ser moderados. Pero se debe considerar que influirá en la velocidad de detección en el ingreso. Para ello se recomienda:
  - Procesador mínimo de 1 núcleo con frecuencia de 2.0 Ghz o más.
  - 1GB de memoria RAM.
  - 1GB de espacio en el disco duro.
  - Cámara estándar.

### **3.2.2. Herramienta de digitalización de caracteres (OCR)**

La tecnología OCR comprende un conjunto de técnicas y algoritmos que al complementarse entre sí, permiten distinguir de manera automática los diferentes caracteres alfanuméricos existentes. Su funcionamiento, no consiste en determinar un carácter a simple vista, sino más bien, realiza una comparación de características similares en una biblioteca previamente entrenada. Por lo tanto se podría decir que la técnica de OCR es un reconocimiento de patrones, pero cuyos procesos están más orientados a la detección de símbolos. (Javier Cano, 2010)

Se debe recalcar que esta técnica no es 100% precisa ya que dependerá de la eficacia del algoritmo utilizado por cada herramienta. Por lo tanto su método de aplicación será esencial para el funcionamiento correcto del sistema a realizar.

La tecnología OCR se encuentra distribuida en varios tipos de herramientas desde aplicaciones de lectura hasta digitalizadores de texto, por tal motivo, es necesario buscar una herramienta que posea un enfoque orientado al ámbito del sistema y que se acople al mismo para obtener los resultados deseados. Tesseract presenta uno de los algoritmos de reconocimiento más precisos, además al ser la única herramienta de software libre que puede ser modificada y utilizada para múltiples fines, esta es la que se seleccionará para el sistema.

## Tesseract

Originalmente desarrollado por Hewlett Packard en el año de 1985 en lenguaje C, para posteriormente determinarlo como software libre en el 2005, luego en el año 2006 fue tomado por Google y migrado a C++. Actualmente se encuentra disponible para Windows Mac y Linux, posee soporte para varios idiomas incluidos el español, actualmente su última versión 3.0 permite la detección de varios tipos de imágenes.

Se le considera una de las herramientas más precisas en la actualidad llegando a un porcentaje de certeza del 97.6% por palabra. Esta herramienta fue diseñada para que sirva de motor de OCR ya que no presenta interfaz gráfica y todos sus procesos se los realiza mediante comandos, por lo que requerirá un nivel de conocimiento avanzado para su utilización. Todas estas características complementadas por una buena interfaz se transforman en una potente herramienta que brinda una mayor comodidad y facilidad de manejo para el usuario. (Smith, 2007)

Las interfaces que utilizan el motor Tesseract son: Free OCR, gImagen, Text Ripper, Web OCR, Mobile OCR, OCR Feeder, etc.

### 3.2.2.1. Características

Entre las principales características que hacen de Tesseract uno de los mejores motores de OCR se mencionan (Smith, 2007):

- **Algoritmo de búsqueda de línea (Line Finding):** Este proceso consiste en identificar los puntos continuos y direccionados que conforman una línea de texto ya sea en entornos curvos o corrugados, mediante un proceso de filtro de manchas y construcción de líneas. El filtrado de manchas consiste en dividir en capas aquellos puntos o manchas que no forman parte del tamaño estándar del texto o se encuentran sobrepuestas al mismo. Luego se procede a construir las líneas de texto organizando todos los caracteres en x coordenadas, para así identificar las líneas correspondientes.
- **Búsqueda de “Líneas Base”:** Tesseract presenta un algoritmo de ranura cuadrática que permite detectar las “líneas base” de texto ya sea en entornos

curvos o inclinados. Este algoritmo consiste en organizar los puntos de texto en bloques, identificar progresivamente la ubicación de los segmentos de texto y luego establecer su orientación.

Volume 69, pages 872-879.

Figura 3.10 Detección de líneas base en texto inclinado (Smith, 2007)

- **Eliminación de letras unidas o cortadas:** Tesseract divide en celdas los caracteres de cada palabra y permite identificar aquellos que se encuentran unidos o cortados.



Figura 3.11 Detección de caracteres en palabras unidas (Smith, 2007)

- **Búsqueda proporcional de palabras:** Para una mayor precisión Tesseract se encarga de identificar y corregir las palabras que se encuentran con caracteres espaciados o cuyos márgenes se encuentran demasiado unidos o separados y así evitar su mala detección o pérdida de significado.

**of 9.5% annually while the Fed-  
erated junk fund returned 11.9%**

***fear of financial collapse,***

Figura 3.12 Problemas de espaciado en palabras (Smith, 2007)

- **Separación de caracteres unidos:** Cuando la detección de un carácter determinado no es satisfactoria. Tesseract analiza las superficies de los mismos para determinar puntos de unión o superficies unidas para así separarlas y determinar cada carácter correctamente.

- **Asociación de caracteres rotos:** Si aún con la división de caracteres estos no son reconocidos, el programa se encarga de asignar estos a un “Asociador” que establecerá pesos de acuerdo a sus combinaciones. Al final se asignarán los pesos que coincidan para formar la letra o palabra.
- **Clasificador de caracteres estáticos:** Para el proceso de entrenamiento Tesseract realiza un proceso de clasificación que consiste en establecer un vector temporal de clases que pudieran encajar con el carácter a entrenar, es decir un vector de bits de clases seleccionadas y otro vector de bits que acumula la suma de características de cada clase.

Luego de obtener el vector prototipo, se realiza la comparación de este con las características del carácter actual, la similitud real entre ellos será computarizada. Cada clase del carácter prototipo es representado por una expresión lógica de suma de producto con cada término denominado “configuración”, por lo que la distancia de proceso de cálculo mantiene un registro del total de evidencia de similitud de cada función en cada configuración en cada prototipo. La mejor distancia combinada, se calcula a partir de la suma de características y evidencias del prototipo, siendo esta la mejor sobre todas las configuraciones almacenadas de la clase.

- **Entrenamiento de datos:** Debido al sistema de clasificación y asociación de caracteres rotos, tesseract no requiere de demasiadas muestras de entrenamiento. Su clasificador fue realizado solo con 20 muestras de 94 caracteres con un solo tamaño de letra con 4 atributos (normal, negrita, cursiva y negrita cursiva) a comparación con otros clasificadores usan 100 tipos de fuentes con más de 1175000 muestras.
- **Análisis lingüístico:** Cuando un módulo de reconocimiento para una palabra está considerando una nueva detección el módulo lingüístico selecciona la mejor cadena de caracteres de acuerdo a diferentes categorías (top palabra más frecuente, top palabra de diccionario, top palabra numérica, top palabra en minúscula, top palabra en mayúsculas, top palabra escogida por el

clasificador). La decisión final es simplemente cuya distancia de calificación sea la más baja, donde cada una de las características mencionada es multiplicada por una constante diferente. Así logrará mayor segmentación para una rápida detección de cadenas de texto.

### 3.2.2.2. Aplicación en el sistema

La herramienta OCR tendrá como funcionalidad principal detectar el conjunto de caracteres que conforman la placa vehicular. Luego el resultado tendrá que ser comparado con la información disponible en una base de datos.

Este proceso tendrá un alto grado de importancia, ya que será el resultado final para una posterior verificación y registro. Además deberá contener una estructura sólida para evitar fallas en la detección o identificar caracteres erróneos. La estructura de este proceso se manejará en etapas, estas se destacan a continuación:

**Información de entrada:** Para esta etapa la imagen de la placa deberá estar en las mejores condiciones posibles; es decir correctamente cortada y editada.

**Proceso de transformación:** La herramienta OCR se encargará de detectar cada carácter y digitalizarlo.

**Información de salida:** El resultado final constara en la placa completamente digitalizada y se almacenará en la base de datos como un registro para posteriormente ser comparado.

Tesseract realizará un proceso previo de entrenamiento similar a open cv, pero con la particularidad de que se lo realizará con un alfabeto de caracteres de la placa, esto con el fin de que se determinen características como el tipo de letra, el color y estructura.

### 3.2.2.3. Requerimientos

La herramienta de OCR tesseract tiene 3 tipos de requerimientos: software, hardware y elementos de detección.

### 3.2.2.3.1. Elemento de detección

Tesseract requiere una imagen para realizar el proceso de digitalización. Esta imagen deberá poseer el texto a identificar con las siguientes características:

- Imagen binarizada.
- Texto claro y completo.
- Formato definido.
- Lenguaje previamente entrenado.

### 3.2.2.3.2. Software

Tesseract no requiere ningún componente de software adicional para su funcionamiento y se encuentra disponible para Linux, Windows y MAC

### 3.2.2.3.3. Hardware

Tanto para la etapa de detección como la de entrenamiento sus requerimientos de hardware son mínimos:

- Sistema Operativo: Windows, Linux o Mac
- Procesador: 1 Ghz o mas
- RAM: 512mb o mas
- Disco Duro: 500mb o más.

## 3.2.3. Herramienta de procesamiento de imágenes

El procesamiento de imágenes consiste en un conjunto de técnicas que toman como condición una imagen inicial y la transforman mejorando sus características y adecuándola para obtener los resultados deseados (Universidad Simon Bolivar, 2013). En esta etapa es necesario resaltar 2 técnicas que serán utilizadas en el sistema:

**Filtrado:** Es una técnica que consiste en transformar una imagen completa o a una parte de ella, aplicando un operador para obtener un efecto especial. Los principales filtros que se pueden realizar con la imagen son:

- Suavizar la imagen
- Eliminar ruido
- Transformar a escala de grises
- Transformar a binario

**Recorte:** método que consiste en remover ciertas partes de la imagen no necesarias para ser mostradas en el resultado final.

A continuación se describirán las características de las herramientas más comunes para el procesamiento de imagen con el fin de determinar la más eficiente para el sistema de detección.

### 3.2.3.1. Características

#### **Open CV Imgproc:**

Se han nombrado todas aquellas características que Open CV posee en relación a la visión artificial, pero debido a su versatilidad esta herramienta posee muchas más. Entre una de ellas se encuentra un módulo denominado Imgproc, este permite la manipulación y procesamiento de imágenes mediante varias funciones todas ellas incluidas en el paquete de instalación de Open CV.

Imgproc posee varias librerías encargadas de la edición y procesamiento de imágenes entre ellas se describen: (Open CV, 2013):

- **Transformaciones geométricas:** Realiza la transformación geométrica de imágenes 2d permitiendo que el contenido no cambie, en vez de ello, la cuadrícula de píxeles se deforma obteniendo la imagen de destino. Posee funciones simples como **resize()** o complejas como **warpPerspective()** que mapea una imagen desde cierta perspectiva.
- **Filtrado de imágenes:** Contiene varias funciones y tipos cuyo objetivo principal es realizar filtrados lineales o no lineales sobre una imagen 2d. Esto permite el cálculo de la posición de cada píxel en la imagen de origen, en función a sus píxeles vecinos. Cada píxel resultante se ubicará en la misma posición del original obteniendo una imagen del mismo tamaño.

- **Extrapolación de imágenes ante píxeles no existentes:** Permite replicar píxeles fuera de la imagen en función del borde de la misma o colocando un color constante. Entre las funciones más utilizadas se definen: **medianBlur()** para dar un efecto de desenfoque, **erode()** para dar efectos de saturación o **dilate()** para dar efectos de desgastado de la imagen.
- **Creación de histogramas:** Los histogramas Son un conjunto de funciones que permiten obtener información de la imagen como; su intensidad o distribución; y cuantificarla en un número de píxeles determinado permitiendo tener un análisis más detallado.

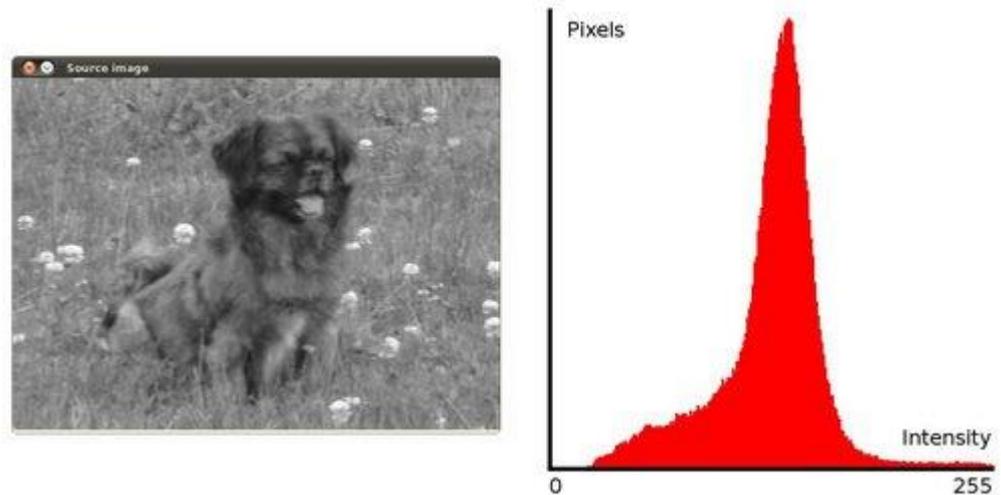


Figura 3.13 Histograma de una imagen (Open CV, 2013)

- **Análisis estructural y descripción de formas:** Conjunto de funciones que definen y manipulan aspectos de contornos y formas. Estas permiten trabajar sobre estos contornos para dibujar o identificar objetos sobre una imagen.

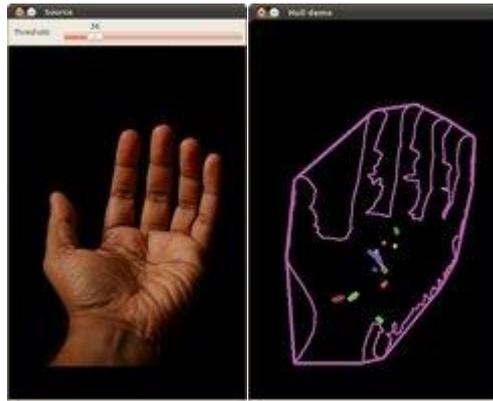


Figura 3.14 Detección de contornos de un objeto dentro de una imagen. (Open CV, 2013)

- **Otras transformaciones:** Corresponde a un conjunto de funciones que son utilizadas regularmente para cambiar aspectos de apariencia en la imagen. Entre los más importantes se listan:
  - **cvtColor ():** Función que sirve para cambiar un espacio de color a otro. Los colores son manejados en canales RGB o BGR (Similar a RGB pero invertido). Maneja aproximadamente entre 150 diferentes tipos de cambios de colores entre ellos COLOR\_BGR2GRAY que cambia una imagen a blanco y negro
  - **threshold():** Permite la segmentación de una imagen. Este método consiste en la separación de regiones de una imagen mediante la variación de intensidad de píxeles de un objeto con su entorno. Esta variación se verá reflejada en una imagen asignando un valor (color) a cada región definida y así identificar sus objetos.

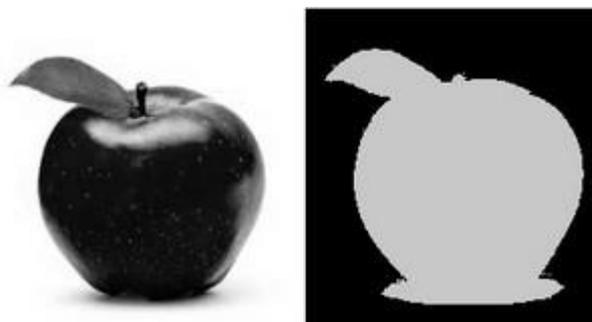


Figura 3.15 Segmentación de un objeto. (Open CV, 2013)

Las múltiples utilidades que dispone esta herramienta la hacen ideal para desempeñar las operaciones de procesamiento de la imagen que manejará el sistema.

## **Processing 2**

Processing es un lenguaje de programación de medios interactivos basado en JAVA. Enfocado en una programación más simple y con medios artísticos, además esta herramienta maneja cada procedimiento a manera de Sketchs y pueden ser compilados tanto en aplicaciones independientes como en applets de JAVA. (Processing, 2013)

Tiene su propio IDE de un estilo minimalista y simple que permite al programador enfocarse al código. Adicionalmente posee librerías para adaptarse a otras IDEs como Eclipse o Netbeans.

Adicionalmente, contiene funcionalidades y características que permite su portabilidad, entre ellas la posibilidad de crear contenido online mediante el uso de Java applets, o sketchs usando su más reciente librería processingjs. También es posible realizar aplicaciones de escritorio para cualquier sistema operativo a través de JAVA Runtime.

En cuanto al contenido móvil processing permite el desarrollo para dispositivos Android y mediante el uso de librerías externas también en IOS.

Al ser orientado a medios artísticos processing presenta una gran variedad de funciones en el procesamiento de imágenes como:

- Escaneo de píxeles en imágenes
- Control del color de imagen incluyendo brillo, tono, saturación, etc
- Efectos de desenfoque, escaneo de contornos, acercamiento y alejamiento de imágenes.
- Puede adaptarse mediante librerías a Open CV permitiendo utilizar la interfaz amigable que ofrece processing con las funcionalidades avanzadas que proporciona Open CV.

Finalmente processing no solo representa una interfaz de desarrollo sino una gran comunidad que contribuye para el crecimiento, proporcionando gran cantidad

de documentación y soporte como foros o webs dedicadas a compartir información: OpenProcessing, SketchPatch, hascanvas o SketchPad.

### **Openframeworks**

Es un conjunto de herramientas de código C++ de licencia libre, cuyo principal objetivo es el desarrollo creativo de contenidos gráficos. Open framework se compone de las siguientes herramientas (openframeworks, 2014):

- **OpenGL, GLEW, GLUT, libtess2 y Cairo** para manejo de gráficos
- **rtAudio, PortAudio, OpenAL y Kiss FFT** para análisis entrada y salida de audio
- **FreeType** para fuentes
- **FreeImage** para almacenamiento y visualización de imágenes.
- **Quicktime, GStreamer y videoInput** para reproducción y manipulación de video
- **OpenCV** para visión artificial
- **Assimp** para carga de modelado 3D.

Esta herramienta en su más reciente versión presenta compatibilidad tanto para los sistemas operativos de escritorio: Windows, Linux y OS; como para los móviles: IOS y Android. Siendo compatible con XCode, Code Blocks, Visual Studio y Eclipse.

Al poseer características de Open CV presentará todas las funcionalidades de procesamiento gráfico de esta herramienta y que la librería imgproc pueda ofrecer.

Su punto negativo es que a diferencia de processing su manejo de memoria no es automático. Es decir requiere de punteros y sierras de memoria para no ralentizar el proceso.

### **Matlab**

Es una interfaz de desarrollo que contiene su propio código de programación denominado lenguaje M, este lenguaje se basa en cálculos numéricos sobre matrices.

Las posibilidades de Matlab son muy amplias ya que a más de su paquete principal, se pueden ampliar sus funcionalidades con paquetes adicionales llamados toolboxes o cajas de herramientas. Para el caso de manipulación de imágenes se emplea el toolbox “Image Processing”. (Matlab, 2014)

**Image Processing Toolbox** suministra un conjunto completo de algoritmos, funciones y aplicaciones para el procesamiento, análisis y visualización de imágenes. Puede realizar:

- Registro de imágenes
- Análisis de imágenes
- Segmentación de imágenes
- Mejora de imagen
- Reducción de ruido
- Transformaciones geométricas

Muchas de las funciones de este toolbox soportan procesadores multi-núcleo haciendo mucho más rápido y eficiente su desempeño. Además soporta varios tipos de imagen, entre ellos imágenes de altas resoluciones, gigapixel, perfil incrustado y tomográfico.

Además posee múltiples funciones de visualización y aplicaciones que permiten explorar imágenes y vídeos, examinar una región de píxeles, ajustar el color y el contraste, crear contornos o histogramas, y manipular las regiones de interés (ROI).

Entre las características más importantes se especifican: (Matlab Image Processing toolbox, 2014):

- **Mejora de imagen:** Filtros morfológicos (dilatación, erosión, etc.), afilado y enfoque, eliminación de ruido, ecualización de histograma, ajuste de gama, ajuste de contraste y de más funciones que permitan el mejoramiento de la imagen acentuando sus características o mejorando su relación señal-ruido.
- **Análisis de imagen:** Incluye funciones como búsqueda de formas, numeración de objetos, identificación de colores, o la medición de sus

propiedades, es decir, todas aquellas funciones que permitan identificar las características de una imagen.

- **Segmentación de imágenes:** Algoritmos que identifican y segmenta los límites de una región de una imagen entre ellos: detección de bordes, segmentación de divisiones, reconstrucción, transformaciones de distancia, etc.
- **Otras características:** Existen muchas más características que posee esta caja de herramientas como transformaciones geométricas, registro de patrones y aceleración de procesamiento en imágenes grandes, etc.

Matlab presenta un completo sistema de desarrollo sobre todo tipo de modificación de una imagen, con un único inconveniente que es una licencia de costo y cada toolbox representa un valor adicional.

### 3.2.3.2. Aplicación en el sistema

El procesamiento de imágenes se encargará de depurar la placa capturada en la detección del objeto. La imagen obtenida por el proceso de reconocimiento de placa necesitará ser mejorada y recortada para que el proceso de OCR brinde resultados satisfactorios.

Las técnicas de filtrado que se emplearán para este proceso serán la segmentación y binarización, esto con el fin de eliminar otros colores que distorsionen la placa para asemejarla más a un texto.

Se dispone de una amplia gama de herramientas que realizan procesamiento de imágenes, pero es necesario seleccionar aquellas que permitan adaptarse al sistema a desarrollar, a las herramientas de visión artificial y al OCR. Para ello deben cumplir ciertas características como:

- **Adaptable a cualquier interfaz de desarrollo:** Tanto Open CV como Tesseract son herramientas que se adaptan a cualquier interfaz gráfica, por este motivo se necesita que la herramienta de procesamiento de imagen permita acoplarse a un framework base para poder lograr una correcta

integración y así realizar las demás operaciones de programación. **Matlab** al tener su propia interfaz y diferente programación no dispone de la versatilidad necesaria para ser implementado en el sistema.

- **Lenguaje de programación unificado (JAVA):** Es necesario que el procesamiento de imagen trabaje sobre el lenguaje de programación en el que se desarrollará el sistema para su fácil manipulación. Debido a que el lenguaje a utilizar será JAVA **openframeworks** será descartado ya que trabaja sobre C++.
- **Complejidad mínima:** Para realizar un proceso tan simple y rápido es necesario que la herramienta tenga la mínima complejidad posible y su utilización no genere demasiados procesos para adaptarla.  
**Imgproc** al ser una librería incluida en OpenCV no presentará el menor inconveniente al acoplarse al sistema ya que forma parte del mismo y no requerirá de ningún proceso adicional en su instalación.
- **Funcionalidad específica:** Al realizar tan puntuales funciones en la imagen, no es necesario incluir otras herramientas más complejas, a pesar de que **processing** cumple con estas características, su principal funcionalidad se basa en operaciones que no son necesarias para el objetivo actual representando un gasto de recursos innecesario.

**Imgproc** de OpenCV es la herramienta seleccionada cumpliendo con los requerimientos antes mencionados y siendo la herramienta más viable para esta etapa de detección de placas.

### 3.2.3.3. Requerimientos

Tanto requerimientos de software como de hardware de Open CV fueron ya especificados en la sección de herramientas de visión artificial. Solo es necesario mencionar que para realizar el procesamiento de la imagen se requerirá que esta se encuentre guardada en un directorio temporal y así poder acceder y ejecutar los procesos de recorte y filtrado.

## **Conclusión**

La prioridad de este capítulo se enfocó en determinar las soluciones a los problemas de ingreso de vehículos. Estas soluciones dan a conocer la necesidad de automatizar el acceso mediante un sistema, que identifique y registre un determinado vehículo mediante técnicas que simulen la visión humana, pero que a su vez aceleren el proceso de comprobación y registro. Para ello se ha optado por la aplicación de la tecnología de visión artificial, que consistirá en aplicar el reconocimiento de la placa del vehículo en tiempo real e identificar su acceso. Para ello será necesario estudiar y seleccionar aquellas herramientas que van a ser utilizadas en el desarrollo de esta tecnología.

Uno de los utilitarios que destacan dentro de este ámbito es Open CV, una herramienta poderosa que contiene un amplio catálogo de librerías que permiten, desde el reconocimiento de patrones, hasta el análisis y procesamiento de imágenes, siendo la herramienta principal a utilizar en el sistema. Como paso final e importante, se realiza un proceso de OCR utilizando la herramienta más versátil y completa de este ámbito como lo es Tesseract. Estos procesos serán tratados a más detalle en el siguiente capítulo.

## Capítulo 4

### Análisis y diseño del sistema.

#### Introducción

Este capítulo presenta una visión global de las implementaciones ya existentes que emplean la tecnología de la visión artificial, también se define los pasos o etapas necesarias que se deberán cumplir para un proceso correcto de implementación del proyecto.

Adicionalmente se visualizará los distintos diagramas que ayudan a describir de mejor manera la estructura del proyecto.

#### 4.1. Recolección de información para el sistema

Es necesario realizar un estudio e investigación con la finalidad de determinar la información que se requerirá, y todas las características que permitirán que el sistema de ingreso de vehículos sea una solución viable a los problemas mencionados en el capítulo 2.

Se procederá a realizar un levantamiento de información de todo lo necesario, para que este sistema presente una correcta estructura y funcionamiento, consolidándose como una herramienta capaz de facilitar y solucionar los problemas presentes en el registro de acceso de vehículos.

##### 4.1.1. Investigación de características de un sistema de seguridad

Es necesario definir las bases que constituirán el sistema, por lo cual, se toma como punto de partida toda la información que se pueda obtener en cuanto a sistemas existentes en la ciudad, además de recursos y características que deberían ser necesarias para esta implementación.

##### 4.1.1.1. Estudios de sistemas existentes

Luego de realizar una observación de campo sobre los métodos o sistemas de ingreso vehicular comúnmente utilizados, se pueden destacar los siguientes:

- **Control de ingreso visual sin registro:** Este método usualmente es utilizado en parqueaderos informales y presenta un bajo número de vehículos, entre 4 a 15 autos aproximadamente (si el número de autos aumenta, el número de responsables también). Este método depende principalmente de un guardia o encargado, el mismo que mediante reconocimiento visual y verbal determina el acceso del vehículo al establecimiento, sin llevar ningún registro de su ingreso. En caso de duda, el encargado solicita documentos para constatar la identidad del conductor, creando dependencia de la capacidad de deducción del guardia, además de la ausencia de registros de acceso.
- **Control de ingreso escrito informal:** Este método de ingreso lleva un registro de manera poco ordenada, usualmente en papel, donde el encargado escribe la información del vehículo, y generalmente los campos usados son: placa, fecha, hora, y nombre del conductor en el mejor de los casos. Como negativa se destaca que el ingreso sigue dependiendo de la capacidad de reconocimiento del encargado, además que el registro es lento y tedioso.
- **Control de ingreso computarizado manual:** Se presenta con el uso de medios computarizados y guarda un registro de ingreso de una manera más segura, en algunos casos con una imagen del vehículo.

Este método consiste en un control realizado por el guardia de seguridad, que comprueba visualmente o mediante una identificación al conductor del vehículo, y busca o valida su información en una base de datos del computador; donde de acuerdo a la información suministrada niega el acceso o lo permite, dependiendo si los datos corresponden a los de la base.

El uso de este método se da en parqueaderos donde requieren una seguridad moderada y la cantidad de autos es considerable, además presenta una desventaja que implica un lapso de tiempo considerable para consultar la información y comprobarla, por lo que retrasa el acceso del automotor.

- **Control de acceso semiautomático mediante identificación:** Este mecanismo de ingreso es usado en establecimientos que manejan un número

elevado de vehículos y su característica principal es que interactúa directamente con el conductor. Para que el sistema funcione es necesario entregar una tarjeta de acceso a cada persona que usará el parqueadero. El proceso es simple y consiste en que el conductor presente su tarjeta en un detector de códigos de barras o un lector de chip infrarrojo, el sistema válida la información obtenida y permite el acceso. Este sistema guarda un registro de ingreso completo, además de una captura de imagen del vehículo, pero su desventaja es que debido al número elevado de vehículos presenta una dependencia con el usuario y éste puede atrasar el ingreso o interrumpirlo por cualquier motivo.

- **Sistema automatizado mediante visión artificial:** Este sistema se encuentra ya implementado en algunos países; tiene la destreza de controlar grandes cantidades de vehículos incluso puede ser utilizado con fines policíacos. Su principal característica reside en que no necesita interacción alguna con ningún usuario. El sistema automáticamente toma la placa vehicular y digitaliza sus caracteres para luego comprobar en una base de datos la información del vehículo y así guardar un registro con la información reconocida, permitiendo un ingreso de vehículos seguro y de mayor velocidad que los métodos anteriores.

#### **4.1.1.2. Investigación de características a implementar**

Luego de realizar un estudio sobre los sistemas que existen para el ingreso vehicular, se ha reunido cada una de sus características con la finalidad de construir un sistema de ingreso de vehículos sólido y seguro.

Este tema se complementará a mayor profundidad en el capítulo siguiente. En esta sección solo se explicará en rasgos puntuales.

- **Registro de ingreso confiable.-** Como cualquier sistema, este requiere que se registre la información de ingreso de cualquier vehículo con su respectiva fecha y hora, además de imágenes que confirmen visualmente el acceso vehicular. El sistema debe presentar una estructura de datos robusta con todos los campos necesarios, adicionalmente deberá poseer almacenes de datos para

recopilar las imágenes y así posteriormente poder generar consultas de actividades. Se debe tener en cuenta que una base de datos sólida y bien estructurada permitirá una más rápida y eficiente transaccionalidad.

- **Integración con la base de datos del establecimiento.-** En cuanto al proceso de comprobación, el sistema podrá consultar información del usuario y no solo validar el ingreso, sino también dar a conocer la información que posee la institución del vehículo y su conductor.
- **Información de entrada bien definida.-** Ya que el sistema de reconocimiento de visión artificial presenta un tratamiento delicado de la información, es necesario establecer claramente las fuentes de donde se obtendrá la misma, en donde se almacenará y como será capturada (ubicación de la cámara, resolución, etc.).
- **Comprobación confiable.-** La visión y el criterio humano es un factor muy difícil de superar, por su adaptabilidad a entornos o situaciones cambiantes; por lo que se debe afinar el método de comprobación y digitalización, aplicando procesos de depuración óptimos para lograr obtener la información verídica y permitir una comprobación exitosa.
- **Preparación ante problemas.-** El sistema deberá estar preparado ante posibles inconvenientes que puedan llegar a presentarse, es decir, alertar sobre inconvenientes en el ingreso y presentar soluciones alternativas que se deberían tomar. Para lo cual se debe aplicar soluciones alternas como: dar acceso manual para aquellos casos excepcionales, informar el número de accesos de un automotor, permitir accesos eventuales, etc.
- **Sistema independiente.-** El sistema podrá implementarse con una base de datos de la institución, pero es importante que este también pueda trabajar de manera independiente sin requerimientos costosos ni excesivos y solo con un computador y una cámara. Además deberá tener su propio administrador para el ingreso de vehículos, consultas, etc.

#### 4.1.2. Obtención de elementos requeridos de cada herramienta definida

Las imágenes representan el elemento básico para el sistema de visión artificial, tanto para la etapa de reconocimiento como para la etapa de digitalización. A continuación se detallará para cada etapa la obtención de imágenes y su estructura, definiéndolas como elementos de entrada para los distintos procesos del sistema.

##### 4.1.2.1. Obtención de imágenes

Consiste en adquirir las muestras para el proceso de validación, este paso será muy diferente para cada etapa:

##### **Etapa de entrenamiento:**

En esta etapa será necesario capturar manualmente las imágenes mediante una cámara fotográfica con buena resolución y así poder definir de mejor manera la placa, el patrón deberá ser claro y enfocado en todo su contorno sin ningún elemento exterior. Se procurará que las placas estén limpias, bien remarcadas y con sus letras claramente reconocibles. En cuanto a la cantidad de imágenes es recomendable conseguir un número elevado de muestras, para mejorar los procesos de entrenamiento y obtener mayores características de la placa.

En el caso de esta investigación se han tomado alrededor de 544 imágenes de placas guardadas en formato .jpg con una resolución promedio de 900x500px, estas fueron editadas con la finalidad de aclararlas y recortarlas para obtener los mejores resultados en el entrenamiento.



Figura 4.16 Placa tomada para entrenamiento

Nuevamente estas imágenes fueron recortadas y transformadas para ser utilizadas en el entrenamiento OCR (Reconocimiento Óptico de Caracteres). El proceso de depuración para OCR fue diferente ya que se tomó como entrada la

imagen limpia y se procedió a cortar cada carácter de cada placa. Se explicará el proceso a continuación:

- Se recorta cada carácter de la placa



Figura 4.17 Imágenes tomadas de los caracteres.

- Se transforma en modo binario para detectar solo 2 colores representados 0 en blanco y 1 en negro.
- Se coloca cada carácter en un documento final en orden alfabético, en formato tif.



Figura 4.18 Caracteres clasificados y binarizados.

Se deberán recortar todas las imágenes y repetir el mayor número de veces las letras más comunes y más claras para que el proceso de detección sea robusto.

#### **Etapas de detección:**

En esta etapa las imágenes serán tomadas automáticamente en el ingreso del vehículo mediante una cámara web. Las imágenes se guardarán en formato png y serán registradas con su proceso de validación correspondiente (en caso de que no pueda ser validada se guardará con el registro identificado como acceso denegado). El proceso completo, incluyendo el procedimiento de OCR, será el siguiente:

- Una vez que la cámara enfoca el auto se detectará la placa, si esta existe dentro de la muestra tomada, se guardará una captura de ella recortando todo el sector de la placa.



Figura 4.19 Imagen capturada al detectar placa.



Figura 4.20 Imagen recortada.

- La imagen pasará por un proceso de depuración y transformación para ser leída en la etapa de detección de caracteres (OCR).

Finalmente, cabe mencionar que las imágenes para el entrenamiento fueron tomadas de distintos parqueaderos, tanto de empresas privadas, como de venta de automotores, notificando a cada uno de los encargados sobre el uso que se le dará a las capturas; con el compromiso de que no serán usadas con ningún otro fin que no sea el académico para el entrenamiento del sistema.

#### **4.1.2.2. Definición de la estructura de las imágenes**

La estructura de las imágenes tienen mayor importancia en el proceso de entrenamiento, ya que este método requiere que las imágenes se clasifiquen y se

ordenen para que el sistema pueda reconocer el objeto y excluir toda la información que no forme parte de él.

#### 4.1.2.2.1. Clasificación

Las imágenes se dividirán en 2 grupos, para el entrenamiento de la placa y para el entrenamiento OCR.

#### Entrenamiento de la placa

Para esta sección se distinguirán 2 tipos de imágenes:

**Imágenes positivas:** Son todas aquellas imágenes que contienen el objeto a ser reconocido. Estas imágenes deberán ser en formato JPG y deben contener el objeto bien definido y en un tamaño considerable para que sea fácil de detectar. En este caso particular el objeto será la placa.



Figura 4.21 Imagen positiva.

**Imágenes negativas:** Son aquellas imágenes que no contienen el objeto pero deberán tener relación al ambiente en el que se va a desarrollar el mismo para poder diferenciarlo. En este caso serán texturas e imágenes de calles, veredas, ambientes urbanos, tierra, césped, etc.



Figura 4.22 Imagen negativa.

## Entrenamiento OCR

**Imágenes de caracteres:** Son todas aquellas imágenes que contienen cada carácter que será entrenado para la creación de un nuevo archivo de lenguaje. Estas imágenes deben formar parte de uno o varios archivos en formato TIF (Tagged Image File Format) para su entrenamiento.

### 4.1.2.2.2. Numeración

Las imágenes deben tener un estándar y un orden establecido de tal manera que el entrenador pueda clasificarlas y almacenarlas fácilmente para el aprendizaje. Con lo anteriormente mencionado se debe proceder de la siguiente manera:

- Para las **positivas** en el formato p###.jpg
- Para las **negativas** n###.jpg

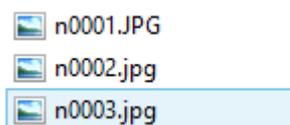


Figura 4.23 Ejemplo de numeración.

### 4.1.2.2.3. Posición interna

En el entrenamiento es necesario especificar, con cada imagen positiva la ubicación inicial exacta del objeto, su alto y su ancho, igualmente se pueden incluir

parámetros adicionales como el ángulo de rotación o las veces que se repite el objeto en la imagen. La posición será especificada en coordenadas (x, y); tanto alto como el ancho se manejarán en píxeles.



Figura 4.24 Ejemplo de coordenadas 604 522 359 152.

#### 4.1.2.2.4. Ubicación

Corresponde al último paso a realizarse en donde se debe especificar un archivo en formato txt con la ubicación, tanto para las imágenes positivas como para las negativas, además se tendrá que incluir los datos de posición interna explicados en el tema anterior. El formato establecido será (ubicación posición\_x posición\_y ancho alto) para las positivas y (ubicación) para las negativas.

Archivo positivas.txt

```
positivas/p001.jpg 1 140 100 45 45
positivas/p002.jpg 2 100 200 50 50
```

Archivo negativas.txt

```
negativas/n001.jpg
Negativas/n002.jpg
```

En el entrenamiento OCR se establecerá un archivo box donde se verán especificadas las coordenadas de cada carácter y a que letra corresponden.

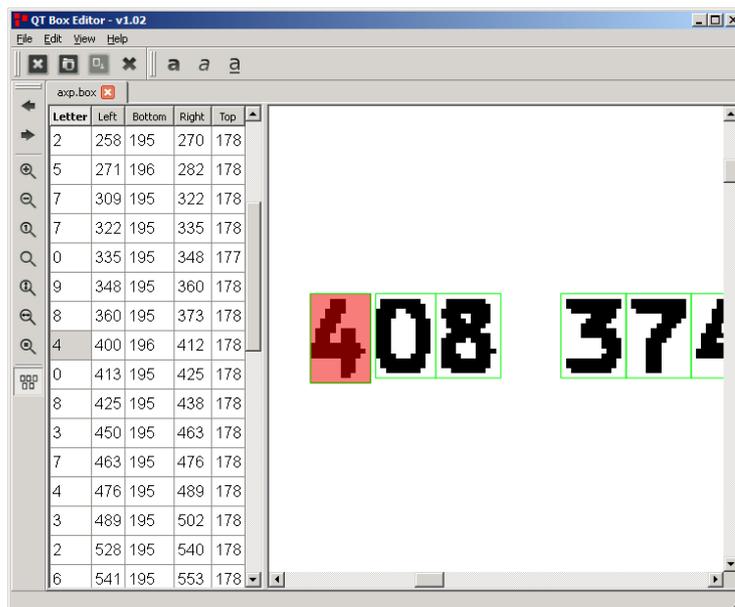


Figura 4.25 Ejemplo de archivo box con QT Box Editor.

## 4.2. Determinación de los requerimientos básicos de hardware

Al tener bien definidos y estructurados los elementos de entrada es necesario especificar los equipos de hardware que deberán estar listos para los procesos de entrenamiento, programación y pruebas de detección.

### 4.2.1. Equipos necesarios para desarrollo y pruebas

A continuación se especificarán los requerimientos mínimos, necesarios tanto en el proceso de entrenamiento como en el proceso de detección.

#### Proceso de entrenamiento

En el proceso de entrenamiento es necesario un equipo con requerimientos altos ya que este necesita un mayor nivel de procesamiento para el aprendizaje de objetos y caracteres. Es aconsejable cumplir con los requerimientos recomendados ya que las características del computador influenciarán directamente en el tiempo de aprendizaje, es decir a mayor procesador menor tiempo de espera.

Computador utilizado:

- CPU: Core i7 3.4Ghz

- RAM: 16Gb
- Disco Duro: 2TB

### **Proceso de programación**

En el proceso de programación no es necesario un computador potente, pero si es recomendado que tenga características moderadas, y así lograr un rendimiento con mayor fluidez al momento de procesar toda la información.

Requerimientos mínimos:

- CPU: Core 2 Duo 2.2Ghz
- RAM: 2Gb
- Disco Duro: 20gb

Requerimientos recomendados:

- CPU: Core Quad o I3 de 2.8Ghz
- RAM: 4Gb
- Disco Duro: 50gb

Computador utilizado:

- CPU: Core i7 3.4Ghz
- RAM: 16Gb
- Disco Duro: 2TB

### **Proceso para uso del sistema**

En el proceso para el uso del sistema, no es necesario un equipo potente ya que el sistema final no emplea grandes recursos, pero es mejor tener los requerimientos recomendados y así evitar cualquier tipo de inconvenientes, minimizando el tiempo lo menor posible, tanto en consultas como en procesamiento.

Requerimientos mínimos:

- CPU: Pentium 4 1.8Ghz
- RAM: 1Gb
- Disco Duro: 20gb

Requerimientos recomendados:

- CPU: Core 2 Duo 2.4 Ghz
- RAM: 2Gb
- Disco Duro: 50gb

Computador utilizado:

- CPU: Core i3 2.4Ghz
- RAM: 4Gb
- Disco Duro: 1TB

En el caso de estudio, se ha desarrollado todos los procesos en dos computadores de diferentes características para evitar cualquier inconveniente, además de desarrollar el proceso de pruebas con mayor rapidez y poder realizar otros procesos de forma paralela, de esta forma ahorrando tiempo de desarrollo.

### **4.3. Identificación de los procesos que intervienen en el sistema**

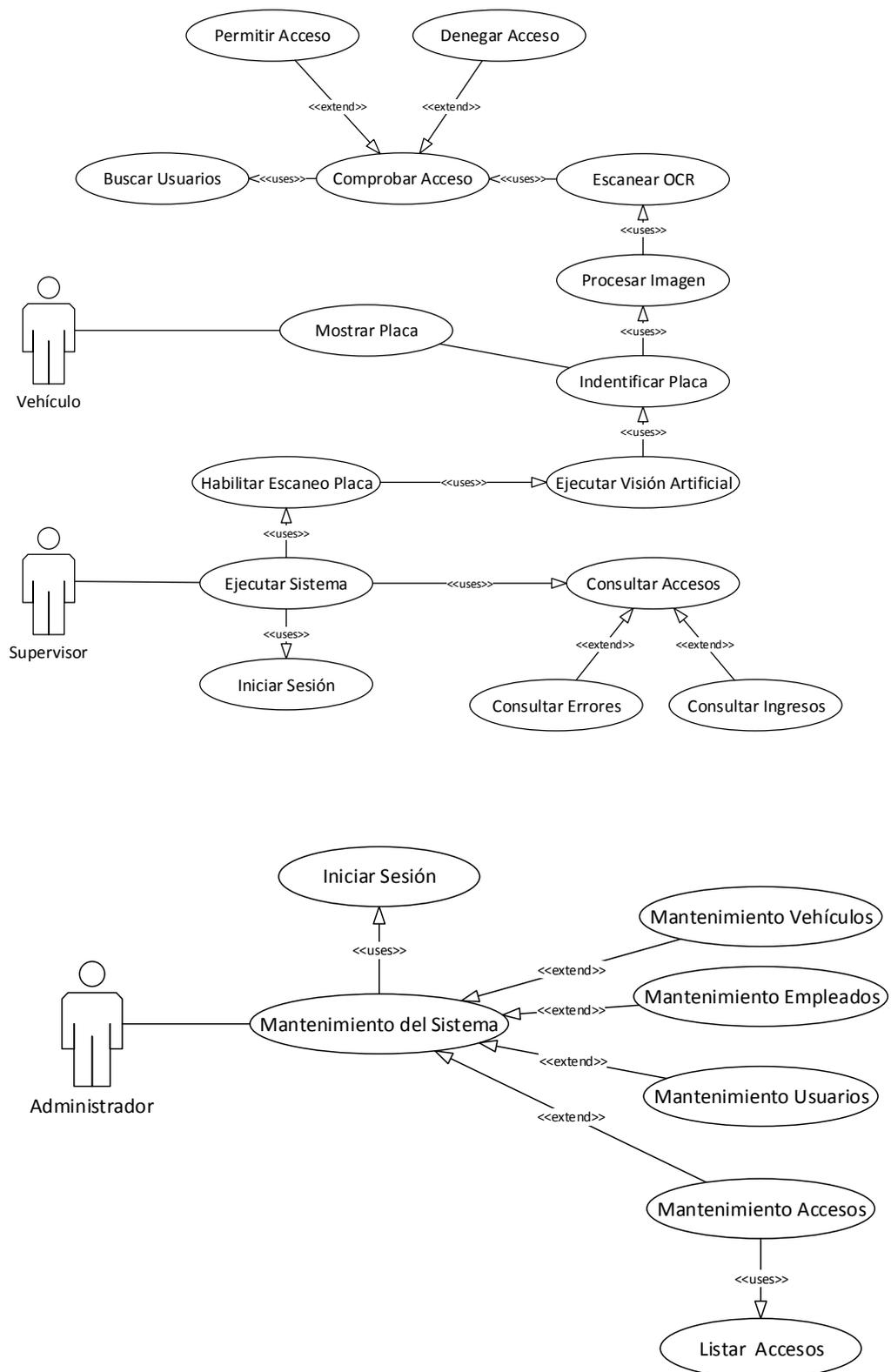
Los procesos se categorizarán en tres etapas importantes, que son:

- Reconocimiento del patrón
- Reconocimiento óptico de caracteres
- Aplicativo que una los resultados de los primeros ejercicios y que compruebe los caracteres del segundo ejercicio con los registrados en la base de datos.

A continuación se especificarán los distintos diagramas propuestos al igual que las características y funcionalidades de cada uno de los procesos que intervienen en el desarrollo del sistema, considerando desde la obtención de información, el procesamiento de los datos y los resultados de salida.

#### **4.3.1. Diagrama de casos de uso**

En este diagrama se indicarán dos procesos por separado. El proceso de detección del patrón y el proceso de administración:



Como se observa en el diagrama, intervienen tres actores: el usuario encargado de ejecutar el sistema, el vehículo que desea tener acceso y el administrador del sistema.

**Supervisor:** Será el encargado de llevar un proceso de “supervisión del sistema”, ya que podrá tomar decisiones no controladas o accesos manuales si el caso lo amerita. Sus funciones implican la ejecución del sistema para el inicio del proceso de detección automática, dar acceso manual a un determinado vehículo o realizar consultas sobre los ingresos en un determinado tiempo.

Este usuario controlará todos aquellos accesos que el sistema no pueda identificar por múltiples factores o inconvenientes de deducción, como pueden ser: la no existencia de una placa en un automotor nuevo, el mal estado de una placa que dificulta el reconocimiento, etc.

**Vehículo:** Este no necesita de algún tipo de acción directa sobre el sistema solamente deberá acercarse a la cámara para ser escaneado, es recomendable que se mantenga un buen estado de la placa para que la misma pueda ser reconocida con mayor precisión.

**Administrador:** Es el usuario con privilegios de creación y modificación de registros en el sistema. El Administrador podrá registrar vehículos, crear empleados, crear usuarios, consultar accesos, etc.

Debido a que la funcionalidad del sistema es automática ningún otro factor intervendrá en los procesos de detección del mismo.

#### **4.3.2. Diagrama de actividades**

Si se desea un correcto funcionamiento del sistema, se deberá especificar aquellas actividades que realizará cada uno de los procesos de detección, depuración y validación, además se necesitará comprobar cómo estos se integran para obtener el resultado deseado. Este diagrama permitirá conocer como fluirá la información en cada caso, definirá las diferentes decisiones que tomará el sistema, analizará el nivel de participación de cada actor y verificará los procedimientos de cada nivel de ejecución.

Los actores que intervendrán en el sistema son:

- **Vehículo:** Corresponde al automotor que será escaneado por la cámara cuya intervención será pasiva, ya que este no realiza ninguna acción directa en el sistema.
- **Usuario (Supervisor):** Será un respaldo al sistema para aquellos casos excepcionales; ya que podrá permitir ingreso manual o registrar nuevos vehículos como solución a posibles problemas que puedan presentarse, tanto de lectura de placas o en accesos informales, etc.
- **Administrador:** Será el encargado del mantenimiento del sistema tanto para el ingreso de los vehículos como las consultas, registros, etc.

El proceso ya explicado con anterioridad se basará principalmente en mantener la cámara en constante monitoreo hasta **detectar un objeto placa**, este proceso será realizado por Open CV tras un previo entrenamiento. Una vez detectado el objeto, se **recortará** la imagen y se **almacenará** por seguridad. El sistema alistaré la imagen para ser procesada por Tesseract, por lo que la imagen será **transformada en binario** y la procesará en formato png (**codificación de imagen**) para obtener el mejor resultado del OCR.

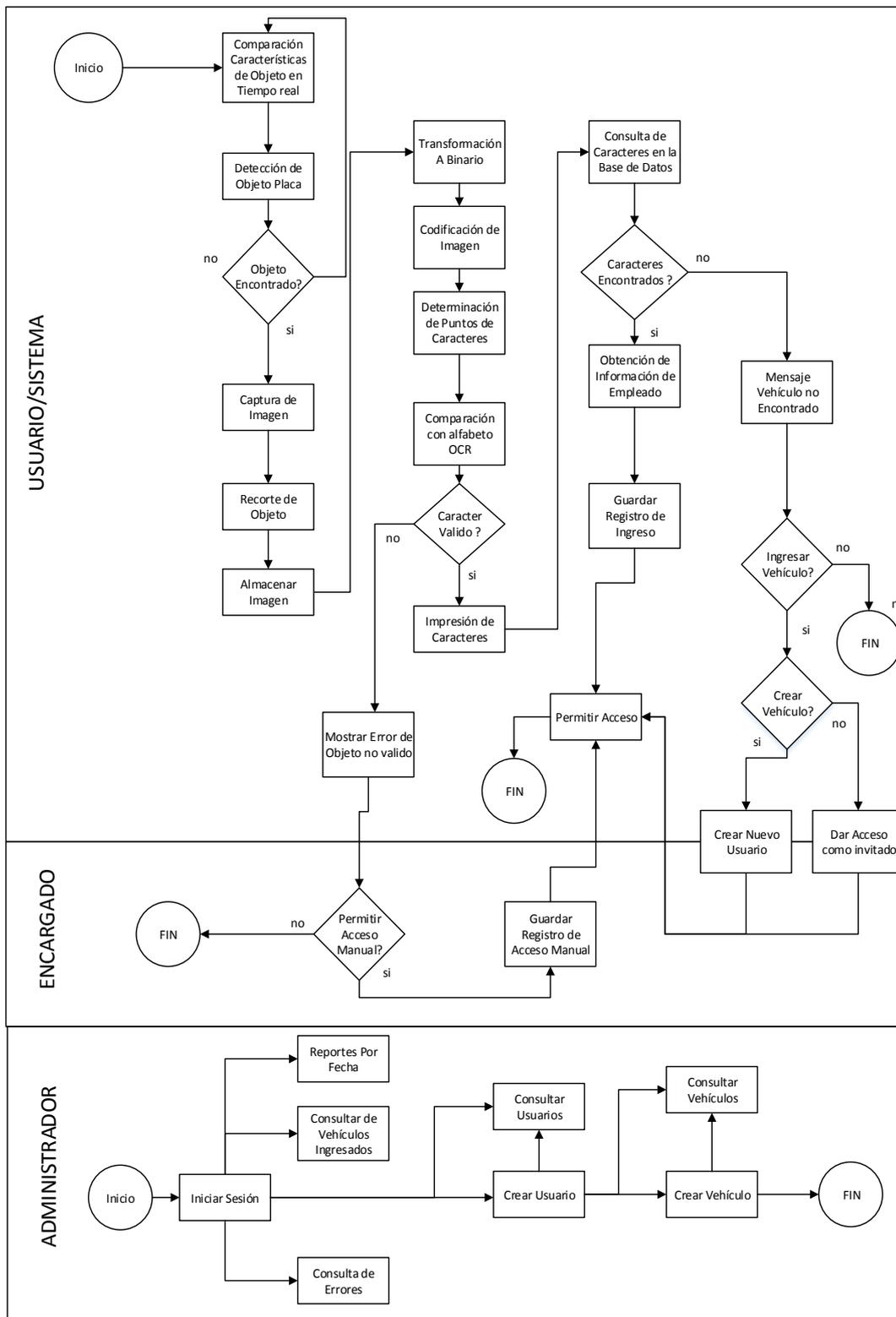
Los procesos de **determinación de caracteres y comparación con alfabeto OCR** son procesos internos de Tesseract que determinarán y almacenarán carácter por carácter las letras de la placa. El sistema al recibir la respuesta de Tesseract verificará si los caracteres pueden ser leídos, caso contrario el resultado denegará el acceso.

Finalmente si se generó la información correcta, el sistema buscará los datos del propietario y los almacenará en un registro de ingreso para dar por completado el proceso, permitiendo el acceso del vehículo. Si el empleado no se encuentra registrado, el supervisor podrá ingresar el vehículo como un nuevo empleado o como usuario invitado.

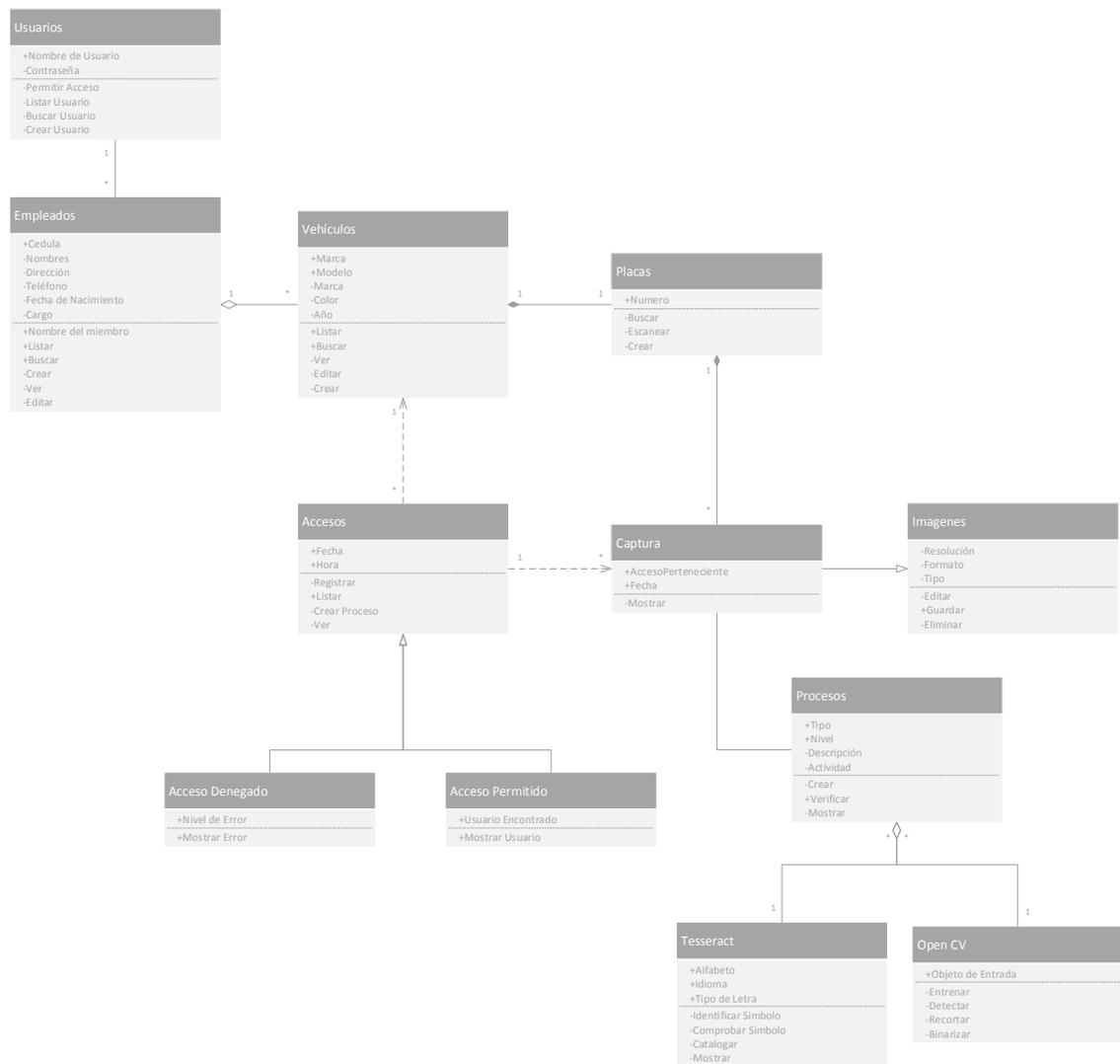
Existirá además un proceso de administración que mediante **el inicio de sesión** de un usuario administrador, permitirá dar mantenimientos a los registros del sistema como puede ser, **ingresar vehículos, empleados, y consultar información.**

Como se observa en el diagrama el 80% del proceso del sistema se realiza sin ningún tipo de intervención ya que el sistema toma decisiones de acuerdo a como se va desarrollando y procesando en cada paso la información. La única intervención de manera “secundaria” es en los accesos manuales ocasionados por algún error en la detección (placa sucia o borrosa), en el proceso de lectura de caracteres o por ingreso temporal (a algún vehículo no registrado) y en la de búsqueda de la placa. Los procesos dependen de un sólido entrenamiento de las herramientas de Open CV y Tesseract, debido a que son procesos fundamentales en el flujo de información y deben ser afinados en su entrenamiento para evitar problemas.

Finalmente, será verificado si se encuentran registrados los datos, para mantener informado al administrador de los errores ocurridos y los accesos automáticos y manuales. Es necesario mencionar que los procesos se realizan en milisegundos para emplear un procesamiento instantáneo.



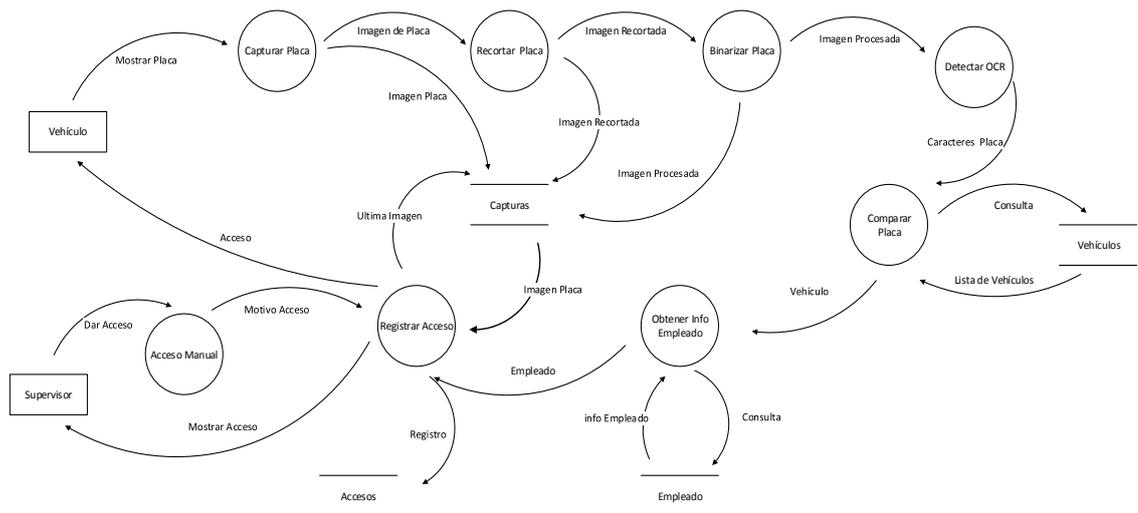
### 4.3.3. Diagrama de clases



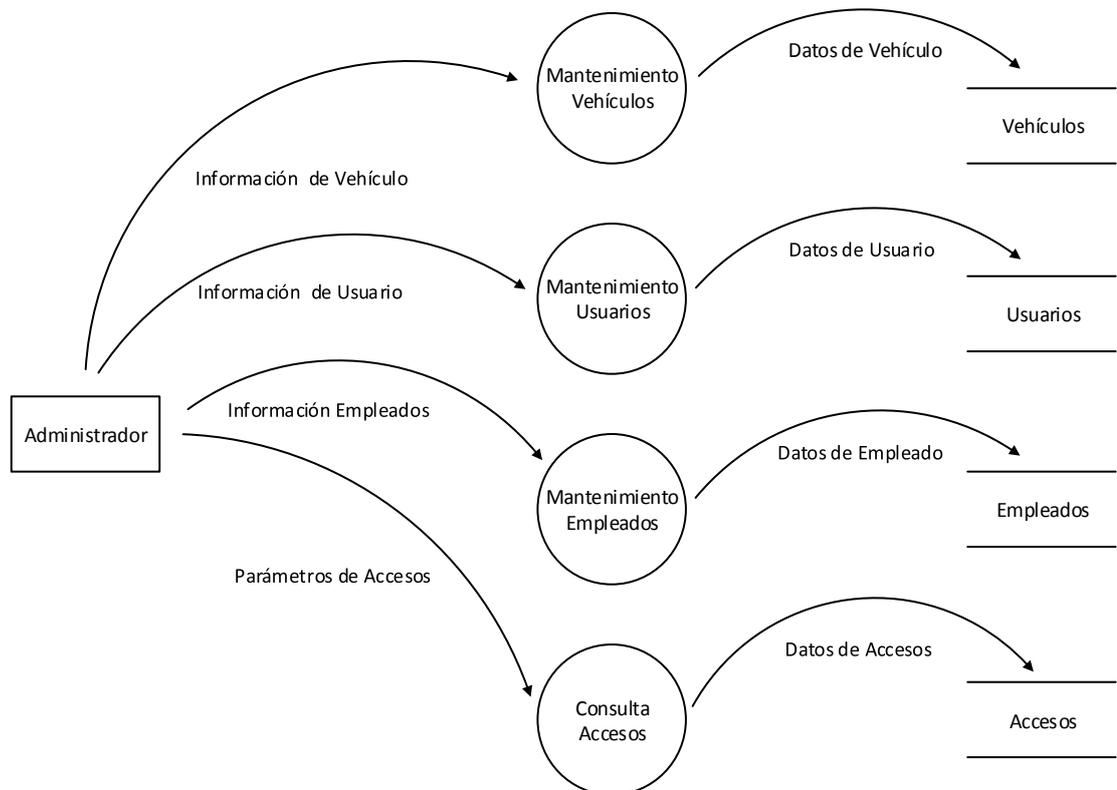
Las clases Open CV y Tesseract muestran los atributos y operaciones que realizan cada librería y que son necesarios para el desarrollo del sistema, estas poseen muchas más características que no serán utilizadas por lo que se encuentran excluidas del diagrama.

### 4.4. Determinación de los flujos de datos

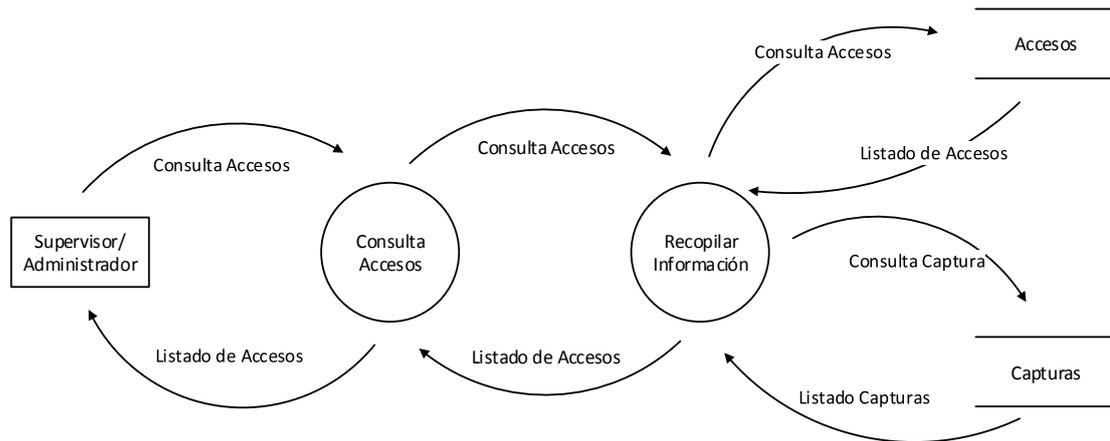
Luego de analizar los procesos y funciones del sistema de reconocimiento de placas, es necesario identificar aquellos que intervendrán en cada estado del sistema para su almacenamiento y consulta.



Se puede visualizar en el flujo de datos del proceso de detección de placas, que por cada acceso valido o denegado se almacena su respectiva imagen, esto es debido a que es necesario controlar y registrar los accesos que se presentan con el sistema.



En la administración del sistema no existe ningún tipo de inconveniente ya que por cada petición se consultará en su respectiva tabla.



En el proceso de consulta o registros de accesos, cabe denotar, que el momento de realizar dicha consulta también se obtendrá la imagen de captura de cada ingreso o error.

#### 4.5. Determinación de la información para los almacenes de datos

El sistema no presenta una estructura compleja de almacenamiento, ya que los datos de las imágenes y los registros serán grabados en una sola ubicación que estará en el computador que aloje el sistema. La información se enfocará en los tipos de datos que serán almacenados, además de los procedimientos de los que provienen:

- **Registros:** Serán conjuntos de datos livianos que se almacenarán en la base de datos del sistema alojados en un servidor o en un computador, no presentan tamaños considerables individualmente, pero debido a que los registros se guardarán en grandes cantidades en el sistema presentarán un flujo de información alto hacia el sistema.
- **De usuarios:** Los usuarios o empleados serán almacenados mediante el administrador y la información se mantendrá en constante consulta por parte del sistema para determinar los propietarios de los vehículos.
- **De vehículos:** Igualmente que los usuarios, los registros de vehículos se mantendrán en constante consulta por el sistema, pero estos estarán en mayor cantidad ya que un usuario puede tener varios vehículos.

Proviene del administrador específicamente del proceso de ingreso manual de cada vehículo.

- **De ingreso:** Son los registros diariamente almacenados por el sistema de manera automática, aquí se representará la mayor transaccionalidad con la base de datos ya que realizará procesos de guardado casi simultáneamente. Estos registros se guardarán cada vez que el vehículo ingrese al parqueadero, identificándolos de dos formas; como registros de ingresos correctos y registros de ingresos denegados.
- **Imágenes:** Son los conjuntos de datos más pesados que causarán un aumento de almacenamiento considerable. Estas imágenes se dividen en tres categorías que son:
  - **Captura inicial:** Esta imagen tratada internamente por el sistema, es obtenida al iniciar el proceso de detección luego de que el sistema haya encontrado el objeto placa. Esta imagen será guardada junto con el registro correspondiente.
  - **Imagen recortada:** Esta imagen también es tratada internamente por el sistema y es obtenida mediante el proceso de delimitación o corte del objeto. Esta imagen no se almacenará en el computador.
  - **Imagen binarizada:** La imagen después de ser recortada pasa por un proceso de binarización, obteniendo una imagen de mayor calidad en formato .png. Luego de pasar por el proceso de comparación de Tesseract será almacenada con su registro de acceso correspondiente.

#### 4.6. Diseño de la base de datos

En la estructura de la base de datos, se ha planteado un manejo simple y objetivo, ya que es necesario evitar consultas largas y complejas cuando se obtiene información, y así lograr que el sistema se desarrolle en el menor tiempo posible.

Al construir la base de datos se debe enfocar en los aspectos más importantes a tomar en cuenta para el sistema: la información del empleado, del vehículo y los registros de ingreso.

**Empleados.-** Esta tabla mantiene los datos básicos del usuario, los mismos que serán asociados con los vehículos correspondientes. Entre sus campos a más de la información básica, se tiene el estado que indicará si el empleado se encuentra habilitado o no.

**Usuarios.-** Es necesario tener una tabla de usuarios para establecer cada acceso o rol en el sistema, ya sea como administrador o encargado.

**Cargos.-** Esta tabla solamente se establecerá por motivos de pruebas e indicará los diferentes cargos de los empleados del sistema de acceso vehicular.

**Vehículos.-** Es una de las tablas más importantes del sistema, cuyos campos contendrán toda la información que pueda describir al vehículo registrado (color, marca, modelo, etc.), además su identificador principal será la placa del vehículo. Esta tabla será la que con mayor frecuencia recorra el sistema en búsqueda de la placa detectada y se enlazará al empleado al que pertenece.

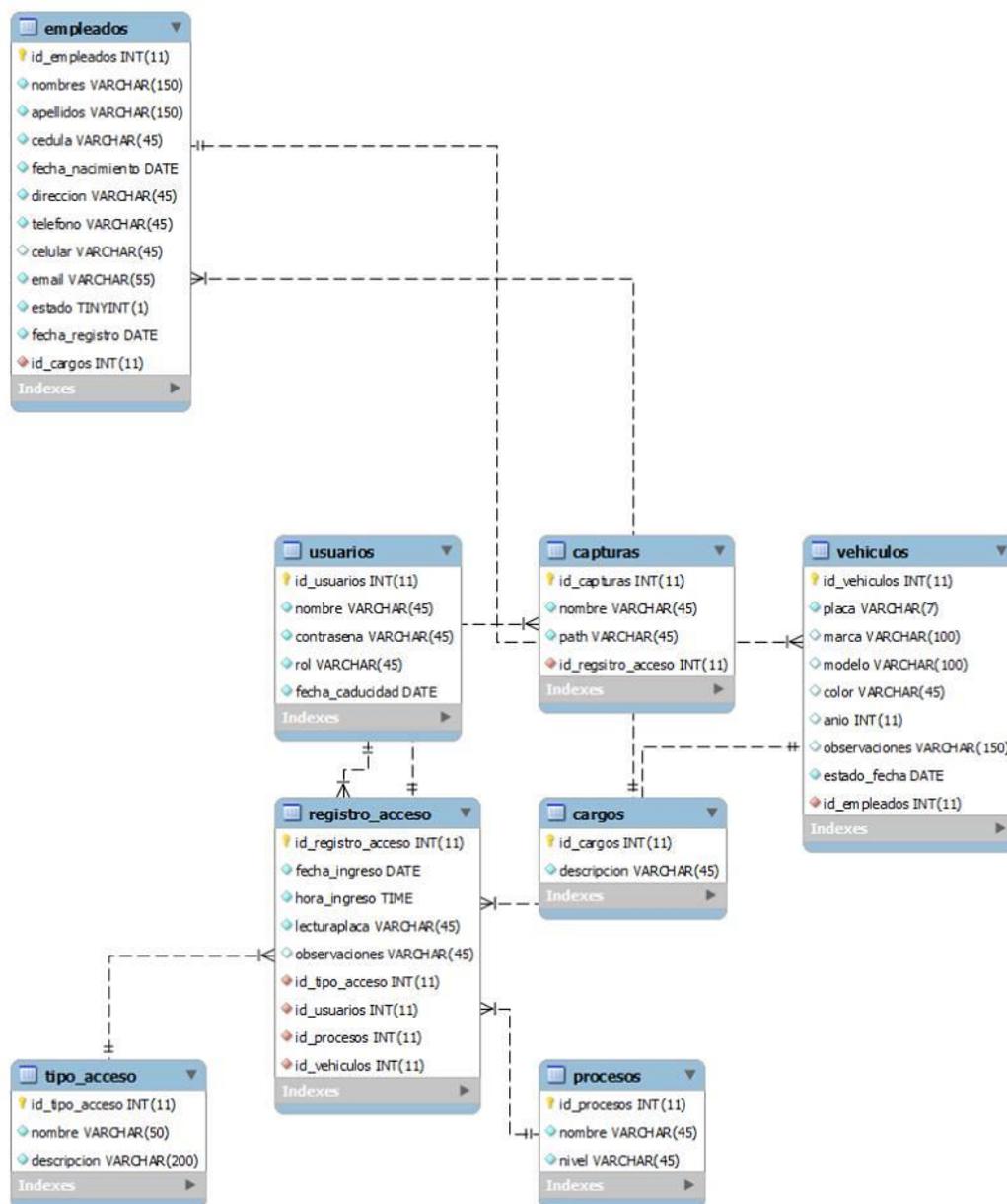
**Registro de acceso.-** Será la tabla que guarde todos los registros de intento de acceso de los vehículos al parqueadero, es decir, manejará los tipos: ingreso automático, ingreso manual y acceso denegado. Se relaciona directamente con una tabla de tipos de acceso y también con una tabla de capturas. Cada registro incluirá su fecha, hora, un tipo de acceso, la placa leída durante el proceso y el vehículo encontrado (en el caso de ingreso).

**Tipos de Acceso.-** Contendrá los tipos de acceso anteriormente mencionados dentro de la tabla **Registro\_acceso**. Se indicará si el acceso fue automático, manual o denegado y así identificar el estado del registro.

**Capturas.-** Es la tabla que guardará un registro de las imágenes ligadas a cada acceso.

**Procesos.-** Son los diferentes niveles o etapas que contendrá el sistema. Cada proceso estará ligado a un registro y así determinar hasta qué nivel llega la detección.

Las tablas empleados, cargos, y vehículos tendrán su propio mantenimiento y podrán ser modificadas por el usuario administrador del sistema, mientras que las tablas registro\_acceso, y capturas solo podrán ser consultadas ya que son guardadas automáticamente por el sistema.



#### 4.7. Definición de procedimientos en cada herramienta

Cada herramienta tiene su objetivo dentro del proyecto que se desarrolla, cuyos procedimientos tienen etapas que deben cumplirse para lograr el objetivo deseado.

#### 4.7.1. Elementos de entrada

Cada proceso en el que se divide el sistema tiene elementos de entrada, que deberán cumplir ciertas condiciones para poder obtener los elementos de salida deseados. Debido a que estos procesos están interconectados, cada resultado influirá directamente como elementos de entrada para el siguiente proceso y notablemente en el resultado final.

Anteriormente se ha definido los tipos de imágenes deseadas para cada fase de desarrollo, en esta sección serán catalogados para cada procedimiento del sistema a un nivel más técnico y directo.

#### Open CV

Nivel de proceso	Nivel 0 o Entrenamiento
<b>Tipo de elemento</b>	Imagen, Archivo TXT
<b>Formato de elemento</b>	JPG, coordenadas y tamaño
<b>Requisitos</b>	Lista de imágenes positivas Lista de imágenes negativas Adjuntar archivo de coordenadas
<b>Procedimiento</b>	Entrenamiento de objeto
<b>Elemento de salida</b>	Archivo haartraining XML
<b>Observaciones</b>	Requiere entre 500 y 1000 imágenes positivas El número de imágenes negativas deben presentar entre 130-150% de las positivas.

Tabla 4.1 Elementos de entrada, para entrenamiento Open CV

Nivel de proceso	Nivel 1 o Reconocimiento
<b>Tipo de elemento</b>	Objeto
<b>Formato de elemento</b>	Placa física del vehículo
<b>Requisitos</b>	Buena iluminación Angulo de cámara aceptable Rango de visión aceptable
<b>Procedimiento</b>	Detección y captura de placa
<b>Elemento de salida</b>	Captura JPG
<b>Observaciones</b>	Placas ilegibles serán excluidas

Tabla 4.2 Elementos de entrada, para reconocimiento Open CV

Nivel de proceso	Nivel 2 o Procesamiento
<b>Tipo de elemento</b>	Imagen
<b>Formato de elemento</b>	JPG
<b>Requisitos</b>	Debe contener la placa Placa bien diferenciada del ambiente.
<b>Procedimiento</b>	Corte y binarización de la placa
<b>Elemento de salida</b>	Imagen binaria PNG
<b>Observaciones</b>	Placas ilegibles serán excluidas. Se debe obtener el mejor resultado

Tabla 4.3 Elementos de entrada, para procesamiento Open CV

### Tesseract (OCR)

Nivel de proceso	Nivel 0 o Entrenamiento
<b>Tipo de elemento</b>	Imagen
<b>Formato de elemento</b>	TIF
<b>Requisitos</b>	Caracteres en el alfabeto a entrenar Imágenes claras

	Mínimo 5 repeticiones por letra
<b>Procedimiento</b>	Entrenamiento de letras placa
<b>Elemento de salida</b>	Alfabeto OCR
<b>Observaciones</b>	Puede tener uno o varios archivos TIF. Se recomienda combinaciones de palabras y letras sueltas

Tabla 4.4 Elementos de entrada, para entrenamiento Tesseract

Nivel de proceso	Nivel 3 o OCR
<b>Tipo de elemento</b>	Imagen binarizada
<b>Formato de elemento</b>	TIF
<b>Requisitos</b>	Caracteres legibles Formato correcto
<b>Procedimiento</b>	Identificación y digitalización de caracteres.
<b>Elemento de salida</b>	Cadena de caracteres de la placa
<b>Observaciones</b>	Si no es reconocido devolverá un error

Tabla 4.5 Elementos de entrada, para reconocimiento OCR

Nivel de proceso	Nivel 4 o Comparación
<b>Tipo de elemento</b>	Placa digitalizada
<b>Formato de elemento</b>	Cadena de texto
<b>Requisitos</b>	Textos en el rango [A-Z] – [1-9]
<b>Procedimiento</b>	Comparación
<b>Elemento de salida</b>	Resultado de la comparación
<b>Observaciones</b>	Al ser correcto se procede a consultar la información.

Tabla 4.6 Elementos de entrada, para comparación de resultados

## **4.7.2. Procedimientos a realizar**

Para una mayor comprensión, se definirá a mayor detalle cada uno de los procedimientos que identificarán las etapas del proyecto. Los procedimientos a realizarse para cada uno de los ambientes que se manejan dentro del sistema, estarán correctamente delimitados y tendrán una identificación única para así usarlos de guía para la posterior etapa de desarrollo.

### **4.7.2.1. Enumeración de pasos a seguir**

El objetivo principal es definir y delimitar las diferentes fases que tiene el sistema de detección, ya que se deberá trabajar en cada módulo afinando cada resultado de la mejor manera posible.

Se ha optado por nombrar a cada etapa con un nivel, para así identificar los pasos de cada proceso que llevarán al resultado final en la detección:

#### **Nivel 0 o de entrenamiento**

El nivel es nombrado con el numeral cero debido a que este es una fase previa al proceso de detección. Este proceso es realizado con fin de obtener un archivo de entrenamiento para el sistema, una vez obtenido, no se vuelve a realizar. Aun así el entrenamiento es la base para el proceso de detección y para que se pueda identificar el objeto correctamente.

Este nivel se divide en dos fases: para Open Cv y para Tesseract. A continuación se identificarán los pasos que comprende este nivel en el desarrollo:

#### **OPEN CV**

- 1. Recolección de placas positivas.-** Se deberá obtener todas aquellas imágenes que contendrán el objeto a entrenar, es decir, la placa vehicular. Se debe tener en consideración que un número elevado de imágenes positivas darán un mejor resultado, por ello, será necesario tener 300 imágenes como mínimo. Además se debe considerar ciertos factores como: el ángulo de inclinación, la iluminación, la calidad y el estado de la placa a tomar.



Figura 4.26 Captura de placas para entrenamiento

2. **Recolección de imágenes negativas.-** Las imágenes negativas servirán para diferenciar la placa con el ambiente circundante al que comúnmente se verá afectada. Estas imágenes podrán ser fotografiadas o descargadas de internet, haciendo más fácil su obtención, además se deberá duplicar su número con relación a las imágenes positivas para mejores resultados.



Figura 4.27 Ejemplo de imágenes negativas

3. **Generación de archivos de direcciones.-** Estos archivos contendrán las direcciones de todas las imágenes para el entrenamiento.

Para las imágenes positivas a más de su dirección, contendrán sus coordenadas y dimensiones explicadas en capítulos anteriores, de igual

forma, con el estándar ya definido anteriormente para las imágenes a entrenar se usará p### para las positivas, n### para las negativas.

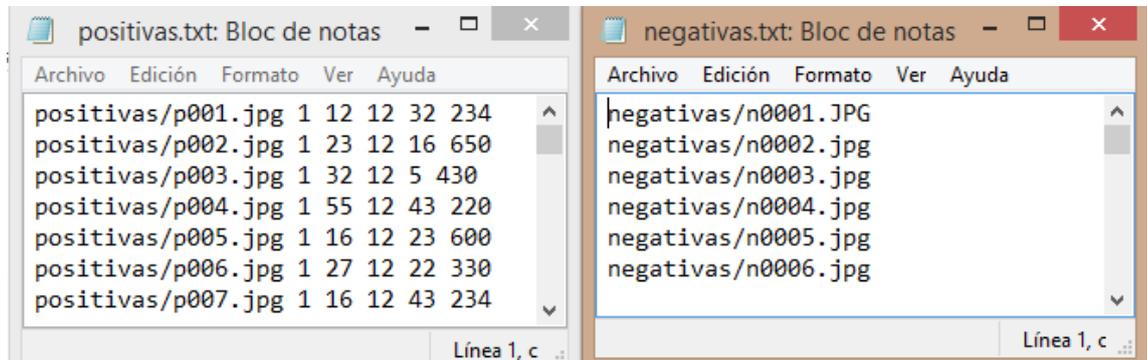


Figura 4.28 Archivos de direcciones para imágenes positivas y negativas

4. **Creación de muestras.-** Mediante el comando **CreateSamples** de Open CV se genera el vector de muestras solo de las imágenes positivas. Este vector especificará el tamaño de las muestras positivas tomando como referencia una resolución estándar. Ejemplo (600x200).
5. **Generación del archivo de entrenamiento.-** Se utilizará el método de entrenamiento **Traincascade** para generar el archivo XML. En este se llamarán a los archivos de direcciones y al vector de muestras. Se realizará diferentes pruebas con la finalidad de obtener el mejor resultado.
6. **Comprobación.-** Finalizada la creación se utilizará el archivo XML de entrenamiento para emplear diferentes pruebas, hasta ver que la placa sea reconocida completamente. Se pretende utilizar del mismo modo la terminal de Open CV para realizar pruebas rápidas.

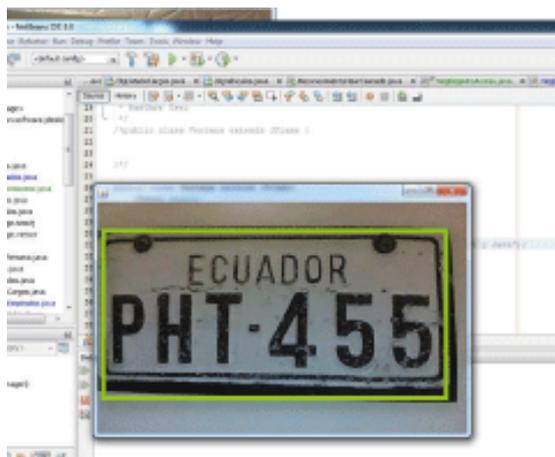


Figura 4.29 Programa de comprobación de resultados

## TESSERACT

1. **Selección de placas positivas.-** Cada placa tomada como muestra positiva para Open CV será seleccionada para formar parte de los archivos en entrenamiento de Tesseract. Las placas con letras ilegibles cortadas o tamaños pequeños serán descartadas para evitar problemas en el entrenamiento.
2. **Edición de placas.-** Este proceso consistirá en extraer cada letra de la placa y pasarla por un proceso de binarización.



Figura 4.30 Extracción de caracteres para entrenamiento OCR

3. **Generación de imagen de entrenamiento.-** Se creará una imagen de formato TIF en la que se agregarán todas las letras de las placas editadas. Es necesario que contenga todas las letras del alfabeto y que mínimo tenga de 4

a 5 repeticiones por carácter, esto con el fin de que la letra este bien definida en la creación del archivo de entrenamiento.

ABCDEFGHIJKLMNOP  
 PQRSTUVWXYZAØÜa  
 bcdefghijklmnop  
 qrstuvwxyz&1234  
 567890

Figura 4.31 Ejemplo de archivo TIF de lenguaje de entrenamiento

- 4. Creación de archivo box.-** Los archivos box contienen las coordenadas y letras en la imagen de entrenamiento. Este proceso, para mayor facilidad, se creará en la herramienta QT. Con el editor solo se seleccionará la letra y se generarán sus coordenadas

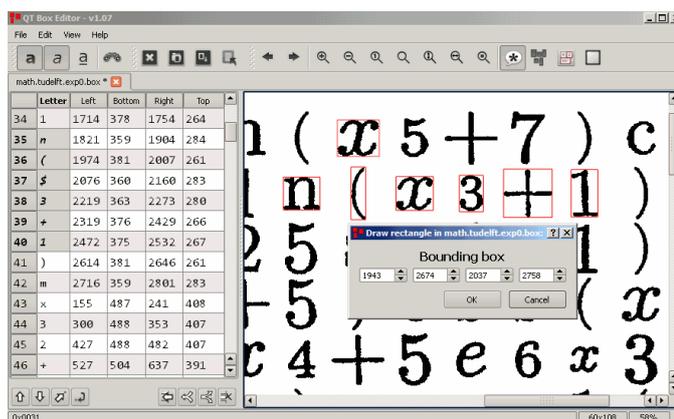


Figura 4.32 Herramienta “QT Box Editor”

Al finalizar el proceso se deberá almacenar su respectivo archivo de formato box.

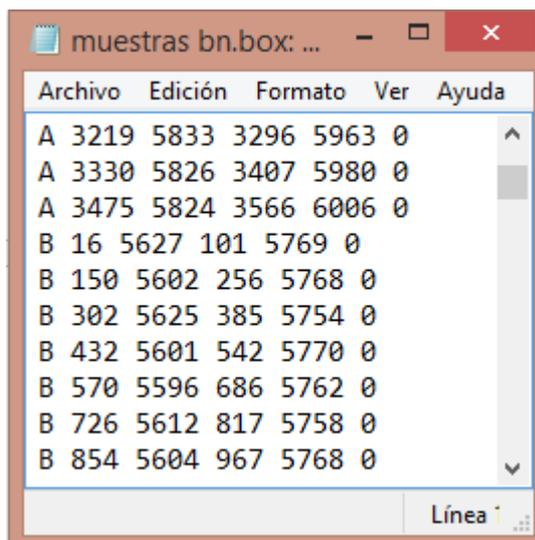


Figura 4.33 Archivo box para entrenamiento

5. **Creación del archivo de entrenamiento.**- Para los archivos TIF y BOX creados se procede a crear un archivo de entrenamiento TR (Training) con tesseract. Esto mediante el comando **tesseract tla.test\_font**.
6. **Creación del archivo UNICHARSET.**- Este archivo relaciona a cada carácter con su código Unicode (Código estándar de caracteres). Este proceso se realiza mediante la herramienta de tesseract **unicharset\_extractor**.
7. **Creación de las propiedades de la fuente.**- Se creará un archivo llamado **font\_properties** con la siguiente estructura:

**<Fontname> <Italic> <Bold> <Fixed> <Serif> <Fraktur>**

**Fontname:** Es el nombre de la fuente que se va a crear.

**Italic:** Se indicará con 1 o 0 si incluirá el tipo Itálica.

**Bold:** Se indicará con 1 o 0 si incluirá el tipo Negrita.

**Fixed:** Se indicará con 1 o 0 si incluirá el tipo Fixed.

**Serif:** Se indicará con 1 o 0 si incluirá el tipo Serif.

**Fraktur:** Se indicará con 1 o 0 si incluirá el tipo Fraktur.

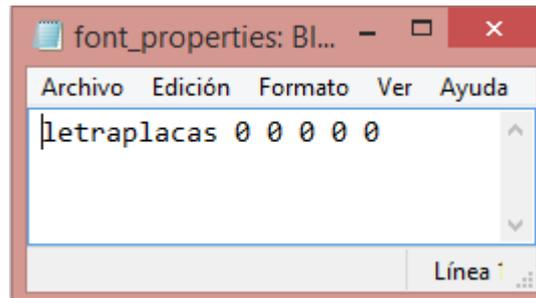


Figura 4.34 Archivo Font\_properties

8. **Creación de la tabla de formas.-** Utilizando la herramienta **shapeclustering** de tesseract se creará el archivo que contendrá la tabla de formas. Adicionalmente se implementará los comandos **mftraining** y **cntraining** para establecer las características de la forma en cada carácter.
  
9. **Organización de archivos para entrenamiento.-** Todos los archivos obtenidos se organizarán en una sola carpeta con el fin de realizar el entrenamiento final. Cada archivo, adicionalmente a su nombre, tendrá un prefijo que indicará que forma parte del mismo lenguaje. Para el sistema se colocará el prefijo **pla**.

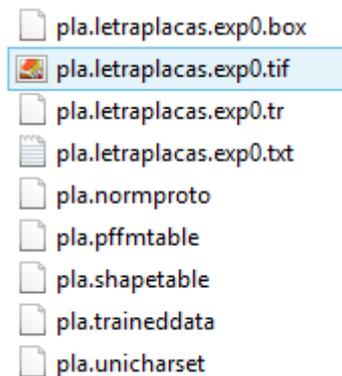


Figura 4.35 Contenido de la carpeta para entrenamiento

10. **Creación del lenguaje de entrenamiento.-** Como último paso se realizará el proceso de creación del lenguaje de entrenamiento con el comando **combine\_tessdata** para empaquetar y generar un único archivo.

## Nivel 1 o detección

Representa el proceso inicial del reconocimiento de la placa. Este implementará los siguientes pasos:

- 1. Ejecución del proceso Open CV.-** Como proceso inicial se ejecuta la llamada al API de Open CV junto con el archivo de entrenamiento lo que ejecutará el escaneo de la cámara. Este proceso se mantiene encendido durante todo el tiempo hasta que el usuario decida apagarlo.
- 2. Detección de la placa.-** Open CV continuará en ejecución hasta detectar la placa, una vez detectada la placa pausará su reconocimiento permitiendo el ingreso del vehículo.
- 3. Almacenamiento de la captura.-** Luego de la detección Open CV almacenará la imagen detectada.



Figura 4.36 Ejemplo imagen almacenada en proceso de captura.

## Nivel 2 o procesamiento

El nivel 2 representa todo lo relacionado a la edición y procesamiento de la placa. Maneja los siguientes pasos:

- 1. Recorte de la imagen.-** Se realizará el recorte de la placa ya delimitada por Open CV, dejando solamente la zona de interés y almacenándola en el equipo.



Figura 4.37 Ejemplo de placa recortada.

2. **Binarización de la imagen.-** La imagen detectada pasa por un proceso de binarización para delimitar solo dos niveles de color: **blanco y negro**



Figura 4.38 Ejemplo de placa binarizada.

### Nivel 3 o OCR

Es el nivel de digitalización de la placa en el que se obtendrá el texto para su verificación en la base de datos. Y consta de los siguientes pasos:

1. **Llamada al API de Tesseract.-** Esta ejecución se realizará mediante la librería Tess4j en donde se realizará el proceso mencionado de digitalización de los caracteres reconocidos.
2. **Ejecución de la lectura de la placa.-** Se realizará mediante el comando `doOCR()`, la lectura de la imagen con el lenguaje entrenado en el nivel 0. Y se obtendrá como resultado los caracteres de la placa.



Figura 4.39 Ejemplo de detección de caracteres

**Nivel 4 o comparación.**

Este nivel será manejado solo por el sistema y el gestor de base de datos. Los pasos a realizarse dentro de este nivel consistirán en:

- 1. Almacenamiento de registro de ingreso temporal.-** Se almacenará la primera vez el registro con los caracteres de la placa obtenidos por la lectura de Tesseract.
- 2. Búsqueda de la placa.-** Se realizará una consulta de la placa a la tabla de vehículos. También se obtendrá la información del empleado registrado con ese vehículo.
- 3. Registro de resultado.-** Se procede a realizar el registro del ingreso con la información del usuario, con el estado y los resultados obtenidos del tratamiento de las imágenes capturadas.
- 4. Muestra de resultados.-** Finalmente se visualizarán los resultados en la interfaz de usuario indicando el acceso, esta visualización corresponderá a la imagen binarizada.

**4.7.2.2. Observaciones**

El sistema de detección no es perfecto, por lo que pueden ocurrir problemas de ingreso en cualquiera de los niveles especificados anteriormente. Los errores pueden ocasionar problemas que llegan a afectar la fluidez del sistema, por lo que se han tomado en cuenta ciertas observaciones para cada nivel de detección.

**Nivel 0**

En los procesos de entrenamiento son necesarios varios intentos, ya sea aumentando el número de muestras, cambiando sus enfoques, tamaños, etc; con la finalidad de llegar al nivel deseado. La cantidad de imágenes de muestra son muy

importantes así que se deben obtener el mayor número posible para un entrenamiento exitoso.

Para el archivo de entrenamiento OCR son necesarias varias repeticiones por letra para identificar correctamente la forma de los caracteres de la placa, además de verificar que el archivo box generado tenga todos los caracteres a considerar en el lenguaje a entrenar.

### **Nivel 1**

En el nivel de detección es necesario colocar la cámara en un ángulo correcto con buena iluminación para obtener el mejor enfoque posible y así evitar problemas de reconocimiento. Además se ha contemplado un botón para acceso manual y así mitigar problemas con vehículos sin placa o defectuosos imposibles de reconocer.

### **Nivel 2**

Corresponde al nivel de procesamiento donde difícilmente podrán ocurrir problemas, ya que estos dependen más del resultado del nivel 1, pero si el proceso llegará a detenerse, se registrará el último proceso en el que se quedó.

### **Nivel 3**

Se debe tener en cuenta que toda placa necesita ser completamente clara en sus letras a detectar para que sean legibles por el OCR. Placas sucias o despintadas pueden resultar mal interpretadas por Tesseract y terminar en resultados no deseados. Se realizarán validaciones de estructura (los 4 o 3 primeros letras, luego guion y finalmente 4 o 3 números) para intentar depurar caracteres incorrectos.

### **Nivel 4**

Cuando la placa no sea encontrada en la base de datos se denegará el acceso, pero el usuario supervisor podrá tener la opción de realizar un acceso manual. Toda esta actividad también será registrada.

#### 4.7.2.3. Resultados esperados

Todos los pasos planteados en este capítulo servirán de guía en el proceso de desarrollo para obtener los mejores resultados posibles. Como se ha indicado con anterioridad el sistema maneja una cadena de operaciones en la que se inicia con la detección del objeto y finaliza con la lectura de la placa, en cada nivel esta se irá depurando para obtener los resultados deseados.

De cada proceso se espera los mejores resultados teniendo en cuenta las siguientes características:

- **Integración correcta de las herramientas.-** Todas las herramientas han sido seleccionadas para presentar el mínimo inconveniente posible en la integración, al implementarlas en el proceso de programación no se debería tener problemas de compatibilidad.
- **Velocidad y procesamiento mínimo.-** Se han seleccionado las herramientas y procesamientos puntuales para cada operación en el proceso de detección y reconocimiento, estos no realizan un consumo excesivo de recursos y ejecutan sus operaciones en milisegundos. Debido a ello toda la cadena de detección debe tener una duración mínima.
- **Entrenamientos de placas y caracteres robustos.-** Los procesos de entrenamiento fijados junto con un número elevado de muestras, logran un proceso de entrenamiento robusto para las herramientas Open CV y Tesseract.
- **Errores mitigados y notificados.-** Con el nivel de control e informes que poseerá el sistema permitirá registrar excepciones en el proceso de detección así como notificará las excepciones para permitir acceso manual.
- **Detección de placas exitosa.-** Representa el mayor objetivo a lograrse por parte del sistema, ya que si no se obtiene este resultado el sistema no funcionaría. Todo el sistema está planteado para procesar la imagen de tal manera que se obtengan buenos resultados, para ello se ha organizado niveles

y procedimientos en cada etapa, para poder enfocarse en el desarrollo y la obtención de buenos resultados en cada nivel.

### **Conclusión**

Este capítulo desarrolló todos los procesos detallados técnicamente para definir la estructura que se manejará a lo largo de la construcción del sistema. Cada proceso se identifica por niveles para mantener siempre un orden, y permitir diferenciar su información de entrada y salida. Finalmente se ha especificado los procedimientos que se deben seguir para las etapas de entrenamiento y desarrollo, con el objetivo de guiar y organizar la futura implementación del sistema.

## Capítulo 5

### Desarrollo del sistema

#### Introducción

Luego de preparar toda la información pertinente al sistema se procederá a realizar su desarrollo. Inicialmente se definen todas aquellas herramientas y utilitarios que servirán de complemento, con el fin de permitir una total sincronización con los módulos principales de desarrollo (Open CV y Tesseract). Se partirá de los procesos de entrenamiento para luego realizar la codificación del sistema. Además se especificará todo el flujo de desarrollo, los elementos que intervienen, los cambios en el sistema y los resultados obtenidos en cada fase.

#### 5.1. Herramientas de programación

A continuación se definirán las principales herramientas de desarrollo que serán utilizadas para el proceso de codificación y de almacenamiento de los datos.

##### 5.1.1. Gestor de Bases de Datos

Existen diversos tipos de bases de datos en la actualidad, pero por motivos de rendimiento, fácil manejo y portabilidad, se ha escogido el conocido gestor de bases de datos My Sql.

#### My Sql

Es un gestor de bases de datos escrito en C y en C++, de manejo simple y con una exigencia mínima de recursos, además de realizar transacciones rápidas y efectivas por lo cual es ideal para ser utilizada en el desarrollo del sistema. A continuación se listarán sus principales características cuya funcionalidad va ligada a los objetivos que debe cumplir el sistema: (MySql, s.f.)

- **Portabilidad.-** Esta herramienta se encuentra disponible para múltiples plataformas entre ellas Windows, Linux, Open Solaris y Mac OS por lo que permitirá al aplicativo de detección ser adaptable a cualquier sistema operativo, además posee diversas APIs desarrolladas en C, C++, Eiffel, Java,

Perl, PHP, Python, Ruby, y Tcl, que puedan ser utilizadas para adaptar el sistema a otros lenguajes de programación como es el caso de las herramientas empleadas: Open CV y Tesseract.

- **Rapidez.-** Mysql utiliza diversas tecnologías que la hacen rápida, entre ellas sus algoritmos de multi-threaded, multi-join y reserva rápida de memoria, que junto a unas librerías altamente optimizadas permiten aprovechar múltiples CPUs al realizar una consulta, lo que las hacen ideales para un alto grado de transaccionalidad y velocidad de ejecución que requerirá el sistema.
- **Segura y estable.-** Mysql soporta grandes cantidades de registros sin presentar inconvenientes, por lo que es ideal para brindar estabilidad al sistema a desarrolla. En cuanto a seguridad MySQL, tiene un sistema de verificación basado en el host que permite cifrar todo el tráfico al conectarse a un servidor.
- **Liviano, fácil y libre.-** Como se ha especificado, Mysql es una herramienta poderosa y totalmente gratis, por lo que no habrá inconveniente para el usuario al momento de instalar el sistema y no tendrá que preocuparse por licencias ni costos adicionales de software, además sus requerimientos no son excesivos y posee un sin número de herramientas o interfaces fáciles de usar para la administración de la base de datos.

Entre los frameworks (marcos de trabajo) más populares para MySQL se ha seleccionado “MySQL Workbench”

### **MySQL Workbench**

Por motivos de desarrollo se ha utilizado la herramienta gratuita recomendada por Oracle para el manejo de MySQL denominada Workbench 6.1, ya que se ha comprobado que es una herramienta muy completa, que permite realizar desde un modelado previo (diagrama E/R) y relacionarlo con una base de datos automáticamente. Además, permite la conexión ágil a cualquier servidor de bases de

datos externo, de una manera rápida y fácil, para trabajar desde diferentes equipos sin ningún problema (Oracle Corporation, 2014).

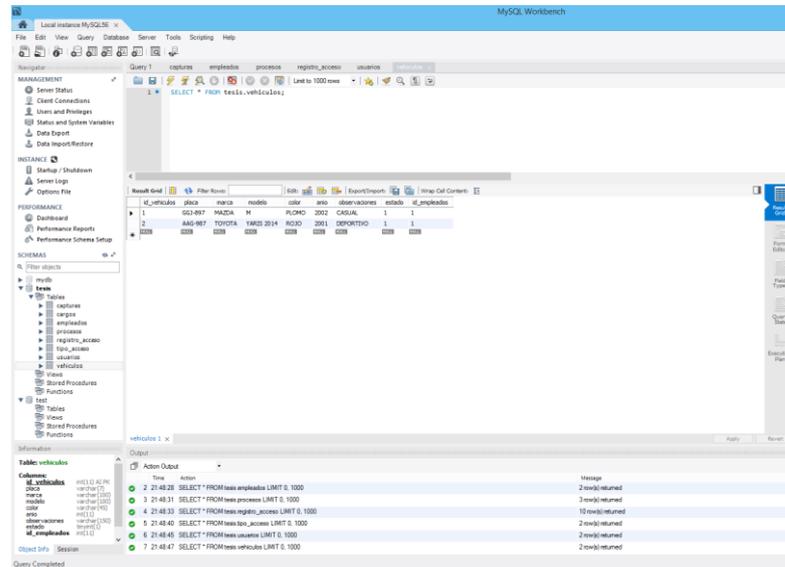


Figura 5.40 Interfaz Mysql Workbench

### 5.1.2. Lenguajes de programación

Los lenguajes de programación presentarán la estructura principal del sistema, permitirán el control, tanto de la base de datos, como de las APIs que serán utilizadas para el funcionamiento del sistema. La decisión de seleccionar el lenguaje de programación, no solo repercutirá en su calidad como lenguaje, sino también en la compatibilidad y soporte que tenga sobre las APIs de visión artificial y OCR.

Luego de una investigación previa, se decidió seleccionar el lenguaje de programación **JAVA** y su framework más popular y robusto que es Netbeans.

## Netbeans IDE

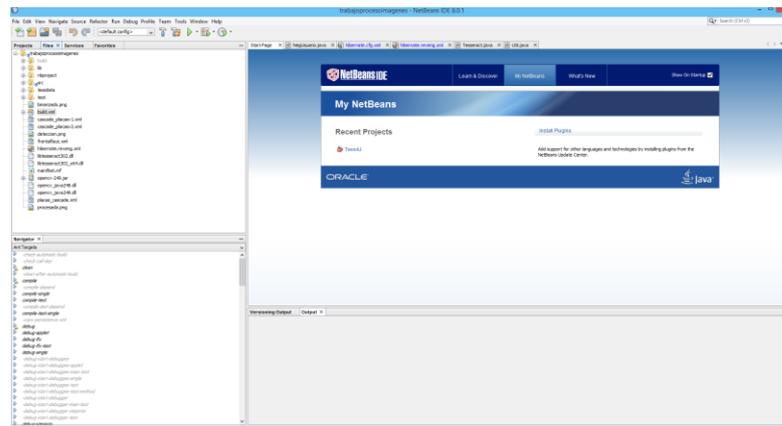


Figura 5.41 Interfaz Netbeans.

Netbeans es un entorno de desarrollo mayormente en código JAVA que permite escribir, compilar, depurar y ejecutar programas. Esta completa herramienta presenta un gran número de características que ayudarán al desarrollo del sistema (Netbeans, s.f.):

- **Soporte.-** Netbeans es una interfaz muy robusta con una gran comunidad que han desarrollado y utilizado herramientas de visión artificial, además de presentar tutoriales y foros de consulta que dan soporte a la correcta adaptación entre sus librerías.
- **Compatibilidad.-** Netbeans es el framework más popular de código JAVA, con millones de usuarios desarrolladores que usan esta interfaz, por lo que presenta una gran cantidad de aplicativos que permiten la compatibilidad con varias tecnologías incluida la visión artificial.
- **Integración Mysql.-** Desarrollada por la misma empresa que Mysql, Netbeans permite una correcta integración con este gestor de bases de datos, incluyendo herramientas propias que realizan registro de servidores, modificación de bases de datos, etc.

Además de la integración de bases de datos, es necesario integrar las herramientas de Tesseract y Open CV, para ello se identifican 2 librerías que facilitan la gestión:

- **Tess4j.-** Conjunto de librerías que proveen la utilización de todas las bondades del motor de OCR Tesseract. (Tess4j, 2014)
- **OpenCV-249.-** OpenCV dispone desde su misma instalación una librería java para el uso de sus características de visión artificial en Netbeans (Open CV developer team, s.f.).
- **Creación dinámica y rápida.-** El principal objetivo del sistema es el manejo de visión artificial, por lo que la fácil creación de formularios visuales y su gestor de creación automática JSF CRUD permitirá crear rápidamente la interfaz.
- **Versionamiento.-** Debido a que este proyecto se basará en el mejoramiento progresivo de cada nivel de detección, es necesario mantener un orden y respaldo de cada versión que se vaya desarrollando. Netbeans facilita esta tarea con herramientas que permitirán controlar y compartir el contenido del sistema con cada uno de los miembros del proyecto. En este caso se utilizará la herramienta “Subversion” para realizar dicha gestión.

## 5.2. Evaluación y selección de componentes

Es necesario determinar todos aquellos componentes que formarán parte de la etapa de desarrollo y pruebas. Estos componentes deben ser evaluados para alinearse a los requerimientos que debe cumplir el sistema.

### 5.2.1. Componentes externos necesarios

Se analizará las características que debe tener un componente para determinar los parámetros que luego se evaluarán.

## Cámara

Como se había indicado anteriormente, la cámara es el elemento fundamental para obtener la imagen y de ella dependerá la calidad de la captura. Se establecerá qué tipo de cámara se utilizará para la simulación del ambiente de detección.

Entre los parámetros a considerar se tienen:

- **Resolución equilibrada:** A mayor resolución, una mejor detección, pero mientras más grande sea la imagen, más lenta será el proceso de binarización o más recursos necesitará.
- **Zoom para distancias grandes:** Ya que la cámara se ubicará a una distancia considerable del vehículo, es necesario contemplar la posibilidad que requerirá un nivel de zoom para una mejor detección.
- **Captura sin atrasos:** Las imágenes captadas deben ser transmitidas con el menor atraso posible para que la detección sea inmediata.
- **Cámara ajustable:** Debe acoplarse a cualquier ambiente y tener un amplio rango de visión.

## Iluminación y ambiente

La zona de detección debe contener una buena iluminación y espacio para obtener buenos resultados en la detección.

Entre las características recomendadas para una detección óptima se enumeran:

- **Distancia de detección considerable:** La cámara debe ubicarse a una distancia de 1 a 2 metros de la zona de detección del vehículo.
- **Ajuste de iluminación:** La iluminación debe ubicarse de tal manera, que no permita un brillo demasiado intenso para evitar reflejo en la placa o sombras. También existen cámaras que facilitan este proceso atenuando la imagen para adaptarla en ambientes oscuros.

## Computador para entrenamiento y programación

Ya se han listado los requerimientos mínimos necesarios para las etapas de entrenamiento y detección, ahora se tendrá en cuenta que el equipo cumpla con las especificaciones mencionadas.

### 5.2.2. Evaluación de componentes

Cada componente debe pasar por un proceso de aprobación y debe cumplir con las características mencionadas.

#### Evaluación de cámara

Se necesita escoger una cámara ideal para el proceso de pruebas y simulación, estas deben cumplir todas las características propuestas, por ello se creará un cuadro comparativo, y así seleccionar la mejor opción:

	 Klip Xcam 2100	 Microsoft HD-3000	 HDE- Y101
<b>Resolución</b>	2 megapíxeles (1600x1200)	5 megapíxeles (2560x2048)	5 megapíxeles (2560x2048)
<b>Zoom</b>	2x	3x	2x
<b>Movilidad</b>	Rotación omnidireccional	360° de rotación	Ajuste de altura y rotación
<b>Iluminación</b>	No	TrueColor (Ajuste de brillo y saturación de color en ambientes oscuros)	6 leds
<b>Respuesta</b>	Lenta	Rápida	Lenta

Cuadro 5.7 Comparación de cámaras web

## Equipos para entrenamiento y desarrollo

Los requerimientos mínimos y recomendados se han establecido con anterioridad ahora se han designado equipos que se utilizarán para los diferentes procesos

		
	<b>Equipo de escritorio</b>	<b>Equipo portátil</b>
<b>Procesador</b>	Intel Core i7	Intel Core i3
<b>RAM</b>	16 Gb	3Gb
<b>Disco Duro</b>	1TB	500gb
<b>Sistema Operativo</b>	Windows 8.1	Windows 7
<b>Designación</b>	Procesos de entrenamiento de Tesseract / Open CV y programación	Programación y pruebas de rendimiento y velocidad del sistema

Cuadro 5.8 Evaluación de equipos

Cada equipo tiene tareas designadas que cumplen con los requerimientos de cada proceso de programación. La computadora portátil tiene recursos bajos, lo que será ideal en pruebas de rendimiento, y de esa forma, verificar la velocidad de respuesta del sistema; mientras que la computadora de escritorio posee potentes características para agilizar el proceso de entrenamiento.

### 5.3. Desarrollo del sistema

Para el proceso de desarrollo, se preparará todos los elementos de hardware y software necesarios para la programación e instalación de todas las herramientas requeridas como: Netbeans, Mysql Server, Mysql Workbench, Tesseract y Open CV. Finalizado el proceso se realiza la fase de codificación.

#### 5.3.1. Fase de codificación

Como fase inicial, se procede a realizar el entrenamiento de objetos en OpenCV y Tesseract. Cada herramienta tiene un proceso diferente de entrenamiento, por ello, es necesario establecer una serie de parámetros, que permitan el entrenamiento de la placa, así como delimitar y validar las características de detección de los caracteres.

##### 5.3.1.1. Entrenamiento

###### Open CV

Se recopila la información requerida para el entrenamiento en Open CV:

Elementos disponibles para el entrenamiento Open CV	
Imágenes positivas	554 imágenes en JPG
Imágenes negativas	1016 imágenes en JPG
Archivos de direcciones	Positivas.txt y Negativas.txt

Cuadro 5.9 Elementos para entrenamiento Open CV

Una vez preparada la información, se debe realizar el proceso de entrenamiento y para ello se necesita la consola de Windows para ejecutar los archivos de entrenamiento.

Inicialmente es necesario ubicarse en la dirección donde se encuentran todos los archivos binarios de entrenamiento. En este caso, se utilizará un sistema operativo de 64 bits por ello la dirección será:

```
Cd E:\opencv\build\x64\vc12\bin
```

Luego se procede a crear un archivo de muestras mediante el método **create\_samples** solo con las imágenes positivas, en donde se especifica:

```
opencv_createsamples  
-info placas/positivas.txt  
-vec entrenamiento27102014/positivas.vec  
-num 554 -w 60 -h 27
```

**-info:** La ubicación del archivo de información de direcciones positivas.txt.

**-vec:** El nombre del vector resultante en este caso positivas.vec.

**-num:** El número de imágenes positivas.

**-w:** Ancho final de las muestras.

**-h:** Alto final de las muestras.

El ancho y alto final de la muestra definirá la proporción y tamaño del objeto detectado, esto especificará el tamaño del objeto en el ambiente.

Una vez creado el archivo se obtiene un resumen de la información del vector de muestras creado.

```

E:\Tesis\opencv\build\x64\vc12\bin>opencv_createsamples -info placas/positivas2.
txt -vec entrenamiento27102014/positivas.vec -num 554 -w 60-h 27
Info file name: placas/positivas2.txt
Img file name: <NULL>
Vec file name: entrenamiento27102014/positivas.vec
BG file name: <NULL>
Num: 554
BG color: 0
BG threshold: 80
Invert: FALSE
Max intensity deviation: 40
Max x angle: 1.1
Max y angle: 1.1
Max z angle: 0.5
Show samples: FALSE
Width: 60
Height: 24
Create training samples from images collection...
placas/positivas2.txt(520) : parse errorDone. Created 519 samples
E:\Tesis\opencv\build\x64\vc12\bin>

```

Figura 5.42 Generación del vector de muestras Open CV

Luego de generar el vector de muestras, se procede a la creación del archivo de entrenamiento, para ello se hace el uso del comando **opencv\_traincascade**.

```

opencv_traincascade -data data/cascade27
-vec entrenamiento27102014/positivas.vec
-bg placas/negativas.txt
-numStages 20
-minHitRate 0.999
-maxFalseAlarmRate 0.5
-numPos 554
-numNeg 1016 -w 60 -h 27
-mode ALL
-precvalBufSize 2048 -precalcIdxBufSize 2048

```

**-data:** Indica donde se guardarán los archivos de entrenamiento.

**-vec:** La ubicación del vector de muestras creado anteriormente positivas.vec.

**-bg:** El archivo de información de las imágenes negativas que serán usadas como fondo para el entrenamiento.

**-numStages:** Número de etapas de entrenamiento recomendado en 20.

**-minHitRate:** Tasa mínima de falsa alarma para cada etapa de entrenamiento, es proporcional al número de etapas.

- maxFalseAlarmRate**: Tasa máxima de falsa alarma para cada etapa de entrenamiento, es proporcional al número de etapas.
- numPos**: Número de imágenes positivas.
- numNeg**: Número de imágenes negativas.
- w**: Ancho final de la placa.
- h**: Alto de la placa.
- mode**: **All**; Determina que se posean todas las características de entrenamiento vertical y 45° de rotación.
- precalcValBufSize**: Tamaño en MB del buffer de valores precalculados a la memoria del sistema
- precalcIdxBufSize**: Tamaño en MB del buffer de índices precalculados a la memoria del sistema. (Open CV, 2013)

Se inicia el proceso de entrenamiento desde la etapa 0 hasta la 19, donde se calcula un tiempo estimado de 4 a 5 días de entrenamiento.

```

C:\Windows\system32\cmd.exe - opencv_traincascade -data data/cascade26 -...
PARAMETERS:
cascadeDirName: data/cascade26
vecFileName: entrenamiento26102014/positivas.vec
bgFileName: placas/negativas.txt
numPos: 554
numNeg: 1016
numStages: 21
precalcValBufSize[Mb] : 1024
precalcIdxBufSize[Mb] : 1024
stageType: BOOST
featureType: HAAR
sampleWidth: 60
sampleHeight: 20
boostType: GAB
minHitRate: 0.999
maxFalseAlarmRate: 0.5
weightTrimRate: 0.95
maxDepth: 1
maxWeakCount: 100
mode: ALL

==== TRAINING 0-stage ====
<BEGIN
POS count : consumed   554 : 554
NEG count : acceptanceRatio  1016 : 1
Precalculation time: 9.516
+-----+-----+
| N |   HR |   FA |
+-----+-----+

```

Figura 5.43 Generación archivo de entrenamiento Open CV

El proceso finaliza con la obtención de un listado de archivos de entrenamiento, sin embargo, el único aplicable en el sistema es el denominado **cascade.xml**

 cascade.xml	31/10/2014 10:21	Archivo XML	53 KB
 params.xml	30/10/2014 23:46	Archivo XML	1 KB
 stage0.xml	30/10/2014 23:46	Archivo XML	1 KB
 stage1.xml	31/10/2014 0:01	Archivo XML	2 KB

Figura 5.44 Cascade.xml generado

Este archivo de entrenamiento servirá para la identificación de la placa en el proceso de detección.

### Tesseract

El entrenamiento es similar a Open CV, ya que se debe usar la consola de Windows para ejecutar los comandos de entrenamiento. Para ello se prepara la información necesaria para el entrenamiento:

Elementos disponibles para entrenamiento Tesseract	
Imágenes de letras	Selección de placas que contienen las letras en perfecto estado para entrenamiento

Cuadro 5.10 Elementos para entrenamiento Tesseract

Una vez seleccionadas las placas con los caracteres de todo el alfabeto, se procede a recortar y editar a cada una para obtener un carácter limpio y bien definido

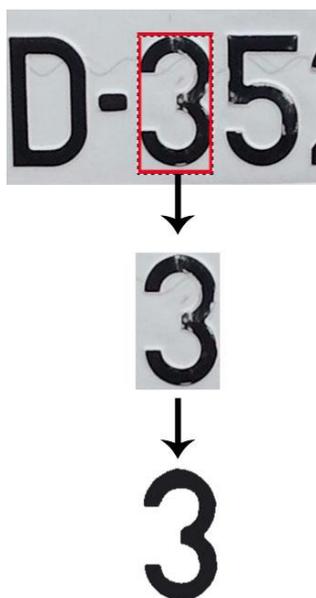


Figura 5.45 Depuración de los caracteres para entrenamiento.

Se obtiene todos los caracteres del alfabeto y los 10 dígitos numéricos para guardarlos en un archivo de imagen de formato TIF. Los caracteres se repiten 5 veces cada uno para cumplir con los estándares recomendados de entrenamiento de Tesseract

**POI-462 OQD-941 BUX-385 UNP-923 RSU-146**  
**ANL-316 TCE-159 ZJO-029 NEY-407 QVR-378**  
**BEI-574 MAY-237 XED-165 XJK-730 SRV-129**  
**TPB-933 PGZ-680 QWS-352 HDQ-975 VLM-015**  
**ACA-081 LXF-142 JDZ-183 YCM-124 HNQ-164**  
**EHC-295 UFW-397 WFV-528 LZP-426 AJV-627**  
**ALD-827 RVK-804 OYT-709 IYX-195**  
**GIP-640 GW0-152 HSU-241 MHJ-538**  
**GEM-153 FBK-597 RKC-547 KSW-673**

Figura 5.46 Archivo para entrenamiento de Tesseract

A continuación se crea el archivo box con la ayuda de un editor denominado **QT Box v1.10**, éste permite de una manera más rápida detectar las coordenadas de cada carácter.

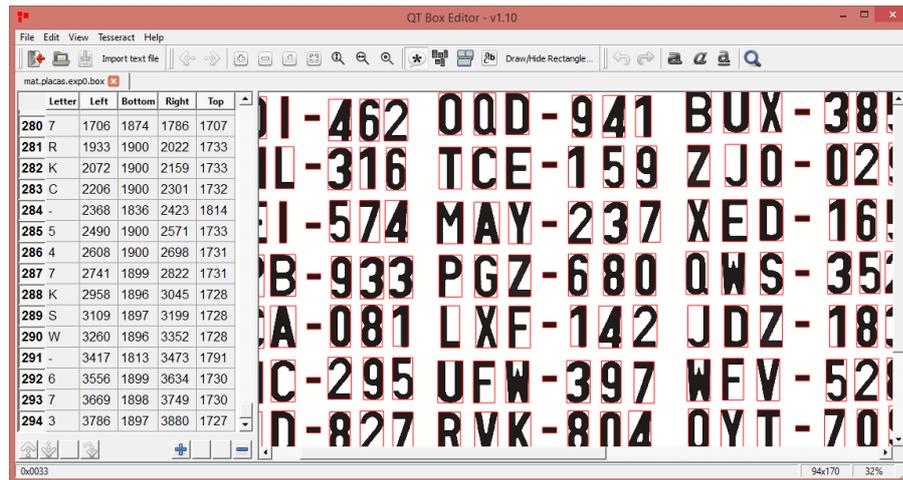


Figura 5.47 QT Box generador de coordenadas de caracteres

El resultado final será un archivo .BOX que indicará la ubicación de cada carácter dentro de la imagen de entrenamiento TIF.

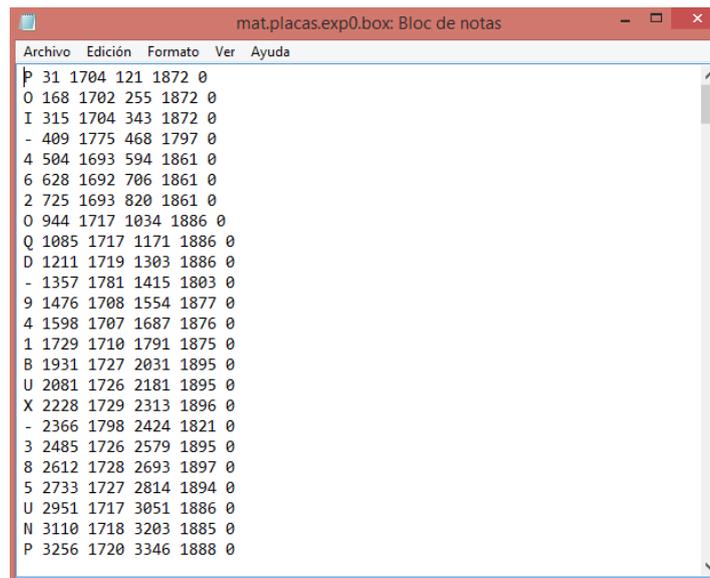


Figura 5.48 Archivo box generado

Una vez obtenido los archivos TIF y BOX, se renombran los mismos para mantener una estructura ordenada y reconocible para tesseract, en este caso en particular los archivos serán:

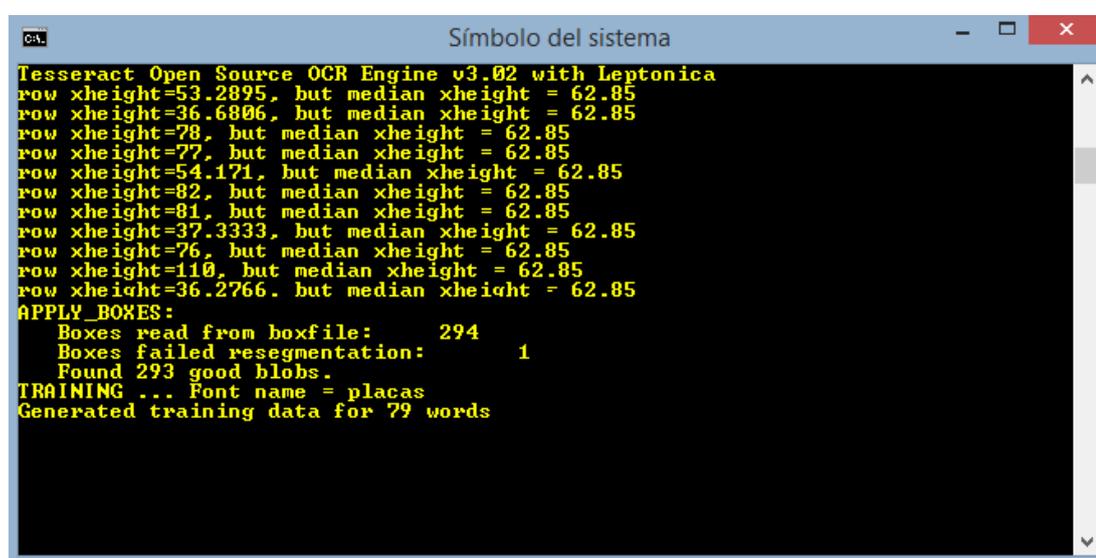
 mat.placas.exp0.box	02/11/2014 11:09	Archivo BOX
 mat.placas.exp0.png	02/11/2014 10:39	Archivo PNG

Luego se inicia el entrenamiento de las letras de la placa como un nuevo lenguaje en tesseract, para ello se ingresa en la terminal de Windows y se ubica el directorio de los archivos binarios de Tesseract.

```
Cd E:\tesseract\Tesseract-OCR
```

Se empareja la imagen TIF con el archivo BOX mediante el comando **box.train**.

```
tesseract mat.placas.exp0.tif mat.placas.exp0 nobatch box.train
```



```

Tesseract Open Source OCR Engine v3.02 with Leptonica
row xheight=53.2895, but median xheight = 62.85
row xheight=36.6806, but median xheight = 62.85
row xheight=78, but median xheight = 62.85
row xheight=77, but median xheight = 62.85
row xheight=54.171, but median xheight = 62.85
row xheight=82, but median xheight = 62.85
row xheight=81, but median xheight = 62.85
row xheight=37.3333, but median xheight = 62.85
row xheight=76, but median xheight = 62.85
row xheight=110, but median xheight = 62.85
row xheight=36.2766, but median xheight = 62.85
APPLY_BOXES:
  Boxes read from boxfile: 294
  Boxes failed resegmentation: 1
  Found 293 good blobs.
TRAINING ... Font name = placas
Generated training data for 79 words
  
```

Figura 5.49 Generación del archivo training

Luego se genera el archivo unicharset con el comando siguiente:

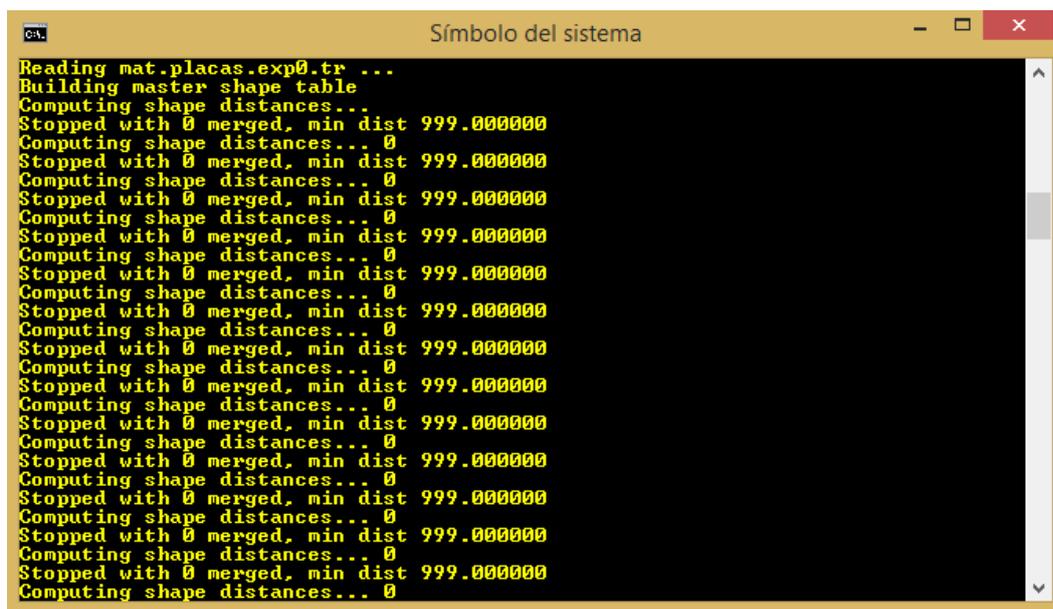
```
unicharset_extractor mat.placas.exp0.box
```

Se procede a crear el font\_properties o propiedades de la fuente. Para el sistema la fuente se llamará “placas” con ninguna característica especial ya que la letra será estándar, por lo que el archivo se verá de la siguiente forma:

```
placas 0 0 0 0 0
```

A continuación, se debe crear el archivo de formas de cada carácter **shapetable**, para ello se añaden los archivos font\_properties y el unicharset.

```
shapeclustering -F font_properties -U unicharset mat.placas.exp0.tr
```



```

C:\> shapeclustering -F font_properties -U unicharset mat.placas.exp0.tr

Reading mat.placas.exp0.tr ...
Building master shape table
Computing shape distances...
Stopped with 0 merged, min dist 999.000000
Computing shape distances... 0
Stopped with 0 merged, min dist 999.000000
Computing shape distances... 0
Stopped with 0 merged, min dist 999.000000
Computing shape distances... 0
Stopped with 0 merged, min dist 999.000000
Computing shape distances... 0
Stopped with 0 merged, min dist 999.000000
Computing shape distances... 0
Stopped with 0 merged, min dist 999.000000
Computing shape distances... 0
Stopped with 0 merged, min dist 999.000000
Computing shape distances... 0
Stopped with 0 merged, min dist 999.000000
Computing shape distances... 0
Stopped with 0 merged, min dist 999.000000
Computing shape distances... 0
Stopped with 0 merged, min dist 999.000000
Computing shape distances... 0
Stopped with 0 merged, min dist 999.000000
Computing shape distances... 0
Stopped with 0 merged, min dist 999.000000
Computing shape distances... 0
Stopped with 0 merged, min dist 999.000000
Computing shape distances... 0
Stopped with 0 merged, min dist 999.000000
Computing shape distances... 0
Stopped with 0 merged, min dist 999.000000
Computing shape distances... 0
Stopped with 0 merged, min dist 999.000000
Computing shape distances... 0
Stopped with 0 merged, min dist 999.000000
Computing shape distances... 0
Stopped with 0 merged, min dist 999.000000
Computing shape distances... 0
Stopped with 0 merged, min dist 999.000000
Computing shape distances... 0

```

Figura 5.50 Generación de la Shapetable

Se continúa generando los archivos de características y prototipos de caracteres con la sentencia detallada a continuación:

```
mftraining -F font_properties -U unicharset -O mat.unicharset
mat.placas.exp0.tr
```

Seguidamente se crea el archivo de normalización de caracteres para ello se ejecuta el comando:

```
cntraining mat.placas.exp0.tr
```

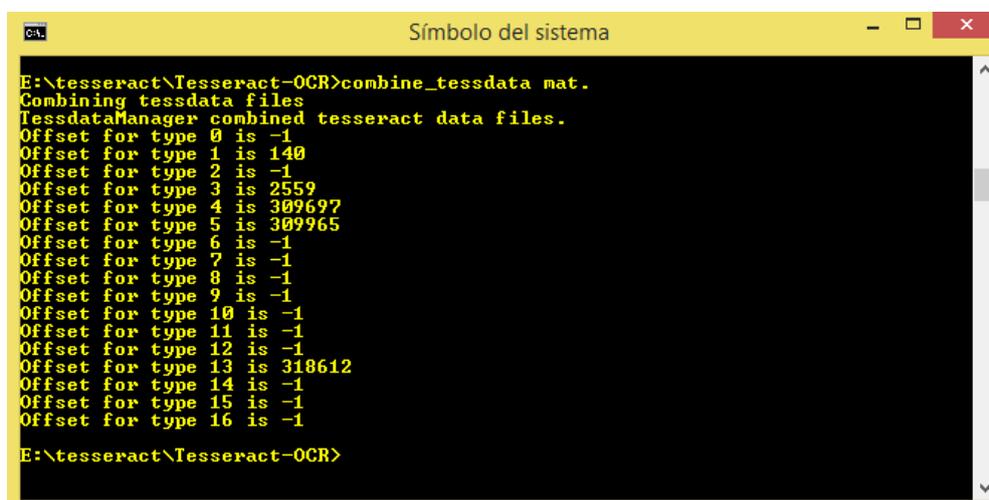
A continuación es necesario renombrar los archivos generados para empaquetarlos, se usará el prefijo **mat** del lenguaje.

-  mat.inttemp
-  mat.normproto
-  mat.pffmtable
-  mat.shapetable

Figura 5.51 Archivos renombrados para entrenamiento

Finalmente con todos los archivos correctamente generados y renombrados, estos se ubican en un mismo directorio para crear el archivo traindata.

```
combine_tessdata mat.
```



```

E:\tesseract\Tesseract-OCR>combine_tessdata mat.
Combining tessdata files
TessdataManager combined tesseract data files.
Offset for type 0 is -1
Offset for type 1 is 140
Offset for type 2 is -1
Offset for type 3 is 2559
Offset for type 4 is 309697
Offset for type 5 is 309965
Offset for type 6 is -1
Offset for type 7 is -1
Offset for type 8 is -1
Offset for type 9 is -1
Offset for type 10 is -1
Offset for type 11 is -1
Offset for type 12 is -1
Offset for type 13 is 310612
Offset for type 14 is -1
Offset for type 15 is -1
Offset for type 16 is -1

E:\tesseract\Tesseract-OCR>

```

Figura 5.52 Generación del archivo final traindata

Luego de generado, el archivo de lenguaje **mat.traineddata** se utilizará para el reconocimiento de caracteres de la placa.

### 5.3.1.2. Sistema (Aplicativo)

En este apartado se enfocará en cómo se desarrolló el aplicativo desde su preparación hasta especificar cada módulo de su interfaz.

#### 5.3.1.2.1. Preparación de librerías para el sistema

Para gestionar la conexión a la base de datos y el manejo de la misma, se aplicó “Hibernate”, esta es una herramienta de mapeo (objeto-relacional) que permite la persistencia de datos, aplicada a las bases de datos relacionales. (Hibernate, s.f.).

Se desarrolló el aplicativo sobre JDK 7 (Java Development Kit) que es la herramienta de desarrollo oficial de JAVA, donde se incluye JRE (Java Runtime Environment) que es compilador e intérprete de este código. Dentro del proyecto se incluirá las librerías:

- **Opencv-249:** Como se indicó anteriormente esta es la librería que permite hacer el reconocimiento del patrón deseado, además de facilitar el manejo de imágenes.
- **Swingx-all-1.6.4:** Es la librería que da una mejor presentación visual a la interfaz del sistema.

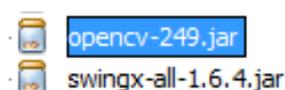


Figura 5.53 Librerías externas JAVA

Al implementar la librería opencv-249.jar se tuvo acceso a todas las funcionalidades Open CV que fueron aplicadas para el reconocimiento del patrón, usando el resultado del entrenamiento que ya fue especificado, la forma de cargar este archivo es la siguiente:

```
CascadeClassifier placaDetector = new CascadeClassifier ("cascade_placav2.xml");
```

Al usar Hibernate para el mapeo de la base de datos a objetos propios del sistema se añadieron las siguientes librerías:

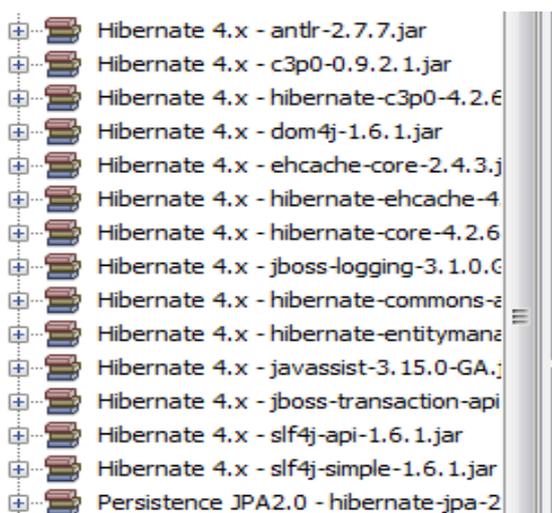


Figura 5.54 Librerías Hibernate

El método usado para el mapeo de las tablas, empezó creando una conexión a la base de datos dentro de los servicios del sistema, como se puede observar en la imagen a continuación:

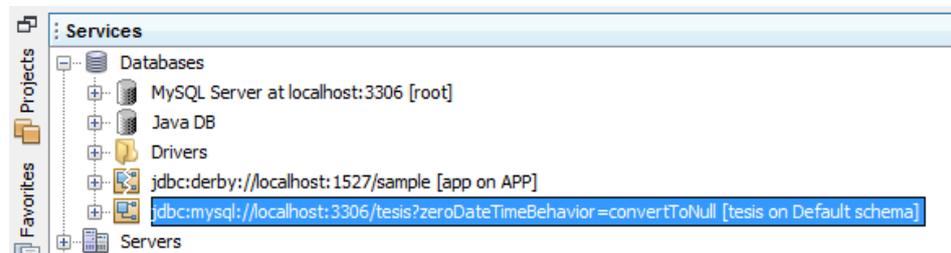


Figura 5.55 Cadena de conexión de la base de datos

El driver que se usó para esta conexión corresponde a MySQL (Connector/JDriver).

Una vez establecida la conexión, se realizó el proceso de mapeo de la siguiente forma:

- Primero se creó dentro del proyecto la opción “Hibernate Configuration Wizard” que es donde se definen todos los datos de la conexión:

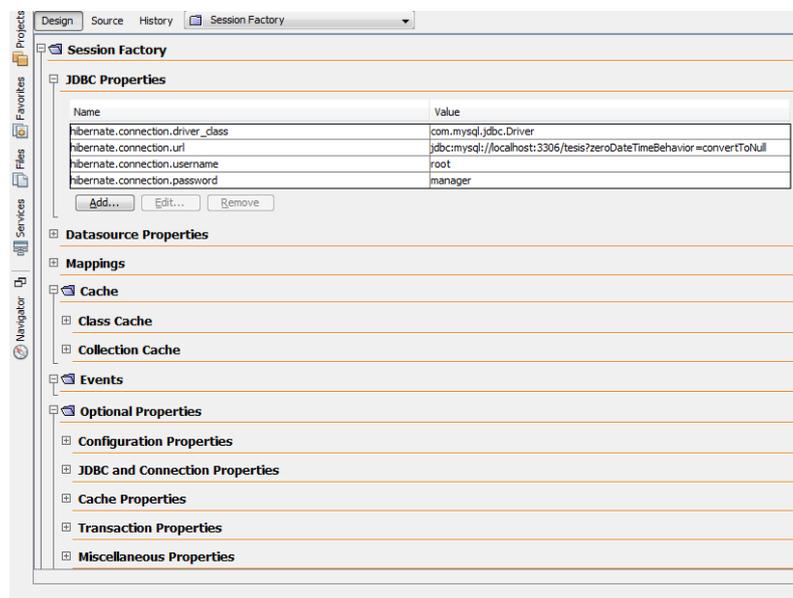
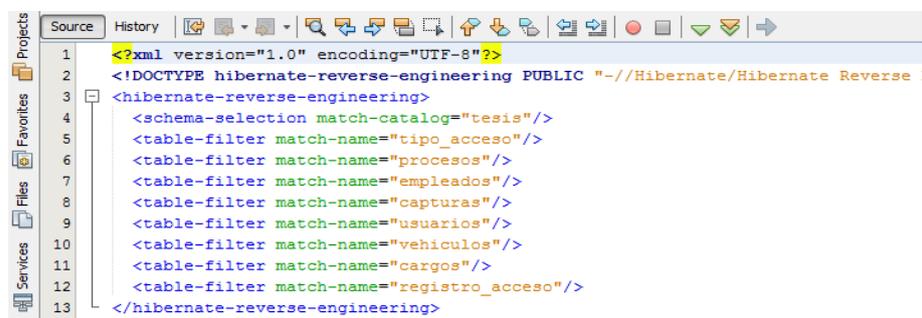


Figura 5.56 Resumen de configuración de la base de datos

- El siguiente paso consistió en seleccionar la opción “Hibernate Reverse Engineering Wizard...” que es donde se crearon todas las tablas que fueron

mapeadas, las mismas que se agregaron a un archivo de formato xml como se muestra en la imagen a continuación:



```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE hibernate-reverse-engineering PUBLIC "-//Hibernate/Hibernate Reverse Engineering DTD" "http://hibernate.sourceforge.net/hibernate-reverse-engineering-dtd.jar" [
3 <hibernate-reverse-engineering>
4 <schema-selection match-catalog="tesis"/>
5 <table-filter match-name="tipo_acceso"/>
6 <table-filter match-name="procesos"/>
7 <table-filter match-name="empleados"/>
8 <table-filter match-name="capturas"/>
9 <table-filter match-name="usuarios"/>
10 <table-filter match-name="vehiculos"/>
11 <table-filter match-name="cargos"/>
12 <table-filter match-name="registro_acceso"/>
13 </hibernate-reverse-engineering>

```

Figura 5.57 Xml de estructura de las tablas mapeadas por Hibernate

Otra herramienta que fue implementada dentro del sistema es la ya especificada API “Tess4J”, que permitió una forma más práctica de realizar la digitalización de caracteres y mejoró la accesibilidad de las funciones del OCR, estas librerías se muestran a continuación:

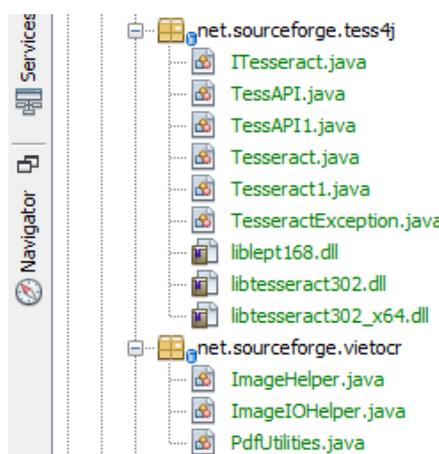


Figura 5.58 Estructura de librerías de Tess4J

Como se puede ver en la imagen 5.54 fue necesario agregar los “dlls” correspondientes a la librería OCR dentro del paquete de código, para este caso las librerías son: liblep168.dll, libtesseract302.dll y libtesseract302\_x64.dll.

Posteriormente se procederá a especificar el sistema con mayor detalle, desde el contenido del mismo hasta las funciones o métodos de procesamiento que se realizaron.

### 5.3.1.2.2. Creación del sistema

Para el proceso inicial de desarrollo se consideró la necesidad de dar niveles de acceso a los usuarios que ocuparán el sistema, por lo tanto creo un módulo de usuarios para el manejo de 2 roles específicos:

- **Supervisor:** Tendrá acceso a los módulos de detección de placas, consulta de empleados y consulta de registro de acceso.
- **Administrador:** Usuario con todos los privilegios, es decir podrá realizar procesos de mantenimientos de usuarios, empleados, vehículos y cargos, además tendrá acceso a los módulos del supervisor.



Figura 5.59 Login del sistema

Las contraseñas se encuentran cifradas, para lo cual se utilizó una librería externa denominada “Crypto”, que permite encriptar la contraseña para brindar una mayor seguridad el momento de validar el dato ingresado en el Login.

Luego del ingreso del usuario, se creó un menú de opciones del sistema, que dará acceso a todos los módulos desarrollados, dependiendo del rol del usuario conectado. Estas opciones se describirán a continuación:



Figura 5.60 Opciones del menú del sistema

- **Archivo:** Se contempló la opción de salir del sistema, incluyendo un atajo por teclado que responde a la combinación de teclas Alt+F4.
- **Accesos:** Dentro de este menú se incluyó la opción “Detección de Placa” que ejecuta el módulo de ingreso automatizado de vehículos.

- **Mantenimiento:** Esta opción será habilitada solo por el rol de administrador y permite realizar los procesos de: creación, modificación y eliminación de los registros de empleados, cargos, vehículo y usuarios en el sistema.
- **Consultas:** Esta opción contiene los registros de acceso. Adicionalmente se creó un módulo de consulta de empleados para comprobar cualquier información específica de un empleado.

### Formulario de detección de placa

Este formulario se desarrolló con todas las características mencionadas en los capítulos anteriores presentando la correcta integración de las herramientas de Open CV y Tesseract.



Figura 5.61 Interfaz detección de la placa

Inicialmente se contempló la realización de cada herramienta de visión artificial en 2 hilos, estos son procesos que se ejecutan paralelamente en el sistema para realizar tareas definidas, las cuales se describen a continuación:

**El primer hilo** está enfocado a realizar las operaciones de procesamiento interno del sistema, este inicia conjuntamente con el formulario de detección y

realiza la tarea de monitoreo constante de la placa del vehículo hasta que sea detectada. Todo esto lo ejecuta con los siguientes comandos:

```
new VideoCapture(0): // Inicia la captura de imágenes
Mat imagen = new Mat(); // almacena la imagen con el objeto detectado
```

Una vez detectada la placa se almacena una captura de la misma, para luego realizar internamente las funciones de recorte y binarización con los comandos integrados de Open CV:

```
Mat region = new Mat(imagen, rect); // Recorte de la imagen
Imgproc.threshold(gris, binario, 100, 255, Imgproc.THRESH_BINARY); //Binarización
```

El resultado final se muestra en la interfaz, en la sección de imagen de la placa.



Figura 5.62 Imagen de la placa binarizada

**En el segundo hilo** se encuentran todos aquellos procesos que realizan la detección OCR, la validación de caracteres y el almacenamiento de registros de acceso, todos ellos se obtienen a partir de la placa binarizada. Estos procesos se ejecutan secuencialmente y se depuran hasta llegar a un resultado final válido.

El proceso inicial consiste en la lectura de los caracteres de la imagen, cuyo resultado se obtiene al ejecutar el comando de detección de Tesseract.

```
String placa = Util.procesalimagen("binarizado.png"); // Ejecuta la digitalización de caracteres
```

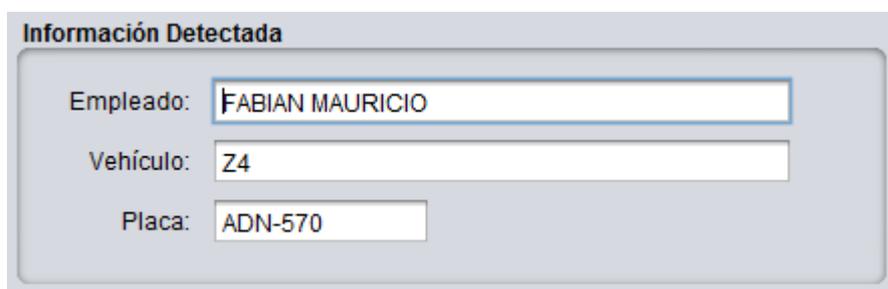
El resultado de la lectura de la placa es sometida a varias validaciones que identifican la integridad de los datos y determinan la estructura correcta de la placa. Estas validaciones consideran parámetros tales como: la eliminación de caracteres especiales, la identificación de la longitud y la determinación de la secuencia correcta.

Una vez que es considerada la detección como una placa válida, se realiza la comprobación de la misma sobre la base de datos, esta determina si la placa existe mediante una consulta HQL (hibernate query language) a la misma. La consulta se realizó en la siguiente función:

```
String hql = " select empleados.nombres from Vehiculos where placa = 'CCCCC'";
```

Luego de comprobada esta información el sistema responderá de 3 maneras diferentes identificadas por un semáforo:

- Si la placa detectada se encuentra dentro de los registros de la base de datos, se consultarán las tablas de empleado y vehículo para visualizar la información que se muestra a continuación:



Empleado:	FABIAN MAURICIO
Vehículo:	Z4
Placa:	ADN-570

Figura 5.63 Información de placa detectada

Este resultado actualiza la imagen del semáforo en verde y reproduce mediante el comando **clip.start()**, un sonido que indica el acceso positivo, la información se graba en la tabla de registro de acceso, además de almacenar la placa en la ubicación `imgplaca` con el código del registro y el tipo: binarizada y detectada.

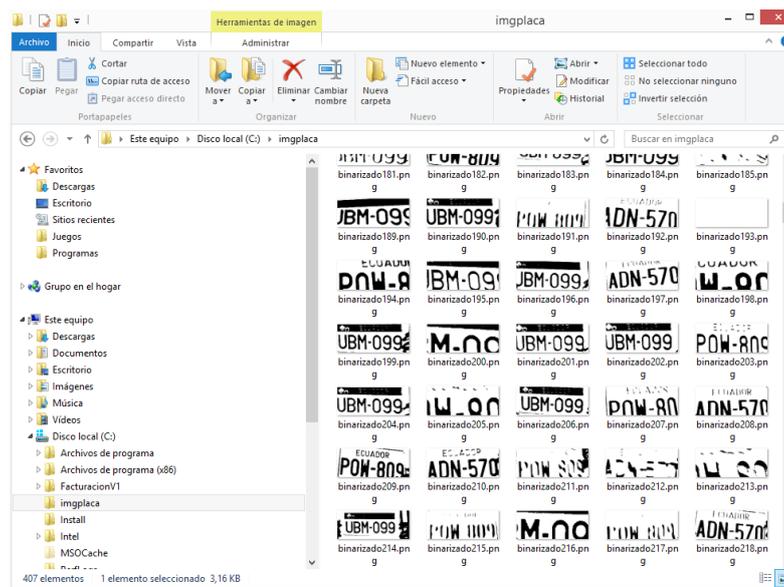


Figura 5.64 Ubicación de las imágenes almacenadas en la carpeta imgplaca.

Adicionalmente, se estableció un nivel de acceso opcional, que permite ingresar una placa no registrada o concede un ingreso eventual como invitado. El proceso solo se ejecuta cuando se ha realizado una lectura correcta de la placa y no se ha encontrado ningún registro con estos caracteres en la base de datos, mostrando el siguiente mensaje.

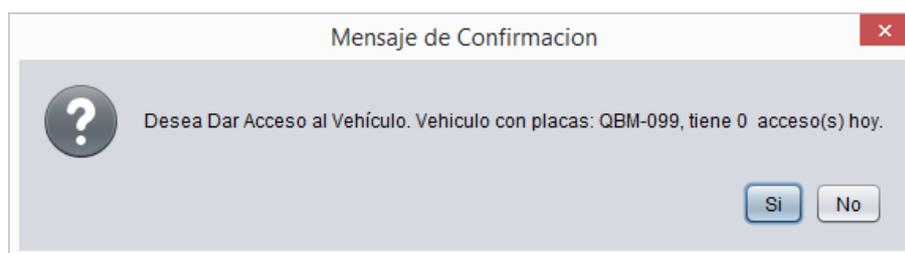
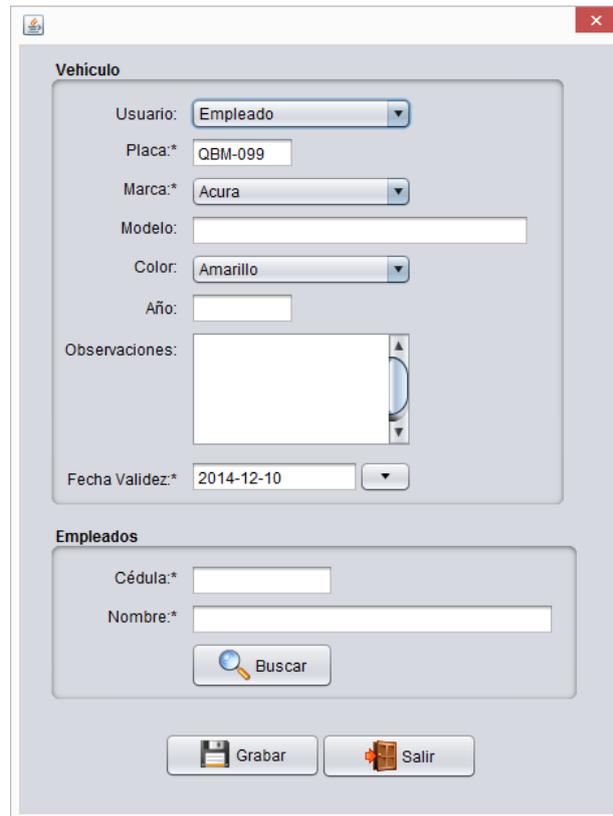


Figura 5.65 Mensaje de confirmación de acceso

Si el usuario presiona la opción “Sí”, se llama al formulario de registro, que permitirá el acceso temporal de invitado o la creación de un nuevo vehículo, vinculándolo a un empleado existente.



The image shows a software window titled "Vehiculo" with a close button in the top right corner. The window is divided into two main sections: "Vehiculo" and "Empleados".

**Vehiculo Section:**

- Usuario: A dropdown menu with "Empleado" selected.
- Placa\*: A text input field containing "QBM-099".
- Marca\*: A dropdown menu with "Acura" selected.
- Modelo: An empty text input field.
- Color: A dropdown menu with "Amarillo" selected.
- Año: An empty text input field.
- Observaciones: A large empty text area with a vertical scrollbar.
- Fecha Validez\*: A date picker showing "2014-12-10".

**Empleados Section:**

- Cédula\*: An empty text input field.
- Nombre\*: An empty text input field.
- Below the name field is a button with a magnifying glass icon and the text "Buscar".

At the bottom of the window, there are two buttons: "Grabar" (with a floppy disk icon) and "Salir" (with a door icon).

Figura 5.66 Formulario de ingreso de vehículo

Al momento de presionar la opción “No”, se almacena el registro con el tipo de acceso denegado, el proceso será diferenciado mostrando la imagen del semáforo en rojo, además se ejecutará otro sonido que indica esta incidencia.

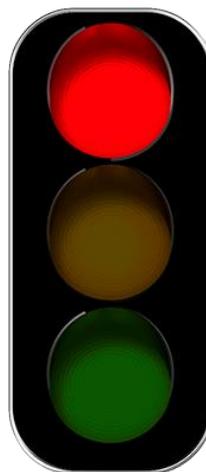


Figura 5.67 Semáforo en rojo representa un acceso denegado

Luego de registrada la incidencia, para todos los casos se consideró un tiempo de espera de 10 segundos en el hilo de detección ejecutado por la función **pause ()**, luego de ello, se muestra la imagen del semáforo en amarillo que activa la cámara en modo captura para la detección de una nueva placa.

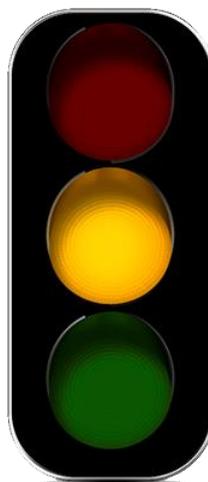


Figura 5.68 Semáforo en amarillo representa el sistema en espera de detección

Como último recurso ante situaciones especiales se desarrolló un botón de acceso manual que permite el ingreso inmediato del vehículo, este desencadena 2 acciones: almacena una captura inmediata de la cámara y graba un registro de acceso enlazado a un usuario especial denominado “invitado”.

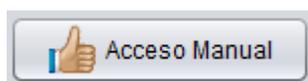


Figura 5.69 Botón de acceso manual

Adicionalmente se implementó un cuadro de información para cada proceso de desarrollo, esta sección muestra en que proceso se encuentra actualmente el hilo de procesamiento.

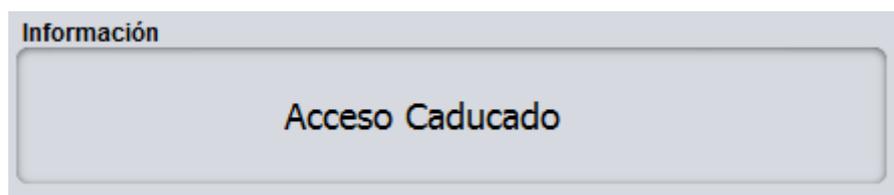


Figura 5.70 Cuadro de información del sistema

## Mantenimientos:

Para la sección de administración se tienen 3 mantenimientos principales, estos son necesarios para los registros que se manejan en el sistema, los cuales se describen a continuación:

- **Empleados:** Son los propietarios de los vehículos que usarán este sistema como mecanismo para ingresar al parqueadero de la institución.
- **Vehículos:** Estos pertenecen al empleado y contienen el patrón a ser reconocido por el sistema. Dentro de este es necesario aclarar que el dato primordial es la placa, sobre la cual se validará la digitalización del patrón reconocido.
- **Cargos:** Identifican la relación laboral que tienen los empleados con la institución responsable del parqueadero.
- **Usuarios:** Son todas aquellas personas que estarán encargadas de la supervisión y administración del sistema.

Para las opciones de mantenimiento se especificó un solo esquema de desarrollo que se centra en aplicar en una sola interfaz, las tareas de: creación, modificación, eliminación y consulta. Este procedimiento se desarrolló con el fin de simplificar el ingreso de información y disponer de todas las operaciones en una sola vista. Se ha tomado en cuenta los módulos de mantenimiento de: vehículos, empleados, cargos y usuarios.

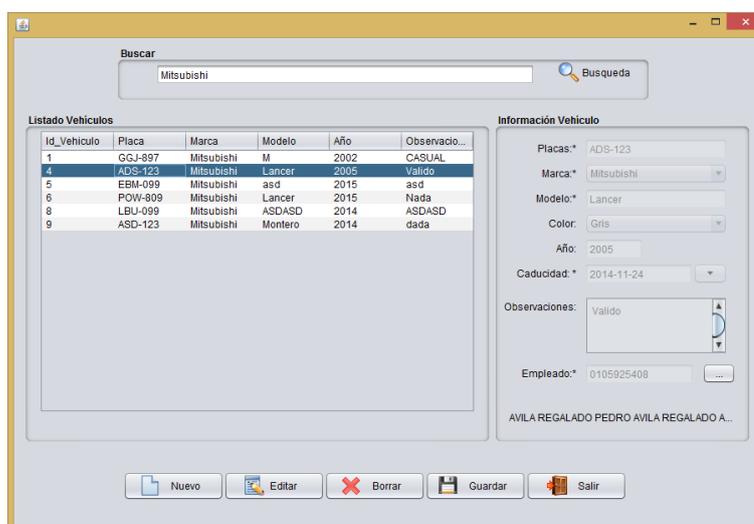


Figura 5.71 Ventana de mantenimiento de vehículos

Esta estructura está definida por un campo de búsqueda que realiza las operaciones de consulta en la base datos. Esta consulta es inmediata y se ejecuta cada vez que el usuario realiza una petición de datos mediante el evento keypress.

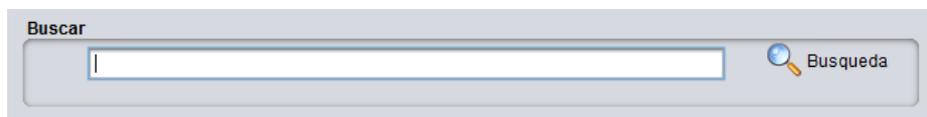


Figura 5.72 Campo de búsqueda del sistema

Este campo posee varios criterios de búsqueda de acuerdo al formulario que se esté manejando:

- **Empleados:** Se consulta la tabla empleados, filtrado por coincidencias en los campos cédula, apellido y nombre.
- **Vehículos:** Se consulta en la tabla vehículos por los términos de placa, marca y modelo.
- **Cargos:** Se consulta por la descripción del cargo.
- **Usuarios:** Se consulta por nombre de usuario.

Listado Vehiculos					
Id_Vehiculo	Placa	Marca	Modelo	Año	Observacio...
1	GGJ-897	Mitsubishi	M	2002	CASUAL
4	ADS-123	Mitsubishi	Lancer	2005	Valido
5	EBM-099	Mitsubishi	asd	2015	asd
6	POW-809	Mitsubishi	Lancer	2015	Nada
8	LBU-099	Mitsubishi	ASDASD	2014	ASDASD
9	ASD-123	Mitsubishi	Montero	2014	dada

Figura 5.73 Ejemplo de criterio de búsqueda

La sección del formulario de información presenta todos los datos que posee un determinado registro. Se consideró mantener los campos bloqueados hasta que se habiliten las opciones de ingreso o modificación, y así evitar problemas involuntarios de ingreso o eliminación de información.

Figura 5.74 Ventana de información para editar los registros

Para las opciones del formulario, se consideró mantener una sola estructura, especificada por una secuencia de botones con sus respectivos iconos. A continuación se especifica cada función implementada:

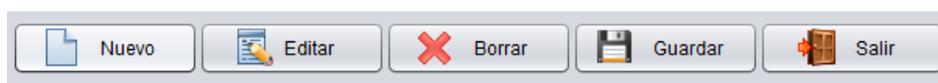


Figura 5.75 Botones que realizarán las acciones en el formulario

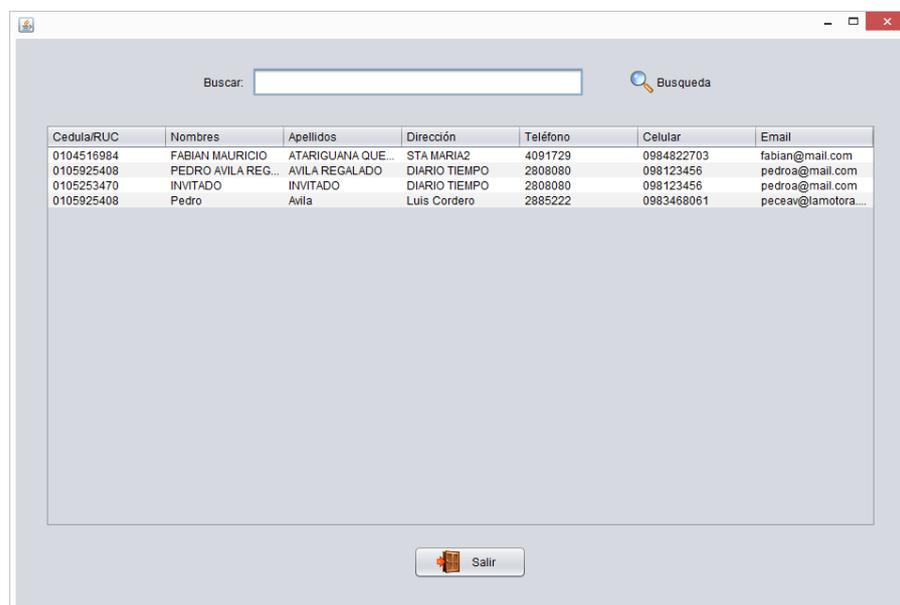
- El botón **nuevo** habilita los campos de textos para el ingreso de la información correspondiente.
- El botón **editar** permite modificar la información de un registro seleccionado.
- El botón **borrar** llama al proceso de eliminación de un registro seleccionado.
- El botón **guardar** es un botón relacionado con el proceso de almacenamiento, permitiendo la creación de un nuevo registro o la modificación de un registro existente. Adicionalmente, se desarrolló un proceso de validación para cada campo, que se ejecuta antes de realizar el proceso de guardado.

Los campos de la ventana de información pueden variar de un formulario a otro, ya que para cada módulo se tienen diferentes campos. En cambio los botones en el formulario se mantienen fijos ya que siempre se realizan las mismas acciones sobre los registros.

### Consultas:

Las consultas que posee el sistema son de acuerdo a la información más relevante que se maneja dentro del mismo, estas son las siguientes:

**Consulta de empleados:** Ejecuta la consulta HQL sobre la tabla empleados. El resultado visualiza los campos que se consideraron importantes en cada empleado.



Cedula/RUC	Nombres	Apellidos	Dirección	Teléfono	Celular	Email
0104516984	FABIAN MAURICIO	ATARIGUANA QUE...	STA MARIA2	4091729	0984822703	fabian@mail.com
0105925408	PEDRO AVILA REG...	AVILA REGALADO	DIARIO TIEMPO	2808080	098123456	pedro@mail.com
0105253470	INVITADO	INVITADO	DIARIO TIEMPO	2808080	098123456	pedro@mail.com
0105925408	Pedro	Avila	Luis Cordero	2885222	0983468061	peceav@lamotora...

Figura 5.76 Consulta de empleados

Al igual que lo implementado en los módulos de administración, se desarrolló una consulta instantánea realizada por el evento keypress sobre el campo de búsqueda.



Figura 5.77 Búsqueda de empleados

**Consulta de registros de acceso:** En esta consulta se presenta un listado de todas las incidencias que son registradas a través del módulo de reconocimiento de vehículos y que fueron almacenadas en la tabla registro\_acceso.

La estructura del módulo mantiene una búsqueda general y una búsqueda avanzada. La búsqueda general se realizó mediante una consulta anidada de los campos más importantes de la tabla registro\_acceso.



Figura 5.78 Búsqueda general de registros de acceso.

En cuanto a la búsqueda avanzada se especifican dos parámetros: por rango de fechas y por tipo de acceso:

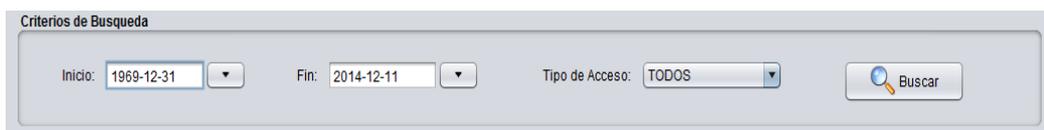


Figura 5.79 Búsqueda avanzada de registros de acceso.

- Por rango de fechas: En esta opción la búsqueda se realizó tomando en cuenta todos los accesos que fueron registrados, entre un límite de fechas determinado en los campos fecha inicio y fecha fin.
- Por tipo de acceso: Esta opción permite delimitar el listado por la tabla tipos\_accesos relacionada al registro de acceso (acceso automático, acceso denegado, acceso manual).

Adicionalmente, se tomó en consideración mostrar las imágenes capturadas de cada incidencia, la primera corresponde a la imagen de la captura del entorno, y la segunda corresponde a la binarización del patrón reconocido, tomadas de la ruta de almacenamiento especificada “imgplaca”.

Cruceros de Busqueda

Inicio: 1969-12-31 Fin: 2014-12-11 Tipo de Acceso: TODOS Buscar

Id	Fecha Ingreso	Hora Ingreso	Lectura Placa	Observacion	Tipo Acceso	Empleado
222	2014-12-11	07:44:00	ADN-570	NO SE PER...	DENEGADO	INVITADO IN...
220	2014-12-11	07:44:00	POW-809	ACCESO AU...	MANUAL	PEDRO AVL...
219	2014-12-11	07:43:00	OWN-809	NO SE PER...	DENEGADO	INVITADO IN...
218	2014-12-11	07:39:00	EDW-809	NO SE PER...	DENEGADO	INVITADO IN...
217	2014-12-11	07:38:00	PTV-172	INGRESO R...	AUTO	PEDRO AVL...
216	2014-12-11	07:38:00	FVU-800	INGRESO R...	AUTO	INVITADO IN...
215	2014-12-08	20:59:00	POW-809	ACCESO AU...	MANUAL	PEDRO AVL...
214	2014-12-08	20:59:00	ADN-570	INGRESO R...	AUTO	INVITADO IN...
213	2014-12-07	18:33:00	AFP-310	INGRESO R...	AUTO	JOHANNA N...
212	2014-12-07	18:32:00	AGB-434	NO SE PER...	DENEGADO	INVITADO IN...
211	2014-12-07	18:32:00	POW-809	ACCESO AU...	MANUAL	PEDRO AVL...
210	2014-12-07	18:31:00	ADN-570	INGRESO R...	AUTO	INVITADO IN...
209	2014-12-07	18:28:00	POW-809	ACCESO AU...	MANUAL	PEDRO AVL...
208	2014-12-04	07:26:00	ADN-570	NO SE PER...	DENEGADO	INVITADO IN...
207	2014-11-27	08:30:00	POW-809	ACCESO AU...	MANUAL	PEDRO AVL...
206	2014-11-27	08:30:00	AGB-434	NO SE PER...	DENEGADO	INVITADO IN...
205	2014-11-27	08:30:00	BUA-359	NO SE PER...	DENEGADO	INVITADO IN...
204	2014-11-27	08:25:00	AGB-434	PENDIENTE...	AUTO	INVITADO IN...
203	2014-11-27	08:23:00	LFP-310	PENDIENTE...	AUTO	INVITADO IN...
202	2014-11-27	08:22:00	POW-809	ACCESO AU...	MANUAL	PEDRO AVL...
201	2014-11-27	08:21:00	POW-809	PENDIENTE...	AUTO	INVITADO IN...
200	2014-11-27	00:02:00	POW-801	ACCESO AU...	MANUAL	PEDRO AVL...
199	2014-11-27	00:01:00	AFP-310	ACCESO AJ...	MANUAL	PEDRO AVL...
198	2014-11-27	00:00:00	ADN-570	ACCESO CA...	DENEGADO	INVITADO IN...
197	2014-11-27	00:00:00	AGB-434	PENDIENTE...	AUTO	INVITADO IN...
196	2014-11-25	00:21:00	POW-809	ACCESO AU...	MANUAL	PEDRO AVL...
195	2014-11-25	00:20:00	ISP-310	NO SE PER...	DENEGADO	INVITADO IN...
194	2014-11-25	00:20:00	POW-809	ACCESO AU...	MANUAL	PEDRO AVL...
193	2014-11-25	00:19:00	ADN-570	ACCESO MA...	MANUAL	INVITADO IN...
192	2014-11-24	22:27:00	ADN-570	PENDIENTE...	AUTO	INVITADO IN...
191	2014-11-24	22:27:00	AFP-310	ACCESO AU...	MANUAL	PEDRO AVL...

Vista Exterior



Vista Placa



Salir

Figura 5.80 Consulta de registros de acceso.

### 5.3.2. Desarrollo y monitoreo de progresos en el sistema

En toda la fase de construcción del sistema se dieron varios cambios, tanto en su interfaz como en su codificación. Se procederá a describir algunos de ellos indicando como influyeron en el desarrollo.

Inicialmente se aplicó especial énfasis en el proceso de detección de la placa, para ello el primer paso consistió en probar que todas las librerías de Open CV funcionarán correctamente en una interfaz de prueba con un archivo de entrenamiento de detección de rostros.

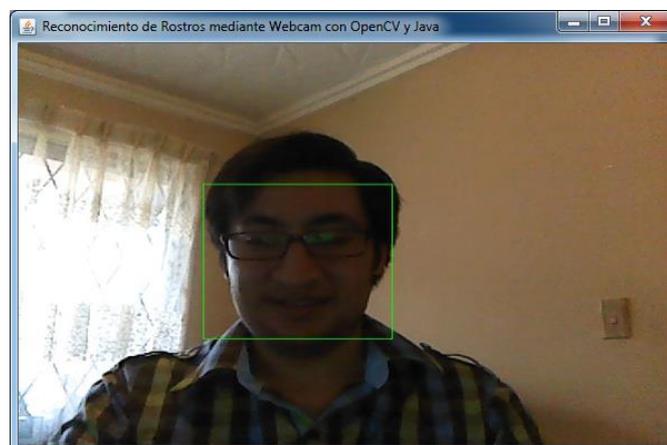


Figura 5.81 Prueba de detección de rostros.

Esta interfaz fue el estado inicial del sistema que se encargaría de detectar el patrón. Para continuar con el desarrollo de la interfaz fue necesario realizar el entrenamiento previo del objeto, en este caso el objeto correspondería a la placa vehicular.

En el proceso de entrenamiento se manejó varios criterios, realizando pruebas y ajustando configuraciones hasta obtener el mejor resultado para el proceso de detección, este proceso fue el más largo en el desarrollo ya que cada entrenamiento dura aproximadamente 5 días en el computador de entrenamiento designado. Con este entrenamiento final se procedió a realizar una interfaz que detecte la placa y muestre una captura de ella. Como se ha indicado anteriormente, Open CV mediante sus comandos de procesamiento de imágenes realiza las tareas de corte y binarización, por ello, se implementó a manera de prueba los tres estados que realizará el sistema con la placa.

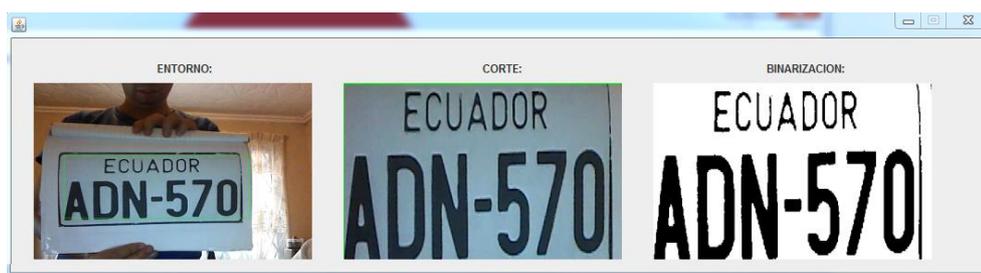


Figura 5.82 Interfaz de pruebas de detección.

Esta implementación concluyó con la de detección de patrones obteniendo la placa deseada y la imagen procesada de la manera más óptima. A partir de esta imagen, se realizó el proceso de OCR que consistió en ejecutar la librería tess4j para la detección de caracteres con un lenguaje previamente entrenado.

El procedimiento de entrenamiento de Tesseract, consistió en ajustar las imágenes de los caracteres para comprobar que se detecte correctamente las letras de la placa. Una vez con el lenguaje correctamente creado, se implementó en la interfaz un campo que devolvía la lectura de la placa. Este resultado no fue muy óptimo ya que debido a múltiples imperfecciones y letras no relacionadas a la placa, no permitieron una correcta comprobación, por ello, se implementó múltiples validaciones para depurar estos errores, tanto a nivel de OCR como a nivel de programación. Estos resultados se implementaron en 3 campos:

- **OCR:** Es el resultado de la digitalización de los caracteres en la imagen binarizada.
- **Depurado:** Es la eliminación de caracteres especiales (/ , % , & , etc) a través de comandos de Tesseract.
- **Validado:** Son las validaciones realizadas en el código fuente, depurando los caracteres no necesarios y la longitud de la cadena, y así obtener una aproximación al formato establecido de la placa.

OCR:	ECUADº/F)- *ADN-570
DEPURADO:	ECUADCF T- ADN-570
VALIDADO:	ADN-570

Figura 5.83 Sección de pruebas de OCR

Una vez integradas las herramientas, se realizó como proceso final la comparación de la cadena resultante con las placas registradas en la base de datos, de tal manera que se pueda conocer el empleado. En el apartado “Detalle de Registro” se indicaron tres campos, dentro de los cuales se procedió a visualizar: el id del vehículo, el nombre del empleado propietario de la placa reconocida y la marca del automotor.

Figura 5.84 Interfaz de integración con la base de datos.

Posteriormente, se realizaron todas aquellas validaciones para adaptar el sistema a las diferentes situaciones del parqueadero como por ejemplo: accesos no autorizados, ingresos eventuales, validaciones por fecha de caducidad, etc.

Como último procedimiento, se modificó la interfaz del sistema para orientarlo a las necesidades del usuario final, es decir, simplificar la interfaz, añadir un indicador de acceso (semáforo) y un indicador de estado (Panel de información).



Figura 5.85 Interfaz final de detección de placas.

Otro cambio importante realizado en el sistema fue el de registros de acceso. Inicialmente se consideró la consulta de los registros parametrizados solamente por rango de fechas.

Inicio:	Fin:	Fecha Ingreso	Hora Ingreso	lectura Placa	Observacion	Tipo Acceso
1900-01-01	2999-12-31	2014-11-15	22:03:00		Acceso Manual	MANUAL
		2014-11-15	22:04:00	ASD-1234	Acceso Manual	MANUAL
		2014-11-15	22:09:00		Acceso Manual	MANUAL
		2014-11-15	22:11:00		Acceso Manual	MANUAL
		2014-11-15	22:12:00		Acceso Manual	MANUAL
		2014-11-15	22:14:00		Acceso Manual	MANUAL
		2014-11-15	22:16:00		Acceso Manual	MANUAL
		2014-11-15	23:32:00	AFP-310	VEHICULO DETECTADO	MANUAL
		2014-11-15	23:35:00	AFP-310	VEHICULO DETECTADO	MANUAL
		2014-11-16	00:47:00		ACCESO MANUAL	MANUAL
		2014-11-16	00:49:00		ACCESO MANUAL	MANUAL
		2014-11-16	00:56:00		ACCESO MANUAL	MANUAL
		2014-11-16	01:00:00		ACCESO MANUAL	MANUAL
		2014-11-16	01:23:00		ACCESO MANUAL	MANUAL
		2014-11-16	01:28:00		ACCESO MANUAL	MANUAL
		2014-11-16	01:41:00		ACCESO MANUAL	MANUAL
		2014-11-16	03:05:00		ACCESO MANUAL	MANUAL
		2014-11-16	03:17:00	AFP-310	VEHICULO DETECTADO	MANUAL
		2014-11-16	03:20:00	AFP-310	VEHICULO DETECTADO	MANUAL
		2014-11-16	03:24:00	AGB-254	ACCESO MANUAL	MANUAL
		2014-11-16	03:24:00	AGB-254	ACCESO MANUAL	MANUAL
		2014-11-16	03:26:00		ACCESO MANUAL	MANUAL
		2014-11-16	03:26:00		ACCESO MANUAL	MANUAL
		2014-11-16	03:26:00		ACCESO MANUAL	MANUAL
		2014-11-16	03:26:00		ACCESO MANUAL	MANUAL
		2014-11-16	03:26:00		ACCESO MANUAL	MANUAL
		2014-11-16	03:26:00		ACCESO MANUAL	MANUAL

Figura 5.86 Interfaz de registro de accesos

Luego se consideró necesario implementar imágenes para respaldar los accesos, por ello, se agregó:

- La vista exterior del acceso.
- La captura binarizada de la placa.
- Un filtro de tipo de acceso.
- Y una búsqueda generalizada de contenido de registro.

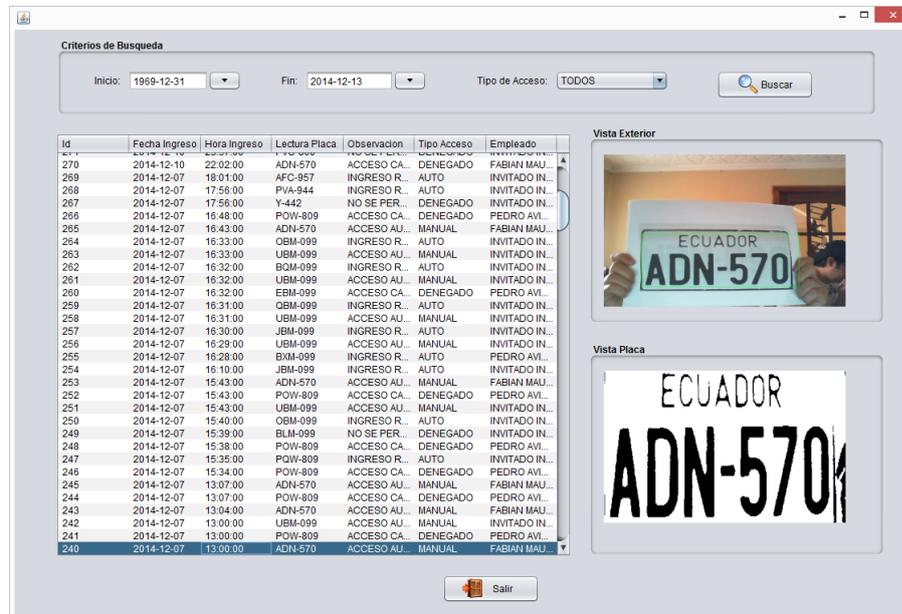


Figura 5.87 Interfaz final de registro de accesos

En cuanto a los mantenimientos, se simplificó el manejo de los mismos en una sola interfaz, debido a que anteriormente se manejaban ventanas exteriores, generando muchos pasos para realizar una sola actividad.

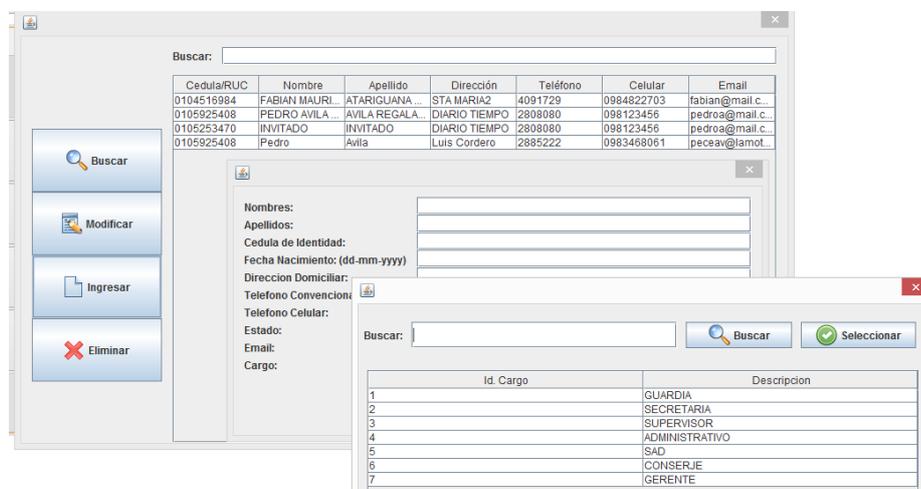


Figura 5.88 Interfaz de mantenimiento de empleados

Además de realizar un cambio en la funcionalidad, también se realizó un cambio de interfaz, con el objetivo de mantener todos los formularios con una misma estructura.

The interface is titled 'Registros Empleados' and includes a search bar at the top. Below the search bar is a table with the following data:

Cedula/RUC	Nombre	Apellido	Dirección	Teléfono	Celular	Email
0104516984	FABIAN MAU...	ATARIGUANA...	STA MARIA2	4091729	0984822703	fabian@mail...
0105925408	PEDRO AVIL...	AVILA REGAL...	DIARIO TIEM...	2808080	098123456	pedroa@mail...
0105253470	INVITADO	INVITADO	DIARIO TIEM...	2808080	098123456	pedroa@mail...
0105925408	Pedro	Avila	Luis Cordero	2885222	0983468061	peceav@lam...

To the right of the table is a form titled 'Informacion Empleado' with the following fields:

- Nombres: EGALADO AVILA REGALADO
- Apellidos: AVILA REGALADO
- Cédula: 0105925408
- Fecha de: 2015-01-10
- Nacimiento: [Empty]
- Dirección: DIARIO TIEMPO
- Teléfono: 2808080
- Celular: 098123456
- Email: pedroa@mail.com
- Estado: 1
- Cargo: SUPERVISOR

At the bottom of the interface are five buttons: 'Nuevo', 'Editar', 'Guardar', 'Borrar', and 'Salir'.

Figura 5.89 Interfaz final de mantenimiento de empleados

### 5.3.3. Integración de resultados

Dentro de este punto se dan a conocer los métodos de tratamiento de los elementos que son necesarios para el funcionamiento del sistema.

El primer elemento corresponde al resultado de aplicar el entrenamiento en OpenCV que consiste en un archivo en formato XML, el cual permitirá identificar el patrón que deberá encontrar el sistema dentro del entorno capturado. Este XML contiene los nodos de entrenamiento, que son las coordenadas de los puntos más relevantes del objeto.

```

1  <?xml version="1.0"?>
2  <opencv_storage>
3  <cascade>
4  <stageType>BOOST</stageType>
5  <featureType>HAAR</featureType>
6  <height>24</height>
7  <width>60</width>
8  <stageParams>
9  <boostType>GAB</boostType>
10 <minHitRate>9.9900001287460327e-001</minHitRate>
11 <maxFalseAlarm>5.000000000000000e-001</maxFalseAlarm>
12 <weightTrimRate>9.499999999999999e-001</weightTrimRate>
13 <maxDepth>1</maxDepth>
14 <maxWeakCount>100</maxWeakCount></stageParams>
15 <featureParams>
16 <maxCatCount>0</maxCatCount>
17 <featSize>1</featSize>
18 <mode>ALL</mode></featureParams>
19 <stageNum>19</stageNum>

```

Figura 5.90 Cascade XML de detección de placas

Al integrar OpenCV con el sistema, se ejecuta el comando CascadeClassifier que se encarga del reconocimiento del patrón según el XML especificado.

```
CascadeClassifier placaDetector = new CascadeClassifier("cascade_placasv2.xml");
```

La integración de este Cascade, permite al sistema reconocer y procesar el patrón detectado, para lo cual ejecuta tres pasos:

- Detección y captura del patrón en el entorno.
- Recorte de la placa.
- Binarización del recorte.

El resultado del OCR se define como un idioma, y corresponde a un archivo de extensión “traineddata” el mismo que al integrarse con el sistema, permite el reconocimiento de caracteres.

```

13 1 8 0,255,0,255,0,32767,0,32767,0,32767 NULL 11 0 0 # # 1 [31 ]0
14 B S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # B [42 ]A
15 U S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # U [55 ]A
16 X S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # X [58 ]A
17 3 8 0,255,0,255,0,32767,0,32767,0,32767 NULL 15 0 0 # # 3 [33 ]0
18 8 8 0,255,0,255,0,32767,0,32767,0,32767 NULL 16 0 0 # # 8 [38 ]0
19 5 8 0,255,0,255,0,32767,0,32767,0,32767 NULL 17 0 0 # # 5 [35 ]0
20 N S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # N [4e ]A
21 R S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # R [52 ]A
22 S S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # S [53 ]A
23 A S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # A [41 ]A
24 L S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # L [4c ]A
25 T S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # T [54 ]A
26 C S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # C [43 ]A
27 E S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # E [45 ]A
28 Z S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # Z [5a ]A
29 J S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # J [4a ]A
30 0 8 0,255,0,255,0,32767,0,32767,0,32767 NULL 28 0 0 # # 0 [30 ]0
31 Y S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # Y [59 ]A
32 7 8 0,255,0,255,0,32767,0,32767,0,32767 NULL 30 0 0 # # 7 [37 ]0
33 V S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # V [56 ]A
34 M S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # M [4d ]A
35 K S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # K [4b ]A
36 G S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # G [47 ]A
37 W S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # W [57 ]A
38 H S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # H [48 ]A
39 F S 0,255,0,255,0,32767,0,32767,0,32767 NULL -1 0 0 # # F [46 ]A

```

Figura 5.91 Idioma de entrenamiento OCR

El archivo traineddata, que contiene el idioma entrenado, es procesado por la librería Tess4j para interpretar los caracteres que posee una imagen. El sistema basándose en este entrenamiento, realiza las funciones de comparación para determinar si el carácter leído corresponde al idioma entrenado. Los caracteres validos se almacenan en el sistema.

El sistema toma estos resultados y los compara con la base de datos, para determinar la validez del acceso, la misma que será reflejada para cada caso en la base de datos.

## Conclusión

La base del proceso de desarrollo se enfocó en establecer una correcta detección, por ello, se realizó varios intentos de entrenamiento con el fin de determinar el mejor resultado posible, esto se realizó: aumentando el número de muestras, editando el objeto de entrenamiento, y aplicando los parámetros más adecuados (número de muestras, dimensiones de las muestras, número de nodos, etc).

Con las herramientas seleccionadas y configuradas, se procedió a realizar el sistema. El proceso de desarrollo se basó en establecer la mejor integración entre sus librerías, todo ello con el fin de obtener la validación de los resultados individuales y la depuración de los mismos para cumplir con los objetivos planteados en el análisis.

La estructura de la programación se desarrolló en hilos, con el fin de que los procesos automáticos del sistema se ejecuten sin interrupciones, hasta detectar el objeto y validarlo. El proceso de detección se estableció en cadena, en la que se inicia por el reconocimiento y procesamiento del objeto entregado por Open CV y finaliza con los resultados interpretados por Tesseract en un conjunto de caracteres. Estos caracteres no representaron el resultado final, ya que necesitaban un proceso de validación que ayudaría a pulir los resultados para compararlos con la base de datos.

Adicionalmente, se vio la necesidad de adecuar el sistema para el usuario final de la manera más simple y amigable, diseñando la interfaz con un mismo esquema y simplificando el proceso de manejo en actividades de registro, mantenimiento y consulta.

## Capítulo 6

### Pruebas y simulaciones

#### Introducción

Luego de concluir el proceso de desarrollo del sistema, es necesario poner a prueba la capacidad del mismo para cumplir los objetivos propuestos. Se realizará detecciones de diferentes placas en ambientes cambiantes para determinar la capacidad de respuesta del sistema, a su vez se pretende verificar la funcionalidad del mismo y como éste influirá en el usuario final.

#### 6.1. Pruebas de detección

Las pruebas de detección se enfocan en determinar la capacidad de reconocimiento del sistema, es decir, hasta qué condiciones detecta la placa vehicular y en qué estado debe estar la placa para su correcta detección.

#### Prueba 1: Placa en buen estado

**Condiciones:** La primera prueba se realiza con una placa en el mejor estado, es decir, con sus dígitos claros y con menores imperfecciones.



Figura 6.92 Placa en buen estado

**Captura:**

Figura 6.93 Placa en buen estado binarizada.

**Resultados:**

Información Detectada	
Empleado:	PEDRO AVILA REGALADO AVILA REGALADO
Vehículo:	ASD
Placa:	POW-809

Figura 6.94 Resultado de la placa en buen estado binarizada.

**Observaciones:** La placa obtuvo una detección correcta e instantánea sin ningún inconveniente.

**Prueba 2: Placa en estado regular**

**Condiciones:** Prueba con placa en estado regular con imperfecciones en las letras, doblada e inclinada ligeramente a la derecha.



Figura 6.95 Placa en estado regular.

**Captura:**



Figura 6.96 Placa en estado regular binarizada.

**Resultados:**

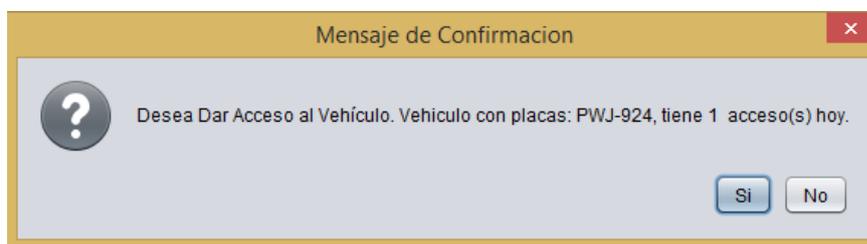


Figura 6.97 Resultados de la placa en estado regular.

**Observaciones:** Se obtuvo un reconocimiento correcto, con la ligera diferencia en el tiempo de detección aumento ligeramente comparado a la prueba 1.

### Prueba 3: Placa en mal estado

**Condiciones:** Prueba con placa en mal estado, con las letras cortadas por las imperfecciones que presentan.



Figura 6.98 Placa en mal estado.

### Captura:

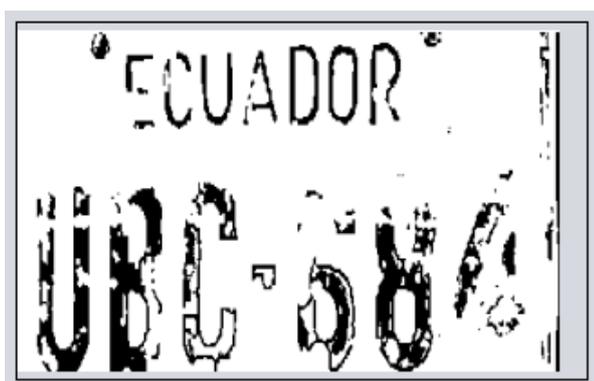


Figura 6.99 Placa en mal estado binarizada.

### Resultados:

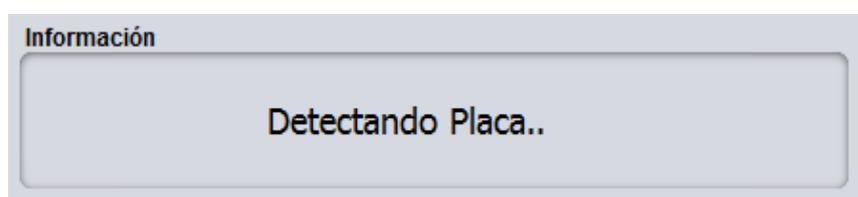


Figura 6.100 Resultados de la placa en mal estado.

### Observaciones:

El patrón fue reconocido sin inconvenientes, sin embargo, al momento de evaluarlo con OCR el resultado es nulo, por lo cual se quedará en proceso de espera hasta encontrar una placa válida.

**Prueba 4: Ambientes oscuros**

**Condiciones:** Detección de placa en ambientes con poca iluminación y espacios cerrados carentes de luz.

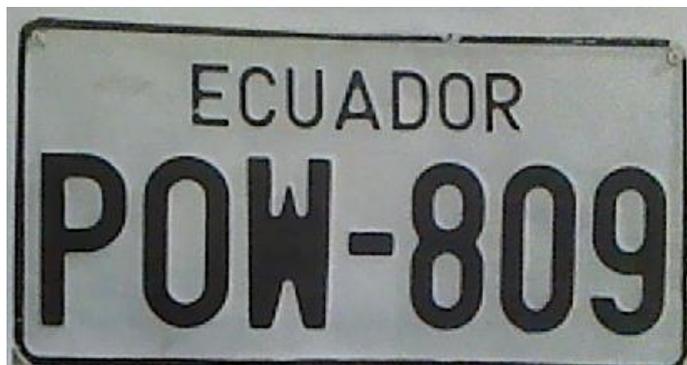


Figura 6.101 Placa en un ambiente oscuro.

**Captura:**

Figura 6.102 Placa binarizada en un ambiente oscuro.

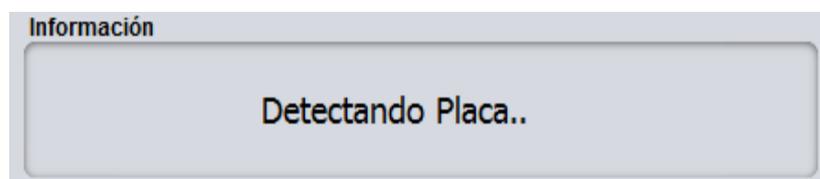
**Resultados:**

Figura 6.103 Resultados de la placa en un ambiente oscuro.

**Observaciones:** Cuando no existe una buena iluminación, la placa es reconocida pero no leída, y se mantiene en el estado de detección como se muestra en la imagen 6.97 “Detectando Placa...”.

#### Prueba 4: Ambientes Claros o Iluminados

**Condiciones:** Corresponde a situaciones en ambientes claros o zonas que presentan sistemas de iluminación.



Figura 6.104 Placa en ambientes claros.

#### Captura:

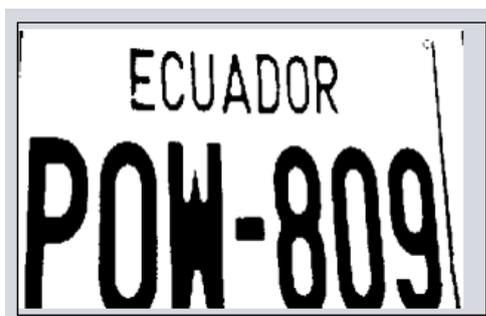


Figura 6.105 Placa en ambientes claros binarizada.

#### Resultados:

Información Detectada	
Empleado:	PEDRO AVILA REGALADO
Vehículo:	ASD
Placa:	POW-809

Figura 6.106 Resultados de la placa en ambientes claros.

**Observaciones:** El reconocimiento se realizó exitosamente, además la información se presentó rápida y sin inconvenientes.

**Prueba 5: Placas con diferente tipografía**

**Condiciones:** Se realiza la prueba de una placa de diferente tipografía.

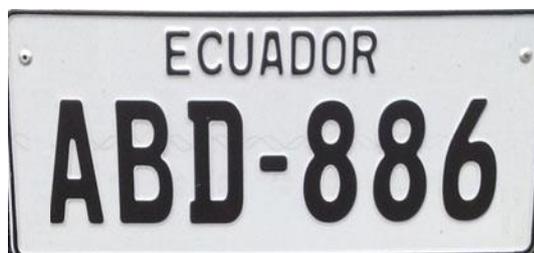


Figura 6.107 Placa con diferente tipografía.

**Captura:**

Figura 6.108 Placa binarizada con diferente tipografía.

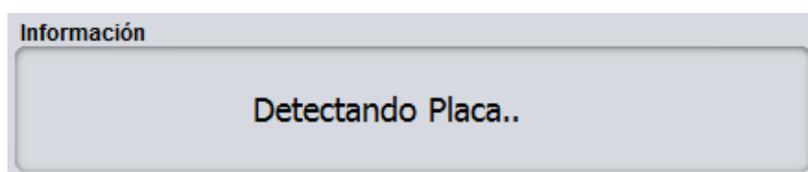
**Resultados:**

Figura 6.109 Resultado de placa con diferente tipografía.

**Observaciones:** El patrón detectado identifica la placa, pero debido a la diferente tipografía de sus caracteres, no corresponden al lenguaje previamente entrenado.

## 6.2. Pruebas de rendimiento

Las pruebas de rendimiento estarán enfocadas en establecer la calidad de las imágenes de capturas, la distancia de detección y la velocidad de ejecución del sistema.

### Prueba 1: Cámara VGA, distancia corta

**Condiciones:** Esta prueba se realizó a una corta distancia promedio de 30 centímetros con relación a la webcam.

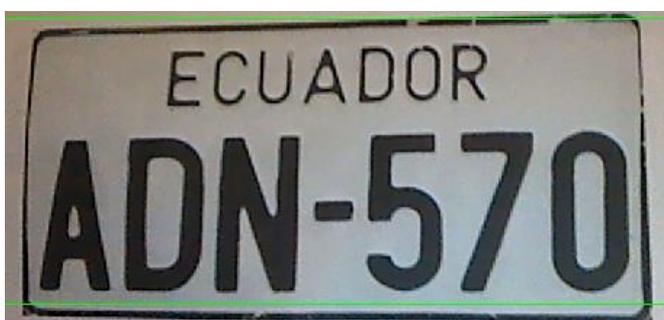


Figura 6.110 Captura con cámara VGA, distancia corta.

### Captura:



Figura 6.111 Binarización con cámara VGA, distancia corta.

### Resultados:

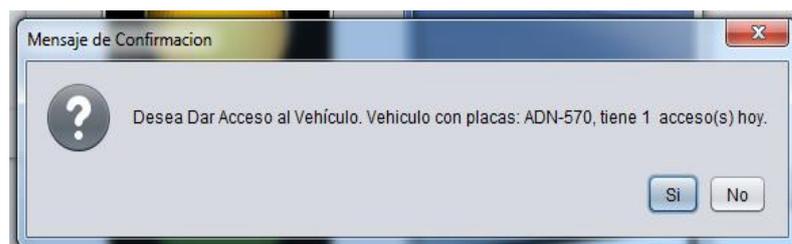


Figura 6.112 Resultado con cámara VGA distancia corta.

**Observaciones:** La detección es correcta, a pesar de que la calidad de la captura no es muy alta, pero ello no influye en el resultado final.

### Prueba 2: Cámara VGA, larga distancia

**Condiciones:** Esta prueba se realizará a una distancia de 1 metro de la cámara web VGA.

**Captura:**



Figura 6.113 Captura con cámara VGA, larga distancia.

**Resultados:**



Figura 6.114 Resultados con cámara VGA, larga distancia.

**Observaciones:** La detección a una distancia de 1 metro en adelante se complica, ya que la imagen capturada por la cámara se encuentra demasiado pixelada para ser reconocida por el OCR.

**Prueba 4: Cámara HD, distancia corta.**

**Condiciones:** Prueba realizada a 30 cm de distancia con la cámara Microsoft hd-300.

**Captura:**



Figura 6.115 Captura con cámara HD, distancia corta.

**Resultado:**

Información Detectada	
Empleado:	<input type="text"/>
Vehículo:	<input type="text"/>
Placa:	<input type="text" value="AFE-059"/>

Figura 6.116 Resultado con cámara HD, distancia corta.

**Observaciones:** El resultado fue correcto e inmediato. La diferencia con la cámara VGA fue la resolución de la imagen que en este caso tuvo mayor nitidez.

### Prueba 5: Cámara HD, distancia media

**Condiciones:** Placa probada a una distancia de 2 metros con la cámara Microsoft hd-300.

**Captura:**



Figura 6.117 Captura de cámara HD, distancia media.

**Resultados:**

Información Detectada	
Empleado:	PEDRO AVILA REGALADO AVILA REGALADO
Vehículo:	ASD
Placa:	POW-809

Figura 6.118 Resultado de cámara HD, distancia media.

**Observaciones:** A pesar que la imagen binarizada se encontraba pixelada, la placa obtuvo una detección correcta, pero su tiempo de respuesta fue aproximadamente en 5 segundos, esto debido a que es necesario que la placa se ubique correctamente para captar la imagen.

**Prueba 5: Cámara HD, distancias mayores a 2 metros**

**Condiciones:** Prueba realizada a 2.5 metros de la cámara HD.

**Captura:**

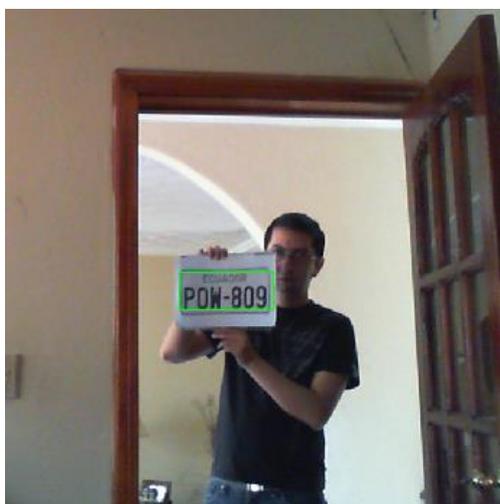


Figura 6.119 Captura de cámara HD, distancia larga.

**Binarización:**



Figura 6.120 Binarización de cámara HD, distancia larga.

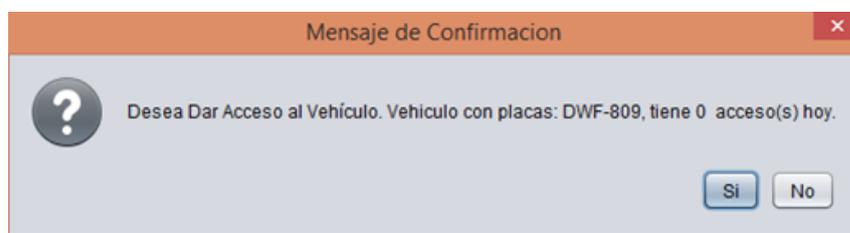


Figura 6.121 Resultados de cámara HD, distancia larga.

**Observaciones:** La detección no fue correcta ya que la imagen fue demasiado pequeña y la tipografía pixelada, lo que llevó a reconocer otro texto.

### Prueba 6: Prueba de inicialización en un PC de gama alta

**Condiciones:** Prueba en velocidad en el arranque del sistema en una computadora i7 16gb de RAM.

**Captura:**

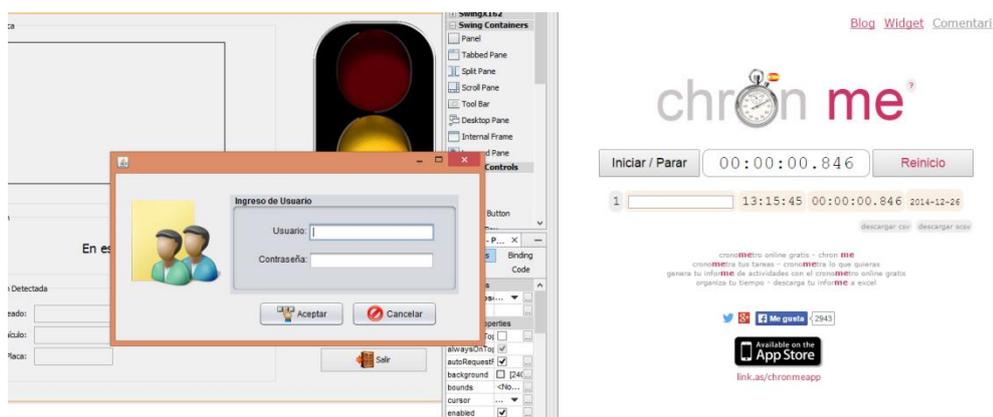


Figura 6.122 Tiempo de inicialización del sistema, pc de gama alta.

**Observaciones:** La respuesta fue inmediata en menos de un segundo.

### Prueba 7: Prueba de detección PC de gama alta

**Condiciones:** Prueba de velocidad de detección en una computadora i7 16gb de RAM.

## Captura:



Figura 6.123 Tiempo de detección del sistema, pc de gama alta.

**Observaciones:** La detección de la placa, desde que esta tomo contacto con la cámara fue 2.5 segundos.

## Prueba 8: Prueba de consulta en un PC de gama alta

**Condiciones:** Prueba de velocidad de consulta con 300 registros en una computadora i7 16gb de RAM.

## Captura:

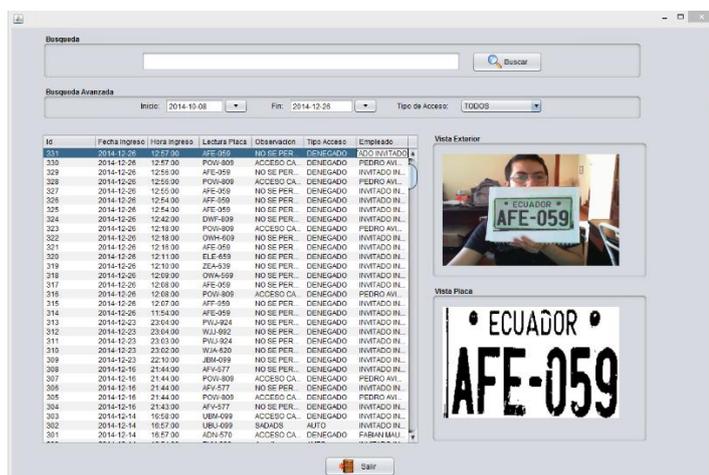


Figura 6.124 Tiempo de consulta del sistema, pc de gama alta.

**Observaciones:** El tiempo de consulta para el número de registros fue muy bajo con un tiempo de menos de un segundo.

### Prueba 9: Prueba de inicialización en un PC de gama media

**Condiciones:** El análisis de tiempos se realizó en un computador de características moderadas.

#### Captura:

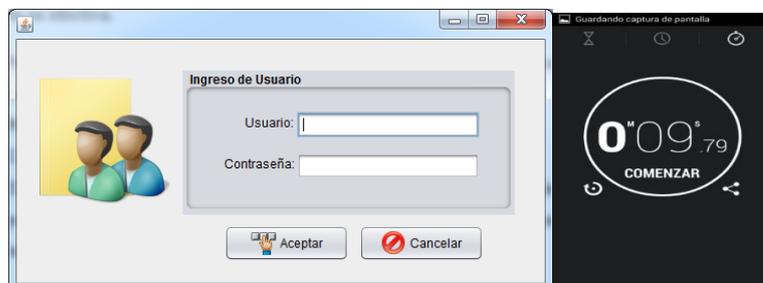


Figura 6.125 Tiempo de inicialización del sistema, pc de gama media.

**Observaciones:** El inicio de la aplicación tomo un tiempo relativamente corto y no presenta ningún factor negativo para el usuario.

### Prueba 10: Prueba de detección en un PC de gama media

**Condiciones:** El proceso para esta prueba se lo realizó teniendo en cuenta las consideraciones básicas para la detección, como un ambiente con iluminación normal.

#### Captura:

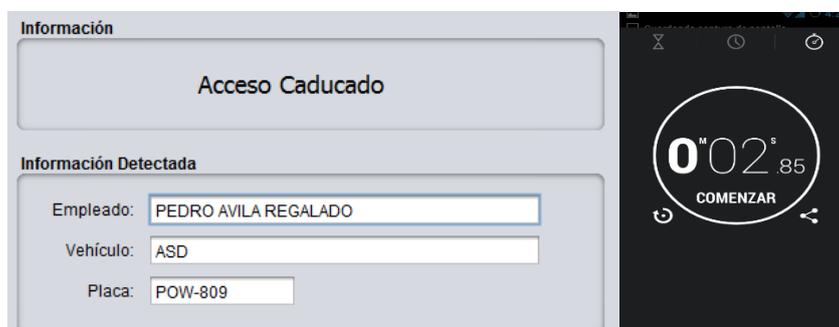


Figura 6.126 Tiempo de detección del sistema, pc de gama media.

**Observaciones:** El reconocimiento del vehículo se realizó en un tiempo corto y similar al resultado del pc de alta gama.

### Prueba 11: Prueba de consulta en un PC de gama media

**Condiciones:** Con un total de 239 registros de detecciones o incidencias en el módulo de reconocimiento se procedió a realizar la prueba.

#### Captura:

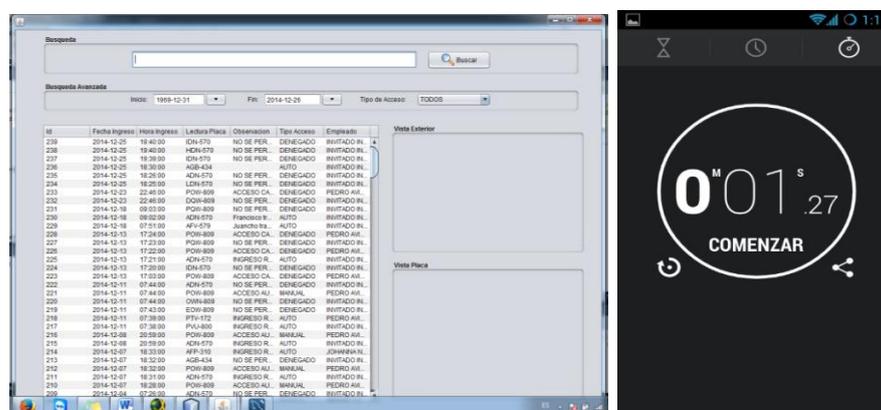


Figura 6.127 Tiempo de consulta del sistema, pc de gama media.

**Observaciones:** Los resultados son visualizados inmediatamente sin presentar ningún inconveniente para el usuario.

### 6.3. Pruebas de interfaz

Estas pruebas se encargan de evaluar, tanto el diseño como la funcionalidad de las diferentes interfaces del sistema.

#### Prueba 1: Interfaz de detección de un acceso correcto

**Condiciones:** Prueba del módulo de detección cuando un vehículo es reconocido correctamente para su acceso.

**Captura:**

Figura 6.128 Interfaz de acceso correcto.

**Observaciones:** En un acceso permitido, se presenta el semáforo en verde con un sonido agudo indicando que se realizó el ingreso. La información del vehículo se muestra correctamente de una manera intuitiva para avisar al supervisor el acceso.

**Prueba 2: Interfaz de detección placa no registrada**

**Condiciones:** Prueba cuando un vehículo es detectado pero no se encuentra registrado en el sistema.

**Captura:**

Figura 6.129 Interfaz de placa no registrada.

**Observaciones:** La interfaz de detección muestra el semáforo en amarillo mientras se visualiza el mensaje de ingreso o denegación. El sistema se conserva en pausa mientras se ingrese la información del registro del vehículo, luego permite su ingreso. Las alternativas y la manera de presentarlas son adecuadas para el supervisor y lo obliga a seguir los pasos respectivos para el acceso, además de facilitar ingresos eventuales.

### Prueba 3: Interfaz de detección de acceso denegado

**Condiciones:** Prueba de detección para placas denegadas por varios motivos.

**Captura:**



Figura 6.130 Interfaz de acceso denegado.

**Observaciones:** La interfaz muestra el semáforo en rojo con un sonido grave que indica que el acceso no fue permitido. El cuadro de información muestra el motivo de esta restricción para mantener informado en todo tiempo al supervisor.

## Prueba 4: Interfaz de mantenimiento

**Condiciones:** Prueba de interfaz de mantenimientos de: vehículos, empleados usuarios y cargos.

**Captura:**

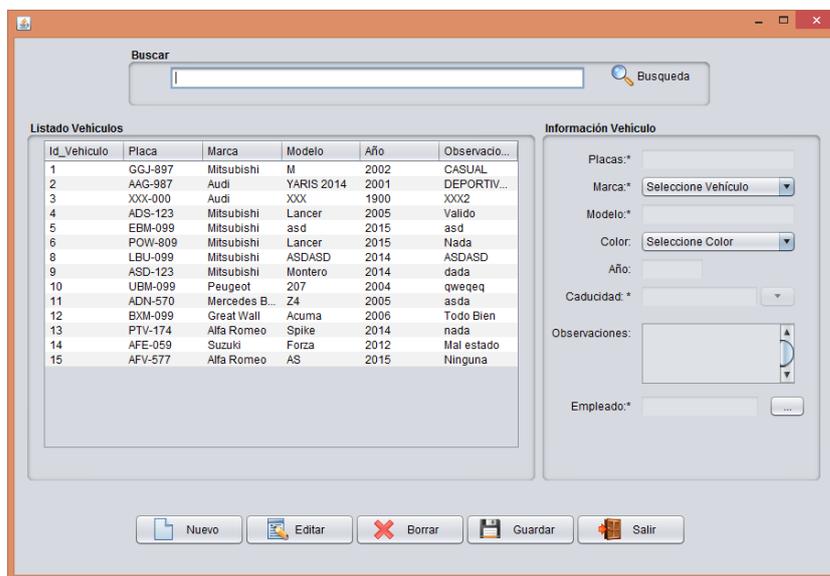


Figura 6.131 Interfaz de mantenimiento.

**Observaciones:** Los procesos de creación, modificación y eliminación se realizan rápidamente sin muchas ventanas emergentes, además la búsqueda es instantánea. Todas las interfaces presentan la misma estructura por lo que el usuario se familiariza rápidamente con las operaciones de los distintos mantenimientos.

## Prueba 5: Interfaz de consultas.

**Condiciones:** Prueba realizada en la consulta de registros de vehículos ingresados.

## Captura:

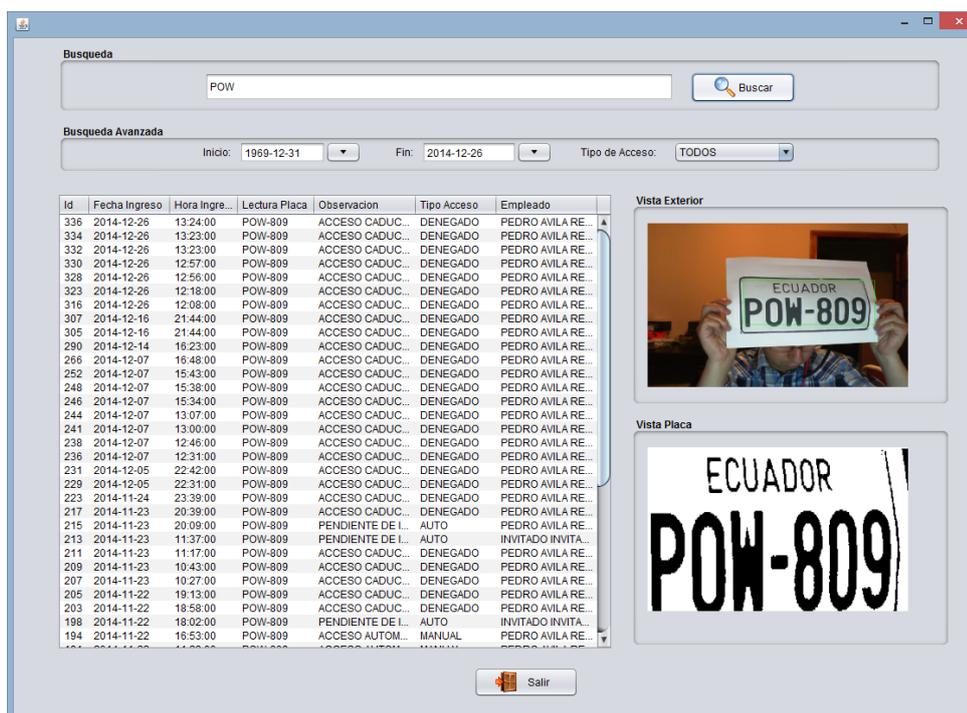


Figura 6.132 Interfaz de consulta de registros de acceso.

**Observaciones:** La interfaz presenta toda la información relevante para consultar los registros de ingreso. La búsqueda contempla todos los criterios necesarios en un solo campo de texto y presenta una efectiva búsqueda avanzada para limitar los resultados por tipos de acceso y rangos de fechas.

## Conclusión

Las pruebas realizadas permiten determinar el rendimiento óptimo del sistema, y de igual forma, permite conocer la capacidad actual del mismo determinando aquellas limitaciones que pueden ser mejoradas a futuro.

Como se había especificado anteriormente la capacidad de detección es afectada por la capacidad de la cámara, además, de factores externos como la iluminación y la distancia.

## Capítulo 7

### Manual de usuario

#### Introducción

Como último punto para esta investigación se plantea especificar un manual de usuario para el correcto uso de cada módulo contemplado en el sistema. Toda la información en este capítulo especificará los pasos necesarios para efectuar las diferentes operaciones del sistema, por ello se dividirá en 2 partes diferenciadas por cada rol que presenta:

- **Supervisor:** Encargado de las operaciones de ingreso, consulta y registro de vehículos.
- **Administrador:** Además de realizar las tareas del supervisor, también realizará los mantenimientos de: empleados, cargos, vehículo y usuarios.

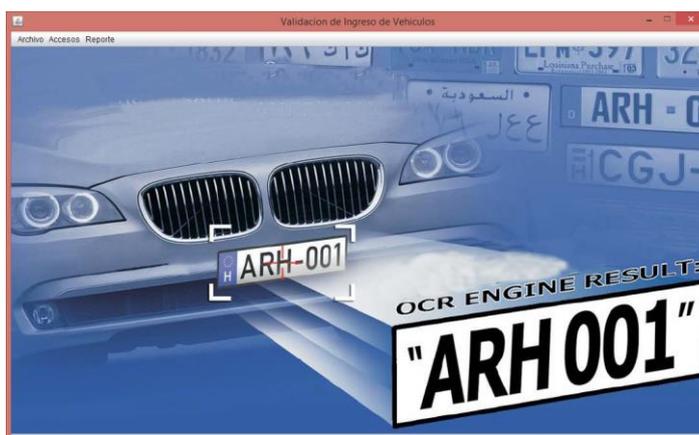
#### 7.1. Manual para el usuario supervisor.

El sistema mostrará la ventana de inicio en el cual se deberá ingresar las credenciales del usuario.



Una vez ingresada la información se presiona en el botón  para ingresar al sistema o caso contrario si se desea salir se presiona el botón .

Una vez ingresado en el sistema aparecerá el panel principal.



### 7.1.1. Menú del usuario supervisor.



Este menú mostrará las siguientes opciones:

- **Archivo:** Permite salir del sistema.
- **Accesos:** En esta sección se encuentra la opción de **detección de placa**, para la ejecución del reconocimiento de placas de vehículos.
- **Consultas:** Presenta la consulta de información en el ingreso y se divide en:
  - **Empleados:** En esta opción se puede consultar la información del empleado.
  - **Registros de Acceso:** Aquí se puede consultar la información de los accesos de los vehículos.

### 7.1.2. Detección de placa

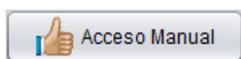
En esta sección se realiza todo el proceso de detección de la placa vehicular para el ingreso, su proceso es automático para las diferentes situaciones y su estructura es la siguiente:



- **Captura de la Placa:** Mostrará la placa capturada.
- **Información Detectada:** Si el vehículo es reconocido y forma parte de un empleado, esta sección mostrará:
  - El nombre del empleado dueño del vehículo
  - El modelo registrado del vehículo
  - La placa leída por el sistema.
- **Semáforo:** Indicador del acceso del vehículo cada color representa un estado:
  - Rojo para acceso denegado
  - Amarillo para sistema en espera
  - Verde para acceso concedido.
- **Información:** Información del estado y las operaciones que realiza el sistema.

#### 7.1.2.1. Acceso manual

Para realizar un acceso manual solo se debe presionar el botón,

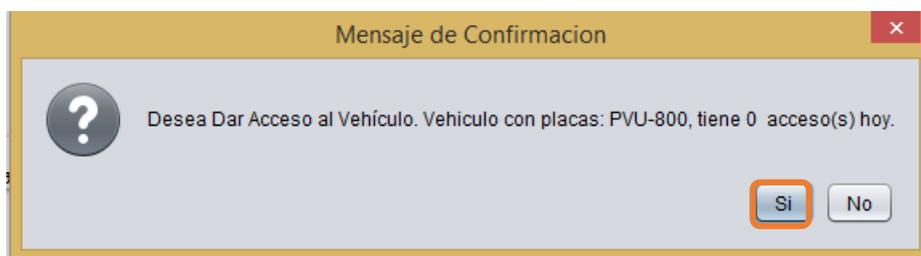


este realizará el acceso del vehículo tomando una captura para respaldar el mismo.

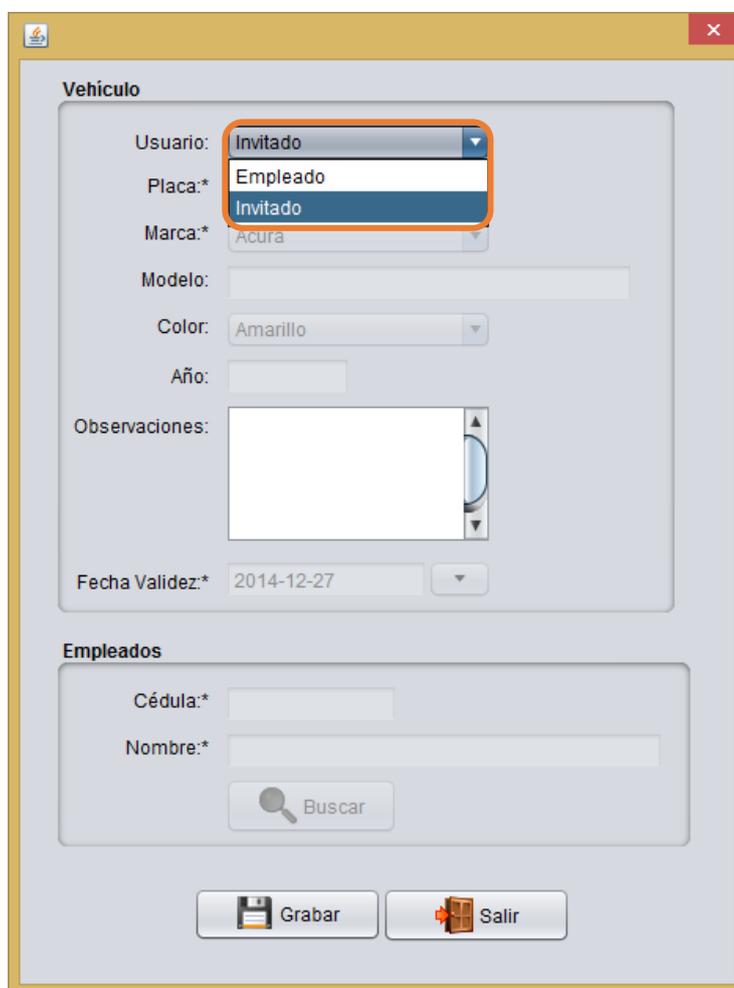
#### 7.1.2.2. Acceso eventual como invitado

Para permitir el ingreso de un vehículo como invitado, en el módulo de detección de placas se debe seguir los siguientes pasos:

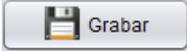
- Al ingresar el vehículo no registrado, aparecerá un mensaje de consulta para la confirmación del registro, presionar sobre la opción “**Sí**”.



- Aparecerá el cuadro de registro de vehículo. En la sección **usuario** seleccionar la opción invitado.

A screenshot of a vehicle registration form. The form is divided into two sections: "Vehículo" and "Empleados". In the "Vehículo" section, the "Usuario:" dropdown menu is open, showing options "Invitado", "Empleado", and "Invitado", with "Invitado" selected. Other fields include "Placa:\*" (empty), "Marca:\*" (Acura), "Modelo:" (empty), "Color:" (Amarillo), "Año:" (empty), "Observaciones:" (empty text area), and "Fecha Validez:\*" (2014-12-27). The "Empleados" section has "Cédula:\*" (empty), "Nombre:\*" (empty), and a "Buscar" button. At the bottom, there are "Grabar" and "Salir" buttons.

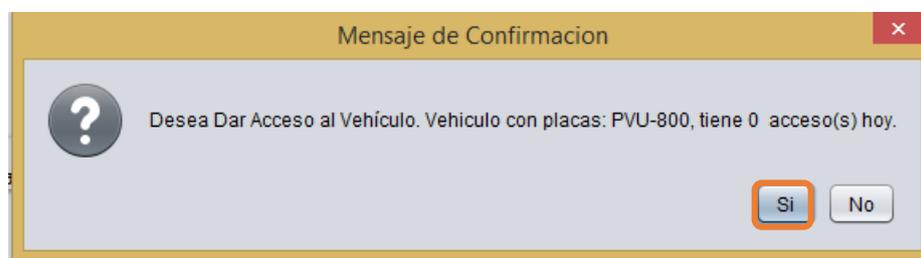
- Todos los campos se bloquearán excepto la sección **observaciones** en la que se podrán escribir algún tipo de información del vehículo que accede. Este campo no será obligatorio.

- Para finalizar presionar en el botón . La información será registrada con éxito y se permitirá el acceso al vehículo.

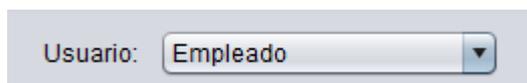
### 7.1.2.3. Acceso de vehículo con registro nuevo

Para permitir el registro de un vehículo con un empleado de la institución en el módulo de detección de placas, se debe seguir los siguientes pasos:

- Al ingresar el vehículo no registrado aparecerá un mensaje de consulta para la confirmación del registro, presionar sobre la opción “**Sí**”.



- Aparecerá el cuadro de registro, verificar que en la sección **usuario** se encuentre seleccionada la opción “**Empleado**” y todos los campos del registro desbloqueados.



- Para la creación del registro se debe llenar los siguientes campos:

**Vehículo**

Usuario: Empleado

Placa:\* PVU-800

Marca:\* Ford

Modelo: Eco Sport

Color: Gris

Año: 2012

Observaciones: Aros Anchos

Fecha Validez:\* 2014-12-31

**Empleados**

Cédula:\*

Nombre:\*

Buscar

Grabar Salir

- **Placa:** Se llenará automáticamente por el número de placa detectado.
- **Marca:** Selección de la marca del vehículo a partir de un listado. Si no se encuentra, poner otros y especificar en observaciones.
- **Modelo:** Campo de texto se para especificar el modelo del vehículo.
- **Color:** Selección de la marca del color de vehículo a partir de un listado.
- **Año:** Campo numérico en el que se especificará el año del vehículo.
- **Observaciones:** Texto no obligatorio para especificar alguna característica del vehículo.
- **Fecha validez:** Se especifica hasta que fecha se validará el ingreso del vehículo, una vez se sobrepase esta, se le denegará el acceso.

- A continuación se debe agregar el empleado propietario del vehículo, para ello, se debe ubicar en la sección **“Empleados”** y presionando el botón **“Buscar”**.

**Empleados**

Cédula:\*

Nombre:\*

Buscar

- Se desplegará un listado de todos los empleados de la institución, se escoge el empleado deseado y se presiona en **“Seleccionar”** para agregarlo al vehículo.

Buscar:

Cedula/RUC	Nombre	Apellido	Dirección	Teléfono	Celular	Email
0104516984	FABIAN MAU...	ATARIGUAN...	STA MARIA2	4091729	0984822703	fabian@mail...
0105925408	PEDRO AVIL...	AVILA REGA...	DIARIO TIEM...	2808080	098123456	pedroa@ma...
0105253470	INVITADO	INVITADO	DIARIO TIEM...	2808080	098123456	pedroa@ma...
0105925408	Pedro	Avila	Luis Cordero	2885222	0983468061	peceav@la...

- Al agregar se mostrará la información del empleado seleccionado en la ventana de registro.

**Empleados**

Cédula:\*

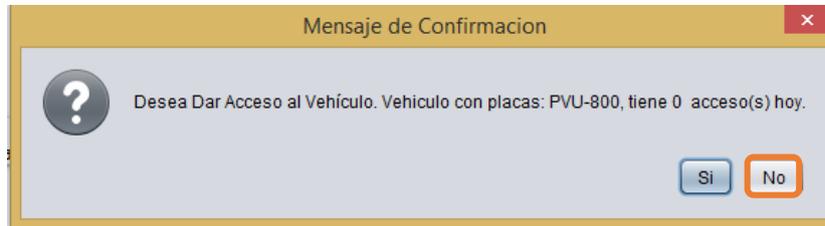
Nombre:\*

Buscar

- Finalmente se presiona en Grabar para finalizar el proceso, almacenar el vehículo y permitir el acceso.

#### 7.1.2.4. Denegar acceso a vehículo no registrado

Cuando se desea denegar el acceso a un vehículo no registrado, luego de aparecer el mensaje de confirmación únicamente se debe presionar en el botón “No”.



La interfaz mostrará el semáforo en rojo y se almacenará la incidencia.

#### 7.1.3. Consultas

Corresponde a todos los módulos de consulta que permiten conocer la información de empleados y de registros de acceso.

##### 7.1.3.1. Consulta de registro de acceso

Para ingresar la consulta de acceso se ingresa desde el menú del sistema a la opción Consultas, posteriormente al submenú Registros de Acceso. Mostrará la siguiente interfaz:

Busqueda

Busqueda Avanzada

Inicio: 1999-12-31 Fin: 2014-12-27 Tipo de Acceso: TODOS

Id	Fecha Ingreso	Hora Ingreso	Lectura Placa	Observacion	Tipo Acceso	Empleado
349	2014-12-27	16:57:00	PVU-800	ACCESO R...	AUTO	FABIAN MAJ...
348	2014-12-27	16:19:00	ADN-570	ACCESO CA...	DENEGADO	FABIAN MAJ...
347	2014-12-27	16:10:00	POW-809	ACCESO CA...	DENEGADO	PEDRO AVI...
346	2014-12-26	18:04:00	L-160	NO SE PER...	DENEGADO	INVITADO IN...
345	2014-12-26	16:15:00	AFV-577	ACCESO AU...	MANUAL	FABIAN MAJ...
344	2014-12-26	16:15:00	PVU-800	NO SE PER...	DENEGADO	INVITADO IN...
343	2014-12-26	16:14:00	AFV-577	ACCESO AU...	MANUAL	FABIAN MAJ...
342	2014-12-26	16:14:00	PVU-800	NO SE PER...	DENEGADO	INVITADO IN...
341	2014-12-26	16:14:00	AFV-577	ACCESO AU...	MANUAL	FABIAN MAJ...
340	2014-12-26	16:12:00	ADN-570	ACCESO CA...	DENEGADO	FABIAN MAJ...
339	2014-12-26	16:08:00	AFV-577	ACCESO R...	AUTO	FABIAN MAJ...
338	2014-12-26	13:25:00	ADN-570	ACCESO CA...	DENEGADO	FABIAN MAJ...
337	2014-12-26	13:25:00	AFE-059	ACCESO AU...	MANUAL	Pedro Avila
336	2014-12-26	13:24:00	POW-809	ACCESO CA...	DENEGADO	Pedro Avila
335	2014-12-26	13:24:00	AFE-059	ACCESO AU...	MANUAL	Pedro Avila
334	2014-12-26	13:23:00	POW-809	ACCESO CA...	DENEGADO	Pedro Avila
333	2014-12-26	13:23:00	AFE-059	ACCESO AU...	MANUAL	Pedro Avila
332	2014-12-26	13:23:00	POW-809	ACCESO CA...	DENEGADO	Pedro Avila
331	2014-12-26	12:57:00	AFE-059	NO SE PER...	DENEGADO	INVITADO IN...
330	2014-12-26	12:57:00	POW-809	ACCESO CA...	DENEGADO	Pedro Avila
329	2014-12-26	12:56:00	AFE-059	NO SE PER...	DENEGADO	INVITADO IN...
328	2014-12-26	12:56:00	POW-809	ACCESO CA...	DENEGADO	Pedro Avila
327	2014-12-26	12:55:00	AFE-059	NO SE PER...	DENEGADO	INVITADO IN...
326	2014-12-26	12:54:00	AFV-577	NO SE PER...	DENEGADO	INVITADO IN...
325	2014-12-26	12:54:00	AFE-059	NO SE PER...	DENEGADO	INVITADO IN...
324	2014-12-26	12:42:00	DWF-809	NO SE PER...	DENEGADO	INVITADO IN...
323	2014-12-26	12:18:00	POW-809	ACCESO CA...	DENEGADO	Pedro Avila
322	2014-12-26	12:18:00	OWH-809	NO SE PER...	DENEGADO	INVITADO IN...
321	2014-12-26	12:16:00	AFE-059	NO SE PER...	DENEGADO	INVITADO IN...
320	2014-12-26	12:11:00	ELE-859	NO SE PER...	DENEGADO	INVITADO IN...
319	2014-12-26	12:10:00	ZEA-639	NO SE PER...	DENEGADO	INVITADO IN...

Vista Exterior

Captura Exterior

Vista Placa

Vista Placa

Busqueda General

Busqueda Avanzada

Listado de Accesos

Salir

- **Búsqueda general:** Campo que permite la búsqueda de accesos por diferentes criterios como: nombres, apellidos, número de placa y observaciones de ingreso.
- **Búsqueda avanzada:** Adicionalmente se puede añadir a la búsqueda los siguientes criterios:
  - **Inicio:** Fecha inicial de búsqueda.
  - **Fin:** Fecha final de búsqueda.
  - **Tipo de acceso:** Selección del tipo de acceso otorgado en el ingreso del vehículo entre ellos: automático, manual y denegado.
- **Listado de acceso:** Listado de los accesos de vehículos en el sistema ordenados por fecha, entre ellos se incluye el nombre del empleado propietario.
- **Captura exterior:** Imagen de la placa y su entorno capturada en el acceso.
- **Captura procesada:** Placa recortada y binarizada capturada en el acceso.

#### 7.1.3.1.1. Búsqueda de registros

Para la búsqueda de cualquier registro, se escriben uno o varios criterios en el campo de búsqueda, luego se presiona el botón “**Buscar**”.



A screenshot of a search interface. It features a text input field containing the name 'PEDRO AVILA' and a button labeled 'Buscar' with a magnifying glass icon. The entire search bar is titled 'Busqueda'.

Si se desea una búsqueda más específica se puede adicionar criterios avanzados como un rango de fechas y un tipo de acceso.



A screenshot of advanced search filters. It shows three dropdown menus: 'Inicio:' with the value '2014-12-01', 'Fin:' with the value '2014-12-28', and 'Tipo de Acceso:' with the value 'DENEGADO'. Each dropdown menu is highlighted with an orange box.

El resultado mostrará un listado de registros con los criterios especificados. Si se desea visualizar las imágenes de las capturas de un determinado acceso, se debe seleccionar el registro deseado y las imágenes se visualizarán automáticamente en la parte derecha de la consulta.

Id	Fecha Ingreso	Hora Ingreso	Lectura Placa	Observacion	Tipo Acceso	Empleado
347	2014-12-27	16:10:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
336	2014-12-26	13:24:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
334	2014-12-26	13:23:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
332	2014-12-26	13:23:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
330	2014-12-26	12:57:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
328	2014-12-26	12:56:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
323	2014-12-26	12:18:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
316	2014-12-26	12:08:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
307	2014-12-16	21:44:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
305	2014-12-16	21:44:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
290	2014-12-14	16:23:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
266	2014-12-07	16:48:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
260	2014-12-07	16:32:00	EBM-099	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
252	2014-12-07	15:43:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
248	2014-12-07	15:38:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
246	2014-12-07	15:34:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
244	2014-12-07	13:07:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
241	2014-12-07	13:00:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
238	2014-12-07	12:46:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
236	2014-12-07	12:31:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
231	2014-12-05	22:42:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...
229	2014-12-05	22:31:00	POW-809	ACCESO ...	DENEGADO	PEDRO AVILA REGALADO ...

Vista Exterior



Vista Placa



### 7.1.3.2. Consulta de empleado

Para ingresar a la interfaz de consulta de empleados, desde el menú principal se debe ir al menú Consultas, posteriormente al submenú Empleados. Aparecerá el formulario con la siguiente información:

Busqueda

Cedula/RUC	Nombres	Apellidos	Dirección	Teléfono	Celular	Email
0104516984	FABIAN MAURICIO	ATARIGUANA QUE...	STA MARIA2	4091729	0984822703	fabian@mail.com
0105925408	PEDRO AVILA REG...	AVILA REGALADO	DIARIO TIEMPO	2808080	098123456	pedroa@mail.com
0105253470	INVITADO	INVITADO	DIARIO TIEMPO	2808080	098123456	pedroa@mail.com
0105925408	Pedro	Avila	Luis Cordero	2885222	098348081	peceav@lamotora...

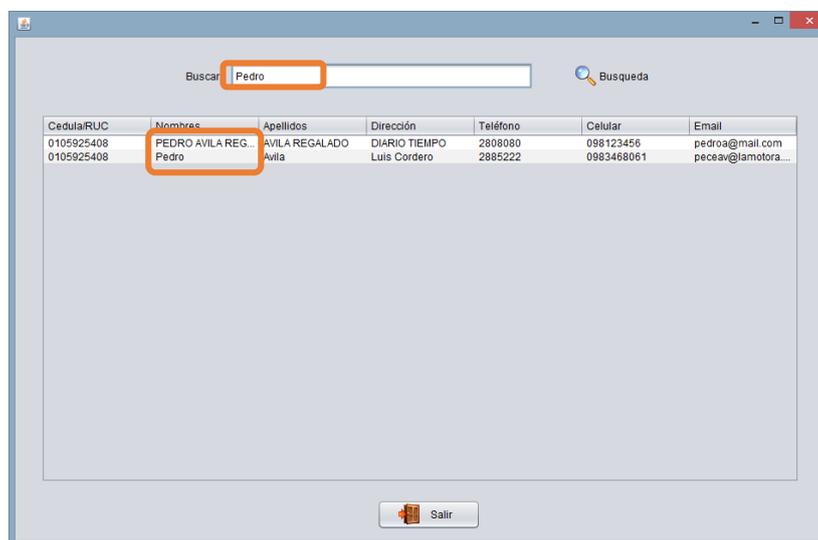
Listado de Empleados

Búsqueda

- **Búsqueda:** Campo que permite la búsqueda de los empleados mediante los criterios: nombres, apellidos y cédula.
- **Listado de empleados:** Listado de los registros de empleados con su información pertinente.

### 7.1.3.2.1. Búsqueda de empleados

El proceso de búsqueda de empleados es tan simple como ingresar el texto a buscar, el listado de empleados se irá filtrando progresivamente hasta mostrar los resultados deseados.

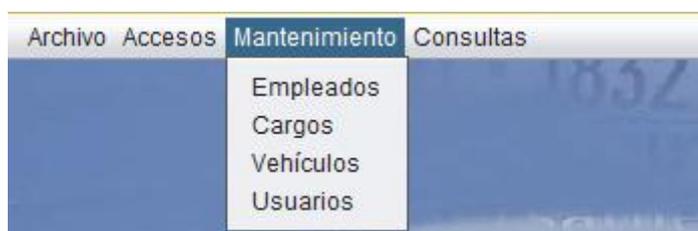


## 7.2. Manual de administrador

El rol de administrador tendrá acceso a todo el contenido del sistema, es decir, podrá ingresar a todos los módulos del supervisor, además de realizar las tareas de mantenimiento.

### 7.2.1. Menú de administrador

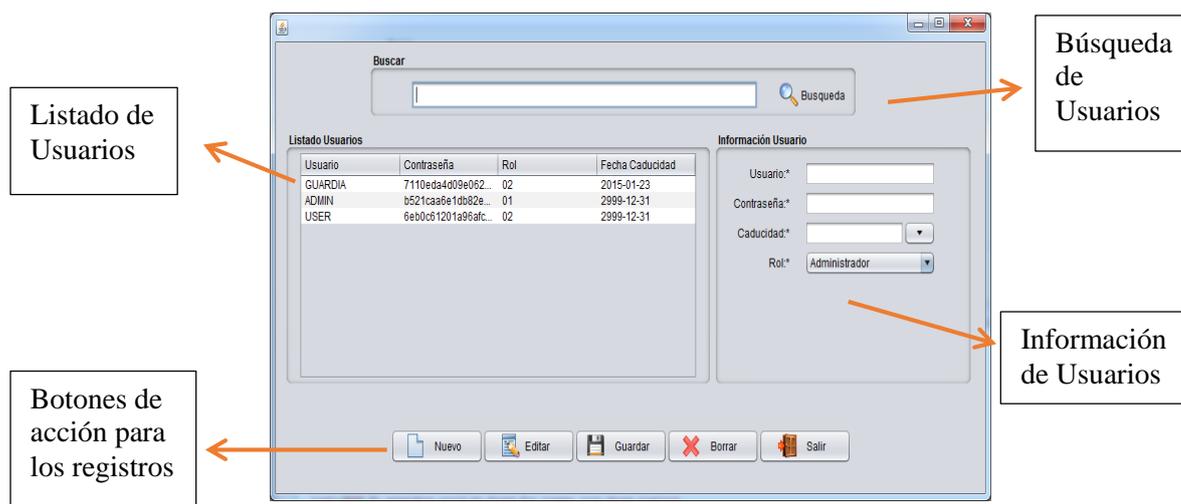
En el menú de administrador se añade la sección de Mantenimientos con las opciones de: Empleados, Cargos, Vehículos y Usuarios.



## 7.2.2. Mantenimientos

### 7.2.2.1. Usuarios

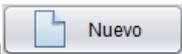
Para ingresar a la sección de usuarios desde el menú de administrador, se debe ingresar en el menú mantenimiento, posteriormente al submenú usuarios, este mostrará la siguiente información:



- **Listado de usuarios:** Permite visualizar todos los registros correspondientes a los usuarios registrados en el sistema.
- **Búsqueda de usuarios:** Permite filtrar el listado de registros de acuerdo al criterio de búsqueda correspondiente al nombre de usuario.
- **Información de usuarios:** Corresponde a toda la información perteneciente al usuario.
- **Botones de acción para los registros:** Son todas las operaciones que se pueden realizar con cada registro existente o al agregar registros nuevos.

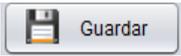
#### 7.2.2.1.1. Ingreso de usuarios

Para ingresar un nuevo usuario se siguen los siguientes pasos:

- Primero deberá dar clic en el botón .
- A continuación debe ingresar los datos en el apartado de información del usuario. Este consta de los siguientes campos:

El formulario 'Información Usuario' contiene los siguientes elementos:

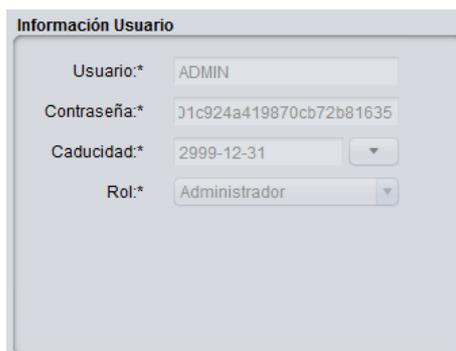
- Usuario:\***: Campo de texto para el nombre de usuario.
- Contraseña:\***: Campo de texto para la contraseña.
- Caducidad:\***: Campo de texto con un botón de flecha hacia abajo para seleccionar una fecha.
- Rol:\***: Menú desplegable con 'Administrador' seleccionado.

- **Usuario:** Campo alfanumérico que especifica el nombre de usuario.
  - **Contraseña:** Campo alfanumérico que indica la contraseña para el usuario.
  - **Caducidad:** Campo que corresponde a una fecha que indicará la validación del usuario.
  - **Rol:** Es un listado de los roles que maneja el sistema. Se deberá elegir uno por cada usuario.
- Se deberá llenar todos los campos marcados con asterisco ya que son obligatorios.
  - Luego de ingresar toda la información requerida, se procede a dar clic en el botón  para que los datos sean registrados.

#### 7.2.2.1.2. Modificación de usuarios

La modificación para cualquiera de los campos del usuario es similar a los pasos especificados para el ingreso, para ello se debe realizar lo siguiente:

- El primer paso es seleccionar el registro que se desea modificar.
- Al seleccionar el registro deseado el apartado “Información de Usuario” cargará la información del mismo.



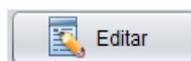
**Información Usuario**

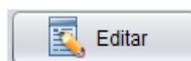
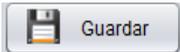
Usuario:\* ADMIN

Contraseña:\* 01c924a419870cb72b81635

Caducidad:\* 2999-12-31

Rol:\* Administrador



- Luego se debe presionar el botón  para habilitar los campos para su edición.
- Posteriormente se deberá modificar la información deseada, especificadas en la sección de información cuyas características ya fueron especificadas en el ingreso.
- Al terminar de modificar todos los campos deseados. Se debe dar clic en el botón  para almacenar la información.

### 7.2.2.1.3. Eliminación de Usuarios

La eliminación de usuarios en un proceso simple, para ello se debe:

- Seleccionar el registro que se desea eliminar.
- Una vez seleccionado el apartado “Información del Usuario” permitirá visualizar los datos a ser eliminados.



**Información Usuario**

Usuario:\* GUARDIA

Contraseña:\* 5e4a390b0a572ac0d2c0220

Caducidad:\* 2015-01-23

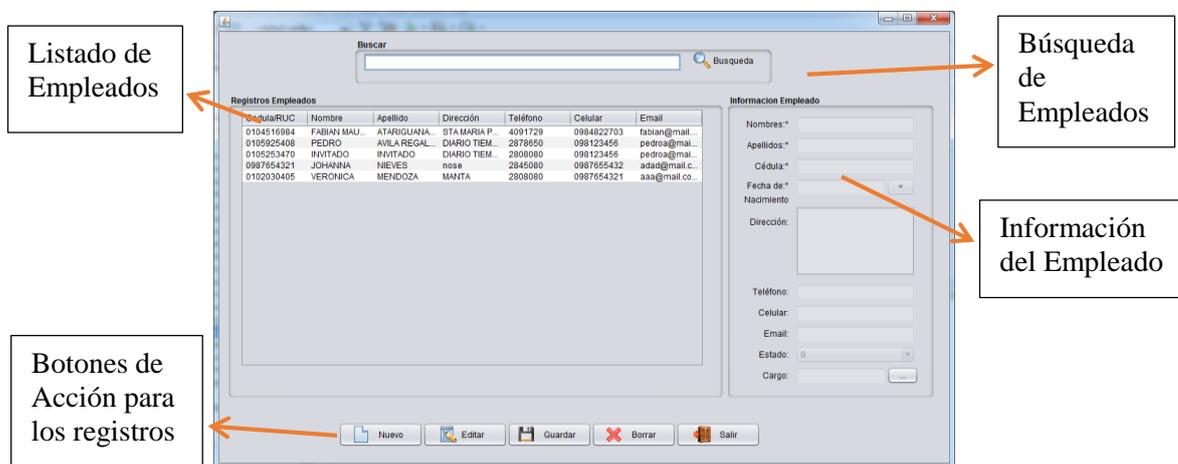
Rol:\* Supervisor



- A continuación se deberá dar clic en el botón Borrar .
- Puede comprobar que la eliminación fue correcta al visualizar que el registro eliminado no aparece en el listado de usuarios.

### 7.2.2.2. Empleados

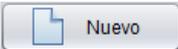
Para acceder a la sección de empleados desde el menú de administrador se debe ingresar al menú mantenimiento, posteriormente al submenú empleados, con lo cual se mostrará la siguiente información:



- **Listado de empleados:** Permite visualizar todos los registros correspondientes a los empleados almacenados en el sistema.
- **Búsqueda de empleados:** Filtra el listado de empleados de acuerdo a los campos: cedula/ruc, nombres y apellidos.
- **Información de empleados:** Corresponde a toda la información perteneciente al empleado.
- **Botones de acción para los registros:** Son todas las operaciones que se pueden realizar con cada registro existente o al agregar registros nuevos.

#### 7.2.2.2.1. Ingreso de empleados

En el ingreso de empleados se seguirán los siguientes pasos:

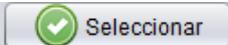
- Primero deberá dar clic en el botón. 
- A continuación se desbloquearán los campos de información para lo cual deberá ingresar la información del empleado:

El formulario 'Información Empleado' contiene los siguientes campos:

- Nombres:\* (campo de texto)
- Apellidos:\* (campo de texto)
- Cédula:\* (campo de texto)
- Fecha de:\* (campo de texto) y Nacimiento (botón de selección de fecha)
- Dirección: (campo de texto grande)
- Teléfono: (campo de texto)
- Celular: (campo de texto)
- Email: (campo de texto)
- Estado: (menú desplegable con el valor '0')
- Cargo: (campo de texto) y un botón de selección de lista (botón con tres puntos)

- **Nombres:** Valores alfanuméricos que corresponden a los nombres completos del empleado.
  - **Apellidos:** Valores alfanuméricos que corresponden a los apellidos paternos y maternos del empleado.
  - **Cédula/ruc:** Valores numéricos que corresponden a la cédula o ruc del empleado.
  - **Fecha de nacimiento:** Campo de tipo fecha que corresponde al nacimiento del empleado.
  - **Dirección:** Dirección del domiciliaria del empleado.
  - **Teléfono:** Número de teléfono convencional del empleado.
  - **Celular:** Número de teléfono móvil del empleado.
  - **Email:** Dirección de correo electrónico del empleado.
  - **Estado:** Corresponde a la selección de un valor ya establecido, define el estado activo/inactivo del empleado.
  - **Cargo:** Cargo al que corresponde el empleado y define las funciones que desempeña dentro de la institución.
- En el caso de "Cargo" se debe dar clic en el botón  el mismo que visualizará una ventana con los cargos existentes.

Id. Cargo	Descripcion
1	GUARDIANIA
2	SECRETARIADO
3	DOCENTES
4	ADMINISTRATIVOS
6	SERVICIO SOCIAL
7	IDIOMAS

- Se selecciona el cargo deseado y se presiona en el botón  para agregarlo al empleado.
- Deberá llenar todos los campos marcados con asterisco ya que son obligatorios.
- Luego de terminar de ingresar toda la información requerida, se procede a dar clic en el botón  para que los datos sean registrados.

#### 7.2.2.2.2. Modificación de empleados

Tomado en cuenta los datos de ingreso, la modificación de empleados se explica a continuación:

- El primer paso es seleccionar el registro que se desea modificar.
- Al seleccionar el registro deseado el apartado “Información de Empleado” cargará la información del mismo.

**Informacion Empleado**

Nombres:\* PEDRO

Apellidos:\* AVILA REGALADO

Cédula:\* 0105925408

Fecha de:\* 0015-01-10

Nacimiento

Dirección: DIARIO TIEMPO 2

Teléfono: 2878650

Celular: 098123456

Email: pedroa@mail.com

Estado: 1

Cargo: DOCENTES

- Como se observa esta información debe ser habilitada para la modificación,

eso se lo realiza al dar clic en el botón Editar



- Al habilitar los campos se deberá modificar la información deseada de acuerdo a las especificaciones anteriormente mencionadas para cada uno.
- Al terminar de modificar todos los datos deseados. Para guardar la

información se debe dar clic en el botón guardar



### 7.2.2.2.3. Eliminación de empleados

El procedimiento necesario para la eliminación de un empleado es el siguiente:

- Primero se debe seleccionar el registro que se desea eliminar.
- Al seleccionar el registro deseado el apartado “Información del Empleado” permitirá visualizar los datos a ser eliminados.

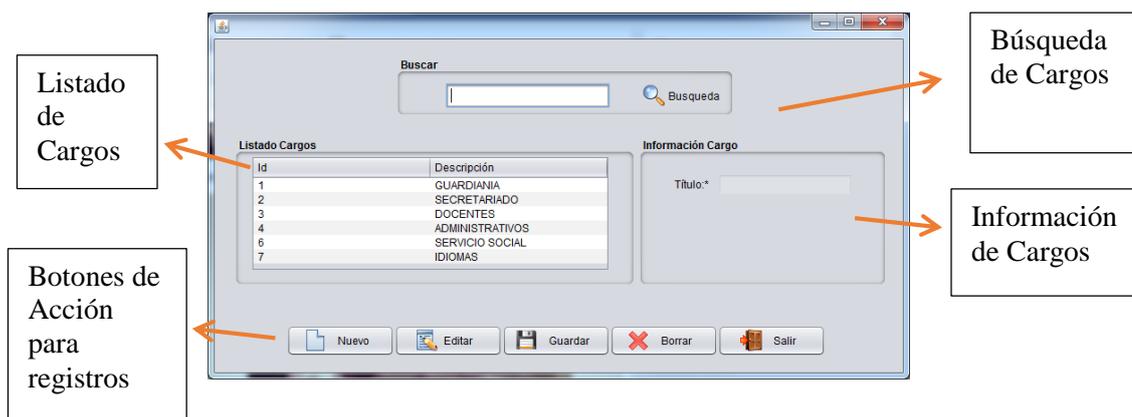
 A screenshot of a web form titled "Información Empleado". The form contains several input fields: "Nombres:\*" with the value "PEDRO", "Apellidos:\*" with "AVILA REGALADO", "Cédula:\*" with "0105925408", "Fecha de:\*" with "0015-01-10" and a dropdown arrow, "Dirección:" with "DIARIO TIEMPO 2", "Teléfono:" with "2878650", "Celular:" with "098123456", "Email:" with "pedroa@mail.com", "Estado:" with "1" and a dropdown arrow, and "Cargo:" with "DOCENTES" and a dropdown arrow.

- A continuación se deberá dar clic en el botón Borrar
- Puede comprobar que la eliminación fue correcta al visualizar que el registro eliminado no aparece en el listado de empleados.



### 7.2.2.3. Cargos

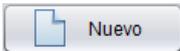
Para acceder al mantenimiento de cargos desde el menú del administrador se debe ir al menú mantenimientos, posteriormente al submenú cargos. En esta interfaz se tiene la siguiente información:



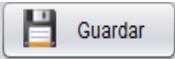
- **Listado de cargos:** Permite visualizar todos los registros correspondientes a los cargos registrados en el sistema.
- **Búsqueda de cargos:** Permite filtrar el listado de cargos de acuerdo a su descripción.
- **Información de empleados:** Corresponde a la información perteneciente al cargo del empleado.
- **Botones de acción para los registros:** Son todas las operaciones que se pueden realizar con cada registro existente o al agregar registros nuevos.

#### 7.2.2.3.1. Ingreso de cargos

El ingreso de cargos especifica el rol del empleado dentro de la organización, por lo que constará solamente del nombre del mismo para ser enlazado posteriormente a un empleado. Los pasos a seguir son los siguientes:

- Primero deberá dar clic en el botón 
- A continuación debe ingresar los datos en el apartado de información del cargo. Este constará de un único campo:

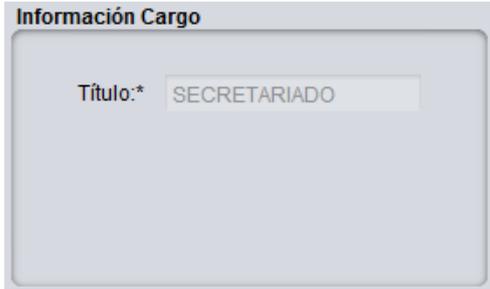
The form titled "Información Cargo" contains a single text input field with the label "Título:\*".

- **Título:** Campo alfanumérico que corresponde a la descripción del cargo para un empleado.
- Luego de terminar de ingresar toda la información requerida, se procede a dar clic en el botón  para que los datos sean registrados.

#### 7.2.2.3.2. Modificación de cargos

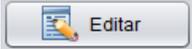
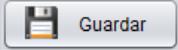
En la modificación de un cargo, se toma en consideración los siguientes pasos:

- Seleccionar el cargo que se desea modificar.
- El apartado “Información de Cargo” visualizará sus datos.



Información Cargo

Título:\* SECRETARIADO

- Como se observa esta información debe ser habilitada para la modificación, eso se lo realiza al dar clic en el botón. 
- Al habilitar los campos se deberá modificar la información deseada en este caso el único campo modificable es el “**Título del Cargo**”.
- Al terminar de modificar todos los datos deseados. Para guardar la información se debe dar clic en el botón. 

#### 7.2.2.3.3. Eliminación de cargos

El proceso de eliminación de un cargo se debe seguir los siguientes pasos:

- Lo primero es seleccionar el cargo que se desea eliminar.
- Al seleccionar el registro deseado el apartado “Información del Cargo” permitirá visualizar los datos a ser eliminados.

**Información Cargo**

Título:\* SECRETARIADO



- A continuación se deberá dar clic en el botón Borrar
- Puede comprobar que la eliminación fue correcta al visualizar que el registro eliminado no aparece en el listado de cargos.

#### 7.2.2.4. Vehículos

Si se desea acceder al módulo de mantenimiento de vehículos desde el menú de administrador se debe ir a la sección mantenimiento, posteriormente ir al submenú vehículos. Se desplegará la interfaz con la siguiente información:

Buscar

Busqueda

Listado de Vehículos

Id_Vehículo	Placa	Marca	Modelo	Año	Observación
1	GGJ-897	MAZDA	M	2002	CASUAL
2	AAG-987	TOYOTA	YARIS 2014	2001	DEPORTIVO
3	XXX-000	XXX	XXX	1900	XXX
4	ACP-110	Mercedes B...	Z8	2013	El Carro es ...
5	POW-809	MERCEDES	ASD	2014	Nada
6	AAA-111	MAZDA	CAMIONETA	2014	NINGUNA
7	AFH-876	LAMBORGH...	MURCIELA...	2014	CHEVERE ...
8	ADX-590	Acura	IX	2010	NINGUNA
9	AFP-310	Alfa Romeo	TUCO	2015	NINGUNA
10	GGJ-082	Suzuki	SZ	2014	NINGUNA
11	PTV-172	Mercedes B...	SZ	2014	NINGUNA

Información Vehículo

Placas:\*

Marca:\* Seleccione Vehículo

Modelo:\*

Color: Seleccione Color

Año:

Caducidad:\*

Observaciones:

Empleado:\*

Botones de Acción para registros

Nuevo Editar Borrar Guardar Salir

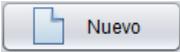
Búsqueda de Vehículos

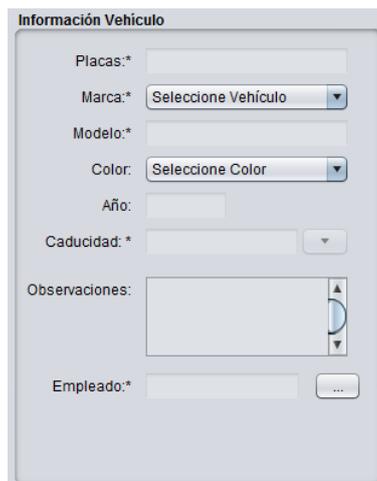
Información de Vehículos

- **Listado de Vehículos:** Permite visualizar todos los registros correspondientes a los vehículos registrados en el sistema.
- **Búsqueda de Vehículos:** Permite filtrar el listado de vehículos de acuerdo a los criterios: placa, marca y modelo.
- **Información de Vehículos:** Corresponde a toda la información perteneciente al vehículo del empleado.
- **Botones de acción para los registros:** Son todas las operaciones que se pueden realizar con cada registro existente o al agregar registros nuevos.

#### 7.2.2.4.1. Ingreso de vehículos

En el ingreso de vehículos se toma en consideración los siguientes pasos:

- Primero se deberá dar clic en el botón Nuevo: 
- A continuación se debe ingresar los datos en el apartado de información de vehículos. Esta sección consta de los siguientes campos:

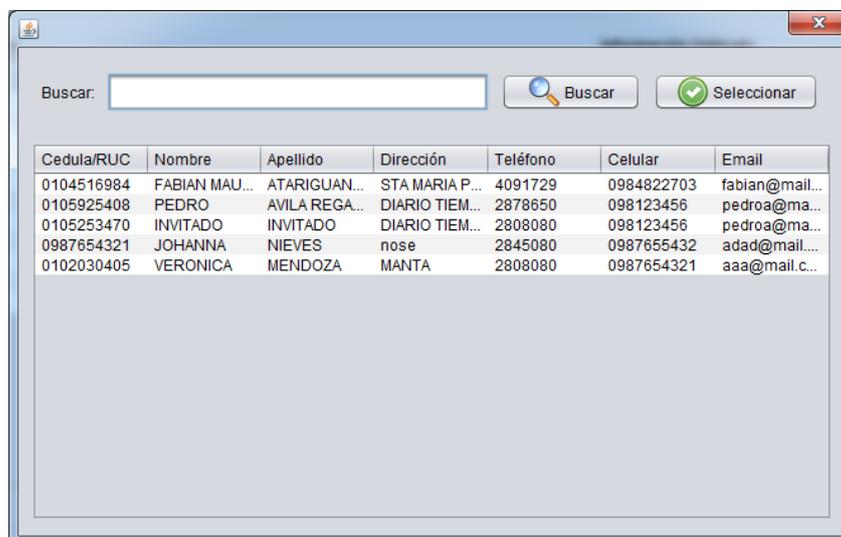


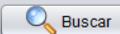
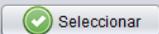
El formulario 'Información Vehículo' contiene los siguientes campos:

- Placas:\* (campo de texto)
- Marca:\* (menú desplegable con 'Seleccione Vehículo')
- Modelo:\* (campo de texto)
- Color: (menú desplegable con 'Seleccione Color')
- Año: (campo de texto)
- Caducidad:\* (menú desplegable)
- Observaciones: (área de texto con barra de desplazamiento)
- Empleado:\* (campo de texto con botón de selección '...')

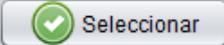
- **Placas:** Campo que corresponde a los dígitos de la placa.
  - **Marca:** Valor establecido por las marca del vehículo, este será seleccionado a partir de un listado ya definido.
  - **Modelo:** Corresponde al modelo del vehículo.
  - **Color:** Valor que corresponde al color del vehículo, este será seleccionado de un listado previamente establecido.
  - **Año:** Valor numérico que representa el año de fabricación del vehículo.
  - **Caducidad:** Valor que define la fecha máxima de ingreso del vehículo al parqueadero.
  - **Observaciones:** Campo que define cualquier detalle adicional sobre el vehículo.
- Todos los campos marcados con asterisco deberán ser llenados ya que son obligatorios.

- A continuación se debe agregar el "Empleado" que será propietario del vehículo, para ello se da clic en el botón  el mismo que visualizará una pantalla con un listado de empleados registrados.



Buscar:   

Cedula/RUC	Nombre	Apellido	Dirección	Teléfono	Celular	Email
0104516984	FABIAN MAU...	ATARIGUAN...	STA MARIA P...	4091729	0984822703	fabian@mail...
0105925408	PEDRO	AVILA REGA...	DIARIO TIEM...	2878650	098123456	pedroa@ma...
0105253470	INVITADO	INVITADO	DIARIO TIEM...	2808080	098123456	pedroa@ma...
0987654321	JOHANNA	NIEVES	nose	2845080	0987655432	adad@mail...
0102030405	VERONICA	MENDOZA	MANTA	2808080	0987654321	aaa@mail.c...

- A continuación se debe seleccionar el empleado deseado y dar clic en el botón  para agregarlo al vehículo
- Luego de terminar de ingresar toda la información requerida, se procede a dar clic en el botón  para que los datos sean almacenados.

#### 7.2.2.4.2. Modificación de vehículos

La modificación de vehículos se realiza de la siguiente manera:

- Primero se debe seleccionar el empleado que se desea modificar.
- La selección mostrará en el apartado "Información de Vehículo" todos los datos del mismo.

**Información Vehículo**

Placas:\* AFP-310

Marca:\* Alfa Romeo

Modelo:\* TUCO

Color: Azul

Año: 2015

Caducidad:\* 2014-12-07

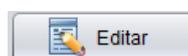
Observaciones: NINGUNA

Empleado:\* 0987654321

NIEVES JOHANNA

- Como se observa esta información debe ser habilitada para la modificación,

esto se lo realiza al dar clic en el botón.



- Al habilitar los campos, se deberá modificar la información deseada de acuerdo a las especificaciones anteriormente mencionadas en el ingreso.
- Al terminar de modificar todos los datos deseados. Para guardar la

información se debe dar clic en el botón.



#### 7.2.2.4.3. Eliminación de vehículos

Si se desea eliminar un vehículo se siguen los pasos similares a los mencionados en previos mantenimientos, estos son:

- Seleccionar el registro que se desea eliminar.
- Comprobar que el apartado “Información del Vehículo” visualice los datos del vehículo a eliminar.

**Información Vehículo**

Placas:\* AFP-310

Marca:\* Alfa Romeo

Modelo:\* TUCO

Color: Azul

Año: 2015

Caducidad:\* 2014-12-07

Observaciones: NINGUNA

Empleado:\* 0987654321

NIEVES JOHANNA

- A continuación se debe dar clic en el botón. 
- Puede comprobar que la eliminación fue correcta al visualizar que el registro no aparece en el listado de empleados.

## CONCLUSIONES

El resultado del tema de investigación realizado, presenta un gran avance en el ámbito de la seguridad con el uso de la visión artificial, ya que, cumple objetivamente su propuesta, planteando una solución viable en el proceso de detección. Como se ha especificado anteriormente, esta tecnología es de gran ayuda en procesos repetitivos que requieren soluciones rápidas, por lo cual, este caso de estudio sirve para demostrar cómo actúa ante un problema real como es el de reconocimiento de placas para ingresos de parqueaderos.

Principalmente el aprendizaje en el proceso de desarrollo ha sido amplio y progresivo, ya que para una correcta ejecución del sistema, fue necesario tener un conocimiento avanzado de la tecnología, y realizar múltiples pruebas determinando su efectividad frente a los problemas planteados. Cada error sirvió de aprendizaje ayudando a trazar nuevas soluciones hasta obtener los resultados correctos, y aun así, posteriormente se continuó afinando sus características para optimizar cada vez más sus resultados.

La tecnología de visión artificial se mantiene en constante evolución, específicamente en su herramienta principal Open CV que obtuvo actualizaciones de sus librerías para facilitar su método de entrenamiento que representa su punto más crítico. Anteriormente Open CV empleaba un método denominado haartraing que utilizaba un solo núcleo del pc, llevando a realizar cada entrenamiento de 7 a 10 días. Luego de obtener la actualización se obtuvo una mejora notable al incluir la técnica traincascade, esta permitía realizar entrenamientos utilizando todos los recursos de la pc y así reducir notablemente el tiempo de entrenamiento a 5 días o menos.

Adicionalmente, para llegar a los resultados deseados en el entrenamiento, fue necesario encontrar el equilibrio, entre el número de etapas y el número de muestras, esta mejora se realizó mediante varias pruebas hasta encontrar la solución más acertada, llevando a determinar 3 factores importantes en este proceso:

- A mayor número de etapas, mayor tiempo de entrenamiento, mayor precisión en detección y requiere mayor número de muestras.

- A menor número de etapas, menor tiempo de entrenamiento, menor precisión en detección y requiere menor número de muestras.
- El número de etapas y muestras dependerá de la complejidad del objeto frente a sus características únicas.

El número exacto de etapas en el caso de entrenamiento fue determinado por múltiples pruebas hasta llegar al número 21, que fue ideal contemplando la cantidad de 544 muestras positivas y 1012 negativas. Todo este proceso determina que no existe la manera de llevar un entrenamiento de un objeto específico, sin la necesidad de establecer la filosofía prueba/error y por ello hacen falta herramientas que permitan realizarlo de manera simple y efectiva, evitando largos procesos de entrenamiento para obtener resultados erróneos.

En las etapas de desarrollo y pruebas, se notó que la funcionalidad más sobresaliente de Open CV es la detección, cada vez que la placa aparece frente a la cámara el reconocimiento es inmediato, inclusive en capturas borrosas o placas en mala estado. En este proceso se determinó que la cámara utilizada es un factor muy importante a considerar, ya que optimiza aspectos como la distancia o calidad de la imagen, y así obtener una mejor lectura de caracteres.

En cuanto a la tecnología OCR, se identificó que la herramienta utilizada Tesseract presenta un sistema de entrenamiento mucho más rápido, pero más complejo que Open CV, ya que el proceso de aprendizaje consta de demasiados factores a considerar, resultando muy complejo al realizar varias pruebas.

Considerando que OCR posee algoritmos avanzados de detección, este proceso requiere características muy específicas y realiza comparaciones muy exactas en la lectura de caracteres, haciendo que letras levemente deterioradas o de diferente tipografía no sean detectadas. Estos inconvenientes llevaron a realizar algoritmos de validación y depuración en el sistema, con el objetivo de mejorar la detección y obtener una lectura correcta. Por ello se establece que si bien el OCR ha evolucionado notablemente en la lectura de caracteres, requiere un mejoramiento orientado a la depuración de resultados para identificar a una letra en un conjunto de símbolos o imperfecciones.

Todas estas tecnologías llevan a cumplir los objetivos propuestos en la realización de un notable sistema de ingreso, de simple interfaz, pero de compleja estructura, que ágilmente propone una alternativa mucho más completa a la actualmente utilizada en la universidad. Todas sus características se enfocaron en establecer un uso sin complicaciones para el usuario, pero que abarque todas las situaciones que se pueden presentar en el ingreso diario de vehículos.

Finalmente, es necesario especificar que el sistema propuesto contiene características que lo hacen escalable, esto quiere decir, que se puede mantener en constante mejora, realizando entrenamientos con un mayor número de muestras o placas con diferentes características. Todo esto debido a que se encuentra estructurado de tal manera, que sus archivos de entrenamiento son considerados como parámetros para permitir una posterior ampliación de características.

## RECOMENDACIONES

A lo largo del presente caso de estudio se consideró varios aspectos para el uso correcto de la tecnología de visión artificial:

- Estudiar y analizar las herramientas de acuerdo a su uso, ya que estas poseen funciones específicas para distintas tareas. En este caso las herramientas de procesamiento de imagen fueron escogidas de acuerdo a su rapidez y su capacidad de integración.
- Mantener siempre actualizadas las herramientas de visión artificial. Open CV realiza actualizaciones con mejoras notables para sus algoritmos de detección.
- Determinar el número de muestras de acuerdo a la complejidad del objeto, tomando en cuenta un mínimo de 200 imágenes.
- Recolectar las muestras de caracteres en la mayor resolución posible y siempre en formato TIF. Adicionalmente, es necesario mantener siempre los caracteres en un formato similar al de la lectura y contemplar un mínimo de 5 muestras por letra.
- Mantener accesibles y documentadas las muestras de entrenamiento. En este proceso será necesario una retroalimentación para mejorar los resultados.
- Investigar herramientas externas que ayuden en el proceso de entrenamiento. Algunos usuarios crean utilitarios para la ayuda de los procesos de entrenamiento, en este caso sirvieron las herramientas Box para Tesseract y “Rename it” para organizar las muestras en Open CV.
- Mantener las herramientas de visión artificial en hilos de procesamiento que ayuden a ejecutar acciones paralelas, esto permitirá que el proceso de detección se mantenga siempre en ejecución mientras se realizan otras acciones en el sistema.
- Emplear cámaras de gran calidad para una mayor distancia de detección y para una captura de mayor resolución. Adicionalmente incluir sistemas de iluminación para ambientes oscuros.
- Emplear el versionamiento de todo el sistema con la ayuda de componentes externos como subversión de Google o apache.org.

- Implementar el uso de Hibernate para procesos que requieren de persistencia en la base de datos, ya que esto permite realizar consultas de manera más dinámica y obtener datos con la misma facilidad con la que se maneja objetos dentro de la programación.

## BIOBLOGRAFIA

- Arlandis, J. (2013). *Instituto tecnológico de información*. Obtenido de <http://www.iti.es/>.
- Carrasco, J. (2003). *Inaoe*. Obtenido de <http://ccc.inaoep.mx/~ariel/recpat.pdf>
- CEDIBA. (2006). *visionartificial.com.a*. Obtenido de <http://www.visionartificial.com.ar/va.htm>
- Centro Integrado Politécnico. (2010). *Etitudela*. Obtenido de Etitudela: <http://www.etitudela.com/celula/downloads/visionartificial.pdf>
- Daniel Bescansa, I. G. (2006). *atalayadelemprendedor.com*. Obtenido de [http://www.atalayadelemprendedor.com/www.atalayadelemprendedor.com/images/pdf/La\\_vision\\_artificial\\_una\\_oportunidad.pdf](http://www.atalayadelemprendedor.com/www.atalayadelemprendedor.com/images/pdf/La_vision_artificial_una_oportunidad.pdf).
- Druzella, U. (2010). <http://www.jcee.upc.edu/>. Obtenido de [http://www.jcee.upc.edu/JCEE2010/pdf\\_ponencias/PDFs/09\\_12\\_10/Vision%20Artificial%20UNI%20TERRASSA%202010.pdf](http://www.jcee.upc.edu/JCEE2010/pdf_ponencias/PDFs/09_12_10/Vision%20Artificial%20UNI%20TERRASSA%202010.pdf)
- Helmut Grabner, H. B. (2006). *On-line Boosting and Vision*. Obtenido de <http://www.vision.ee.ethz.ch/boostingTrackers/Grabner2006On-lineBoostingand.pdf>
- Helmut Grabner, H. B. (2006). *On-line Boosting and Vision* .
- Hibernate. (s.f.). *Hibernate*. Recuperado el 04 de 12 de 2014, de Hibernate: <http://hibernate.org/orm/>
- Infaimon. (16 de Febrero de 2012). *infaimon.com*. Obtenido de <http://blog.infaimon.com/etiqueta/thermalsentry/>
- INFAIMON. (2013). *infaimon.com*. Obtenido de <http://www.infaimon.com/es/menu/aplicaciones>
- Javier Cano, J. C. (2010). <http://www.iti.es/>.
- LSIIS. (2004). *Dep. de Lenguajes y Sistemas Informáticos*. Obtenido de <http://lml.ls.fi.upm.es/ep/entornos.html>
- Matlab. (2014). *Mathworks*. Obtenido de <http://www.mathworks.com/products/matlab/>
- Matlab Image Processing toolbox. (2014). *Mathlab*. Obtenido de <http://www.mathworks.com/products/image/features.html#key-features>
- Microscan. (2013). <http://www.microscan.com>. Obtenido de <http://www.microscan.com/en-us/products/newproducts/visionhawk.aspx>

- MySQL. (s.f.). *MySQL*. Obtenido de MySQL:  
[http://dev.mysql.com/doc/refman/5.5/en/encryption-functions.html#function\\_sha1](http://dev.mysql.com/doc/refman/5.5/en/encryption-functions.html#function_sha1)
- Navarro Ubeda, R. (2004). *Universidad Politecnica de Valencia*. Obtenido de  
[http://www.gpiv.upv.es/publications/pdf/pfc\\_raquel.pdf](http://www.gpiv.upv.es/publications/pdf/pfc_raquel.pdf)
- Netbeans. (s.f.). *Netbeans*. Obtenido de  
<https://netbeans.org/features/platform/features.html>
- Open CV*. (12 de Marzo de 2013). Obtenido de [opencv.org](http://docs.opencv.org/): <http://docs.opencv.org/>
- Open CV developer team. (s.f.). *Open CV*. Obtenido de  
[http://docs.opencv.org/doc/tutorials/introduction/desktop\\_java/java\\_dev\\_intro.html](http://docs.opencv.org/doc/tutorials/introduction/desktop_java/java_dev_intro.html)
- Openframeworks. (2014). *openframeworks*. Obtenido de  
<http://openframeworks.cc/about/>
- Oracle Corporation. (2014). *MySQL.com*. Obtenido de  
<http://www.mysql.com/products/workbench/>
- Processing. (2013). *Processing*. Obtenido de <https://www.processing.org/overview/>
- Quercus Technologies. (2010). *Quercus Technologies*. Obtenido de  
[http://www.quercus.biz/newsletters/2013/030513/Quercus\\_es.html](http://www.quercus.biz/newsletters/2013/030513/Quercus_es.html)
- Quintero, M. A. (Septiembre de 2010). Obtenido de  
<http://dialnet.unirioja.es/descarga/articulo/3739212.pdf>.
- Smith, R. (2007). *An Overview of the Tesseract OCR Engine*. Obtenido de  
<https://tesseract-ocr.googlecode.com/git/doc/tesseractidar2007.pdf>
- Stanford, U. d. (2003). *Stanford Encyclopedia*. Obtenido de [stanford.edu](http://stanford.edu):  
<http://plato.stanford.edu/entries/logic-ai/>
- Tecnologies, A. (2008). *interempresas*. Obtenido de  
<http://www.interempresas.net/Plastico/Articulos/25732-La-robotica-y-los-plasticos-a-debate.html>
- Tess4j. (2014). *Tess4j*. Obtenido de <http://tess4j.sourceforge.net/>
- Universidad Simon Bolivar. (2013). *Slideshare*. Obtenido de  
<http://es.slideshare.net/ingenieriageologica1/capitulo-vi-procesamiento-digital-de-una-imagen>

# ANEXOS

## Anexo 1: Diccionario de Datos

Usuarios							
Nombre de Columna	Tipo de Datos	Llave Primaria	Campo No Nulo	Auto incremento	Llave Foránea	Tabla de Referencia	Comentario
id_usuarios	INT(11)	✓	✓	✓			Identificador del registro.
nombre	VARCHAR(45)		✓				Nombre del usuario del sistema.
contrasena	VARCHAR(45)		✓				Contraseña del usuario del sistema.
rol	VARCHAR(45)						Descripción del rol asignado al usuario del sistema.
fecha_caducidad	DATE		✓				Identificador del empleado dueño del vehículo.

Empleados							
Nombre de Columna	Tipo de Datos	Llave Primaria	Campo No Nulo	Auto incremento	Llave Foránea	Tabla de Referencia	Comentario
id_empleados	INT(11)	✓	✓	✓			Identificador del registro.
nombres	VARCHAR(150)		✓				Nombres del empleado de la institución.
apellidos	VARCHAR(150)		✓				Apellidos del empleado de la institución.
cedula	VARCHAR(13)		✓				Cédula de identidad del empleado de la institución.
fecha_nacimiento	DATE		✓				Fecha de nacimiento del empleado de la institución.
direccion	VARCHAR(150)		✓				Dirección domiciliaria del empleado de la institución.

telefono	VARCHAR(10)		✓				Número de teléfono convencional del empleado de la institución.
celular	VARCHAR(10)						Número de teléfono móvil del empleado de la institución.
email	VARCHAR(55)						Correo electrónico del empleado de la institución.
estado	TINYINT(1)		✓				Estado Activo/Inactivo del empleado de la institución.
fecha_registro			✓				Fecha en la que se creó el registro.
id_cargos			✓		✓	Registro_acceso	Referencia al cargo que tiene el empleado de la institución.

Cargos							
Nombre de Columna	Tipo de Datos	Llave Primaria	Campo No Nulo	Auto incremento	Llave Foránea	Tabla de Referencia	Comentario
id_cargos	INT(11)	✓	✓	✓			Identificador del registro.
descripcion	VARCHAR(45)		✓				Descripción del cargo.

Registro_acceso							
Nombre de Columna	Tipo de Datos	Llave Primaria	Campo No Nulo	Auto incremento	Llave Foránea	Tabla de Referencia	Comentario
id_registro_acceso	INT(11)	✓	✓	✓			Identificador del registro.
fecha_ingreso	DATE		✓				Fecha en la que se registra la incidencia.
hora_ingreso	TIME		✓				Hora en la que se registra la incidencia.

lecturaplaca	INT(11)		✓				Caracteres de la placa detectada.
observaciones	INT(11)						Datos que corresponden a la incidencia ingreso/no ingreso del vehículo.
id_tipo_acceso	INT(11)		✓		✓	Tipo_acceso	Tipo de acceso concedido al vehículo.
id_procesos	INT(11)		✓		✓	Procesos	Identificador del usuario/empleador dueño del vehículo.
id_vehiculos	INT(11)		✓		✓	Vehiculos	Identificador del usuario/empleador dueño del vehículo.

Procesos							
Nombre de Columna	Tipo de Datos	Llave Primaria	Campo No Nulo	Auto incremento	Llave Foránea	Tabla de Referencia	Comentario
id_procesos	INT(11)	✓	✓	✓			Identificador del registro.
nombres	VARCHAR(45)		✓				Descripción del tipo de captura.

Tipo_acceso							
Nombre de Columna	Tipo de Datos	Llave Primaria	Campo No Nulo	Auto incremento	Llave Foránea	Tabla de Referencia	Comentario
id_tipo_acceso	INT(11)	✓	✓	✓			Identificador del registro.
nombre	VARCHAR(50)		✓				Nombre del tipo de registro con el que se guarda la eventualidad.
descripcion	VARCHAR(150)		✓				Descripción del tipo de registro con el que se guarda la eventualidad.

Vehiculos							
Nombre de Columna	Tipo de Datos	Llave Primaria	Campo No Nulo	Auto incremento	Llave Foránea	Tabla de Referencia	Comentario
id_vehiculos	INT(11)	✓	✓	✓			Identificador del registro.
placa	VARCHAR(7)		✓				Placa del vehículo.
marca	VARCHAR(100)		✓				Marca de vehículo.
modelo	VARCHAR(100)						Modelo del vehículo.
color	VARCHAR(45)						Color del vehículo.
anio	INT(11)		✓		✓	Tipo_acceso	Año del vehículo.
observaciones	VARCHAR(150)				✓	Procesos	Observación con respecto al vehículo.

estado_fecha	DATE		✓		✓	Procesos	Estado valido/caducado para el vehículo.
id_empleados	INT(11)		✓		✓	Empleados	Identificador del empleado dueño del vehículo.

Capturas							
Nombre de Columna	Tipo de Datos	Llave Primaria	Campo No Nulo	Auto incremento	Llave Foránea	Tabla de Referencia	Comentario
id_capturas	INT(11)	✓	✓	✓			Identificador del registro.
nombre	VARCHAR(45)		✓				Nombre de la imagen de la captura
path	VARCHAR(45)		✓				Ubicación física de la captura dentro del computador.
id_registro_acceso	INT(11)		✓		✓	Registro_acceso	Referencia del registro de acceso al que corresponde la captura.

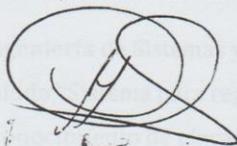


Doctora Jenny Ríos Coello, Secretaria de la Facultad de Ciencias de la Administración de la Universidad del Azuay,

**CERTIFICA:**

Que, el H. Consejo de Facultad en sesión realizada el 7 de febrero de 2013, conoció la petición de los señores **Fabián Mauricio Atariguana Quezada** (39962) y **Pedro Andrés Ávila Regalado** (42614) que denuncian su trabajo de tesis previo a la obtención del título de Ingeniero de Sistemas, con el tema: **Sistema para registro de ingreso de vehículos a la Universidad del Azuay mediante el reconocimiento de placas**". El Consejo acoge el informe de la Junta Académica y aprueba la denuncia de tesis. Designa como Director al Ingeniero Marcos Orellana Cordero y como miembros del Tribunal Examinador a los ingenieros Fabián Carvajal Vargas y Pablo Esquivel León.- De conformidad a las disposiciones reglamentarias los denunciados deberán presentar su trabajo de graduación en un plazo máximo de **DIECIOCHO MESES** contados a partir de la fecha de aprobación, esto es hasta el 7 de agosto de 2014.

Cuenca, febrero 18 de 2013



UNIVERSIDAD DEL  
AZUAY  
SECRETARIA  
IFAC ADMINISTRACION

Atentamente,

Ing. Marcos Orellana Cordero  
DIRECTOR ESCUELA DE INGENIERIA  
DE SISTEMAS Y TELEMATICA

Oficio Nro. 001-2013-DIST-UDA

Cuenca, 01 de febrero de 2013

DECANO DE LA FACULTAD DE CIENCIAS DE LA ADMINISTRACIÓN

**Señor Ingeniero**

**Oswaldo Merchán Manzano**

**DECANO DE LA FACULTAD DE CIENCIAS DE LA ADMINISTRACIÓN**

**Presente.-**

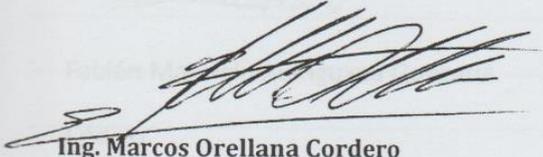
De nuestras consideraciones:

La Junta Académica de la Escuela de Ingeniería de Sistemas y Telemática, reunida el día 29 de enero de 2013, conoció el Proyecto de Tesis titulado "Sistema para registro de ingreso de vehículos a la Universidad del Azuay mediante el reconocimiento de placas.", presentada por los estudiantes Pedro Ávila y Fabián Atariguana, estudiantes de la Escuela de Ingeniería de Sistemas, previo a la obtención del título de Ingeniero de Sistemas.

La Junta considera que el diseño de tesis presenta una estructura teórica, metodológica y técnica objetiva y coherente, razón por la cual solicita, por su digno intermedio, el conocimiento y aprobación por parte del Consejo de Facultad.

Por lo expuesto, y de conformidad con el Reglamento de Graduación de la Facultad, recomienda designar como Director de Tesis al Ing. Marcos Orellana, y como miembros del Tribunal al Ing. Fabián Carvajal y el Ing. Pablo Esquivel.

Atentamente,

  
**Ing. Marcos Orellana Cordero**  
DIRECTOR ESCUELA DE INGENIERIA  
DE SISTEMAS Y TELEMATICA

Cuenca, 04 de febrero del 2013

Ingeniero

Oswaldo Merchán Manzano

DECANO DE LA FACULTAD DE CIENCIAS DE LA ADMINISTRACION.

Ciudad.

De nuestras consideraciones

Nosotros, Fabián Mauricio Atariguana Quezada con código 39962 y Pedro Andrés Ávila Regalado con código 42614, estudiantes de la escuela de Ingeniería de Sistemas de esta prestigiosa Universidad, solicitamos a usted de la forma más comedida y por su intermedio al Honorable Consejo de Facultad, la aprobación del diseño de tesis con el tema "SISTEMA PARA REGISTRO DE INGRESO DE VEHICULOS A LA UNIVERSIDAD DEL AZUAY MEDIANTE EL RECONOMIAMIENTO DE PLACAS", previo a la obtención del título de Ingeniero de Sistemas.

Nos permitimos sugerir el nombre del Ingeniero Marcos Orellana Cordero como director de tesis, puesto que he recibido asesoramiento y contamos con su aprobación.

Agradecemos de antemano la favorable acogida a la presente.

Atentamente

Fabián Mauricio Atariguana Quezada.

Código

39962

Pedro Andrés Ávila Regalado

Código

42614

Oficio Nro. 001-2013-DIST-UDA

Cuenca, 01 de febrero de 2013

**Señor Ingeniero**

**Oswaldo Merchán Manzano**

**DECANO DE LA FACULTAD DE CIENCIAS DE LA ADMINISTRACIÓN**

**Presente.-**

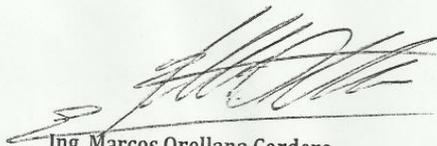
De nuestras consideraciones:

La Junta Académica de la Escuela de Ingeniería de Sistemas y Telemática, reunida el día 29 de enero de 2013, conoció el Proyecto de Tesis titulado "Sistema para registro de ingreso de vehículos a la Universidad del Azuay mediante el reconocimiento de placas.", presentada por los estudiantes Pedro Ávila y Fabián Atariguana, estudiantes de la Escuela de Ingeniería de Sistemas, previo a la obtención del título de Ingeniero de Sistemas.

La Junta considera que el diseño de tesis presenta una estructura teórica, metodológica y técnica objetiva y coherente, razón por la cual solicita, por su digno intermedio, el conocimiento y aprobación por parte del Consejo de Facultad.

Por lo expuesto, y de conformidad con el Reglamento de Graduación de la Facultad, recomienda designar como Director de Tesis al Ing. Marcos Orellana, y como miembros del Tribunal al Ing. Fabián Carvajal y el Ing. Pablo Esquivel.

Atentamente,



**Ing. Marcos Orellana Cordero**

**DIRECTOR ESCUELA DE INGENIERIA  
DE SISTEMAS Y TELEMATICA**



## Universidad del Azuay

Facultad de Ciencias de la Administración  
Escuela de Ingeniería de Sistemas

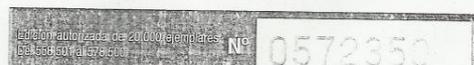
**Diseño de Tesis previo a la obtención del título de Ingeniero en Sistemas**

Título: Sistema para registro de ingreso de vehículos a la Universidad del Azuay  
mediante el reconocimiento de placas.

Autores: Pedro Ávila,  
Fabián Atariguana

Director de Tesis:  
Ing. Marcos Orellana Cordero

Cuenca, Ecuador  
2012-2013





**1. Título**

Sistema para registro de ingreso de vehículos a la Universidad del Azuay mediante el reconocimiento de placas.

**2. Antecedentes**

Actualmente existen grandes inconvenientes con los vehículos al momento de ingresar a los parqueaderos de la Universidad del Azuay comprometiendo tanto el tiempo invertido en el desarrollo de actividades como su seguridad. Diariamente se forman largas colas de espera que afectan al tráfico de las calles que lo rodean causando molestias, retrasos e inclusive accidentes.

A más de este inconveniente, la falta de un registro de actividades nos dificulta saber con exactitud el momento que ha ingresado cierto vehículo, y peor aún si éste está autorizado. Para ello es necesario la implementación de un sistema automatizado que permita un ingreso rápido, un registro de información de cada vehículo y una efectiva identificación de aquellos que forman parte del personal autorizado.

**3. Selección y delimitación del tema**

El presente trabajo de investigación se enfocará en establecer una herramienta que permita controlar el ingreso de vehículos a la Universidad del Azuay. La aplicación permitirá verificarla placa del vehículo, a través de una cámara web con herramientas informáticas de identificación y reconocimiento de patrones, y posteriormente realizar un registro de ingreso en una base de datos, lo que permitirá

posteriores consultas. Las herramientas incluirán procesos como la identificación de la placa, la digitalización de caracteres, la comprobación de placas existentes dentro de una base de datos, etc. Todos estos procesos serán previamente estudiados y explicados en su funcionamiento.

#### **4. Justificación**

Este sistema propone una solución a los problemas anteriormente mencionados, ya que aprovecharía la rapidez con la que un computador puede verificar los patrones de las placas en tiempo real y realizar su identificación en la base de datos. Esto permitirá en un menor tiempo validar si el vehículo es autorizado, y procederá a permitir el acceso a los parqueaderos de la Universidad del Azuay, evitando de esta manera la espera y la conglomeración del tráfico. El sistema no solo automatizará; también indicará la información del propietario además de su cargo y su lugar establecido en el parqueadero.

En resumen, lo indicado anteriormente permitirá reducir significativamente los tiempos de espera de los vehículos que ingresan, además de brindar un mejor control para su autorización, dando mayor comodidad para el personal sin necesidad de realizar algún trámite adicional para el ingreso.

#### **5. Descripción del objeto de estudio**

La investigación se enfoca en presentar la tecnología de visión artificial y las herramientas que conformarán el programa para su funcionamiento.



UNIVERSIDAD DEL AZUAY

Como aplicación a un problema real se tomará el caso del ingreso de vehículos al parqueadero de la Universidad del Azuay y se aspirará a solucionar los problemas de ingreso que actualmente se presentan.

Se diseñarán los procesos necesarios para la aplicación de cada herramienta con su respectivo flujo de información. El sistema será automático, se activará el momento de detectar la placa que esté al alcance de la cámara y tomará en consideración un tiempo prudente para que el auto ingrese.

En cuanto al software, su interfaz será simple e intuitiva, ya que las personas encargadas de un parqueadero generalmente pueden presentar un nivel bajo de conocimientos informáticos, por lo que es necesario que sea entendible y así pueda ser controlado sin inconvenientes.

Para el uso de estas herramientas será necesario tener 3 componentes básicos:

Monitor del computador: instrumento donde se verá reflejado la identificación del patrón así como la conversión del mismo a caracteres.

Cámara Web: dispositivo que toma la información del mundo real y la transmite al software de realidad aumentada.

Software: programa encargado de incorporar las dos tecnologías para interpretarlas y así lograr mostrar un producto final.

**6. Objetivos**

Esta investigación busca cumplir los siguientes objetivos:

**General**

Desarrollar un Sistema de información para el registro de automóviles mediante la detección, digitalización y validación de su placa, aplicando tecnologías de visión artificial e implementando una solución para el ingreso de vehículos en el parqueadero de la Universidad del Azuay.

**Específicos**

- Justificar adecuadamente las tecnologías de reconocimiento y digitalización de patrones utilizadas para el desarrollo de este sistema.
- Capturar y sintetizar la información requerida para su procesamiento con las herramientas seleccionadas.
- Desarrollar el sistema de información para el registro del ingreso de vehículos.
- Realizar una verificación de resultados al finalizar cada procedimiento para así determinar la integridad del sistema.
- Elaborar un manual de uso para el usuario final.



## 7. Metodología

Se fundamentó teóricamente la propuesta de mantener una metodología histórica ya que la evolución e implementación de la tecnología nos surge del resultado de la investigación de la visión artificial, tanto en medios digitales como en medios físicos; es decir una revisión bibliográfica, referenciada por indicadores como son:

- JasVsio.Com Visión Artificial (2010)
- Ni.com Vision Concepts Manual (2005).

Esta propuesta también implementará una metodología lógica que nos permitirá delimitar a lo relacionado con el tema de tratamiento de imágenes; es decir la visión artificial dentro de sus ramas específicas como son el reconocimiento de patrones, la digitalización de caracteres así como el tratamiento de imágenes propiamente dicho, sobre lo cual mantendremos de igual manera una revisión bibliográfica de las fuentes a disposición bajo los indicadores siguientes:

- CherkasovAlek OCR (2004),
- Meyer Ernersto Reconocimiento de Patrones (2012),
- Duda, Hart, Stock Pattern Classification (2001).

Aplicando el método inductivo nos permitiremos plantear ciertos puntos de vista sobre el cual evaluar los resultados que vamos obteniendo, los mismos que responderán las siguientes preguntas, orientadas a elaborar un planteamiento de la problemática a ser resuelta, dichas interrogantes son las siguientes:

- ¿Cuál es el principal problema al ingresar vehículos en la UDA?,
- ¿En que esto afecta a las actividades de la misma?,

- ¿Cómo podemos solucionarlo?,
- ¿Qué herramientas podemos usar?

Dichos indicadores serán resultado de entrevistas con la finalidad de levantar información que nos oriente de mejor manera hacia una solución que satisfaga las necesidades de los usuarios finales.

Estos resultados nos definirán hasta qué punto podrá ayudar este proyecto a mejorar la situación actual, y bajo qué términos podremos aportar una ventaja al resolver los problemas que se presenten, como es la acumulación de tráfico que presenta consecuencias como pérdidas de tiempo que acaban perjudicando de cierta forma a usuarios del parqueadero. Estas soluciones las iremos analizando mediante un método deductivo que será un análisis de las respuestas obtenidas en la entrevista; para de esta manera poder determinar las falencias del sistema actual y estimar las medidas correctivas y aportaciones que presente el nuevo sistema de ingreso vehicular utilizando la tecnología de la visión artificial.

Diagnosticada la realidad referente utilizaremos una metodología experimental bajo la cual se ha podido comprobar que el proyecto es viable en el tiempo. Por su estructura esta tecnología representa un gran avance en la tecnología de la información y tratamiento de la misma, ya que combina el entorno real con lo digital y nos permite interactuar con él. Actualmente esta tecnología comienza a implementarse en móviles, automóviles y demás dispositivos actuales. Por lo que con el tiempo este sistema de reconocimiento de placas será fundamental en todo acceso a cualquier institución.

Como principales instrumentos para el desarrollo de este diagnóstico tenemos: La evaluación de las herramientas, así como un estudio completo de su funcionalidad.



Conocido como los indicadores de esta etapa la referencia a anexo o bibliografía, así como también una serie de pruebas como son las de reconocimiento de patrones, conjuntamente con las herramientas OCR y las que son encargadas del procesamiento de imágenes.

Tras las pruebas; las valoraciones a tener en cuenta serán estadísticas debido a que nos permitirá denotar los principales problemas que pueden causar la falta de un sistema automatizado de ingreso de vehículos, esto de igual manera lo podremos llevar a cabo mediante encuestas y entrevistas a los principales implicados que será el personal de la Universidad del Azuay que son los mayores conocedores de la problemática a ser respondido y resuelta.

Es por lo anteriormente especificado que se ha propuesto una metodología teórica con las siguientes características; modelico, casuístico, debido que esta tecnología se acoplará de acuerdo al caso que se esté estudiando, por lo que este se ajusta al problema de ingreso de vehículos, cuya solución es el uso de la visión artificial, puntalmente el reconocimiento de patrones y digitalización de caracteres para identificar un vehículo por su placa. Para lo que tendremos que analizar el caso con la ayuda de herramientas de diseño del sistema como pueden ser la creación de modelos UML, planteamiento de escenarios, diagramas de actividad, diagramas entidad relación, diagramas de caso de uso, etc.

## 8. Marco teórico

Mediante la tecnología de la visión artificial conocida también como la visión por computador o visión técnica, que corresponde a una rama de la inteligencia artificial, nos permitimos el reconocimiento de cierto tipo de patrones dentro de una imagen o un entorno capturado.

El propósito de la visión artificial es el entrenamiento o programación vía computadora, para que la misma entienda o identifique ciertas características dentro de una imagen.

La visión artificial la componen un conjunto de procesos destinados a realizar el análisis de imágenes. Estos procesos son: captación de imágenes, memorización de la información, procesado e interpretación de los resultados.

Dentro de las características primordiales que tiene la visión artificial se pueden destacar los siguientes:

- La detección de cierto patrón o imagen dentro de otra imagen o dentro de un entorno presentado para su evaluación, segmentación del tipo de patrón a ser reconocido o identificado tomando muestras positivas y negativas sobre las cuales basamos para el entrenamiento, localización del patrón y reconocimiento de ciertos objetos en imágenes.
- Registro de diferentes imágenes de una misma escena u objeto, es decir, hacer concordar un mismo objeto en diversas imágenes, que son la toma de muestras para ejecutar el entrenamiento.



Tomando en cuenta estas características el entorno de visión artificial nos permite realizar una gran cantidad de operaciones como por ejemplo:

- Automatizar tareas repetitivas de inspección realizadas por operadores.
- Realizar controles de calidad de productos que no era posible verificar por métodos tradicionales.
- Realizar inspecciones de objetos sin contacto físico.
- Realizar la inspección del 100% de la producción (calidad total) a gran velocidad.
- Reducir el tiempo de ciclo en procesos automatizados.
- Realizar inspecciones en procesos donde existe diversidad de piezas concambios frecuentes de producción.

Su funcionamiento está especificado por 2 pasos: Segmentación y Análisis.

La segmentación consiste en algoritmos de software que se encarga de decidir qué partes de la imagen necesitan interpretación o análisis y cuáles no. A menudo es posible y necesario refinar la segmentación. Por ejemplo, si la aplicación debe encontrar arañazos o grietas en un vidrio, la primera fase de segmentación suele encontrar elementos que son las propias grietas o arañazos además de suciedad y polvo. A menudo se perfecciona este resultado para eliminar de la imagen la suciedad y el polvo de los elementos a analizar.

Una vez que la imagen ha sido satisfactoriamente segmentada, se procede al análisis, por lo que el procesador realizará una serie de pruebas y mediciones en los elementos de interés que aparecen en la escena. Existen multitud de algoritmos que nos

ayudan a obtener los resultados deseados. Entre ellos, cabe destacar sus funcionalidades:

- Detección de formas.
- Análisis geométrico.
- Comparación con patrones.
- Medición de objetos.
- Detección de objetos difusos.

Estos algoritmos, usados estratégicamente, nos permiten decidir si una pieza se ha fabricado correctamente o no, guiar un robot hacia un objeto concreto, reajustar la maquinaria de producción automáticamente o identificar y validar la información de una placa.

Como resultado del entrenamiento para el reconocimiento de patrones u objetos proporcionados por ejemplo por una librería de tratamiento de imágenes, destinada principalmente a aplicaciones de visión por computador en tiempo real; se pretende realizar el reconocimiento del "patrón" en este caso el de la placa vehicular. Entre las herramientas permitirán trabajar con la visión artificial es OpenCv.

OpenCV es una biblioteca libre de visión artificial originalmente desarrollada por Intel. Su funcionalidad abarca grandes proyectos desde sistemas de seguridad con detección de movimiento, hasta aplicativos de control de procesos, donde se requiere reconocimiento de objetos. Su publicación se da bajo licencia BSD, que permite que sea usada libremente. Es una herramienta muy completa que contiene más de 500 funciones que abarcan una gran gama de áreas en el proceso de visión, como reconocimiento de objetos (reconocimiento facial), calibración de cámaras, visión estéreo y visión robótica.



El desarrollo del reconocimiento de patrones mediante la visión artificial nos da gran ventaja el momento de reconocer nuestra placa vehicular, sin embargo es necesario leer los caracteres que contiene la placa por lo que usaremos un OCR.

OCR o Reconocimiento Óptico de Caracteres, no es más que un proceso que nos permite digitalizar textos o caracteres que forman parte de un alfabeto, los mismos que son identificados dentro de una imagen o de un patrón, para luego ser almacenados en forma de datos.

Un OCR cuenta con sus propios lenguajes de identificación e interpretación de caracteres, pero si es necesario, también permite la implementación de nuevos diccionarios que deberán ser añadidos a manera de entrenamiento, permitiendo la facilidad del reconocimiento y mejorando la interpretación de los datos. Entre las herramientas de OCR tenemos Tesseract que es de software libre y muy completo.

Tesseract es uno de los motores más precisos de OCR, puede leer una gran variedad de formatos de imagen y los convierte a texto en más de 60 idiomas. Fue uno de los 3 mejores motores en la prueba de precisión 1995. Fue liberado por HP y luego fue acogido por Google que realizó grandes mejoras, siendo este usado por sus múltiples aplicaciones.

Para un desarrollo óptimo y una integración total de estas dos tecnologías utilizaremos la herramienta de procesamiento de imágenes denominada Processing.

Processing es un lenguaje de programación visual de código abierto basado en java, es una herramienta ideal para trabajar de una manera fácil en el procesamiento de imágenes tanto 2D como 3D, así como animaciones muy complejas.

## 9. Esquema tentativo

Abstract

Introducción

Objetivos

Metodología

Capítulo 1: Visión Artificial

1.1. Conceptos Básicos

1.1.1. Definición

1.1.2. Clasificación

1.1.3. Aplicaciones

1.2. Funcionamiento

1.2.1. Elementos Requeridos

1.2.2. Estructura General

1.2.3. Entrenamiento y detección

1.3. Futuro en el ámbito de procesamiento de patrones

Capítulo 2: Análisis de la situación actual

2.1 Estudio de la problemática actual.

2.1.1 Causas del problema

2.1.2 Efectos del problema.

2.1.2.1 Principales afectados

2.2 Delimitación del problema.



Capítulo 3: Estudio de Herramientas de visión artificial.

3.1. Soluciones propuestas a los problemas mencionados.

3.2. Herramientas seleccionadas para el sistema.

3.2.1. Herramientas de reconocimiento de patrones.

3.2.1.1. Características.

3.2.1.2. Aplicación en el sistema.

3.2.1.3. Requerimientos.

3.2.2. Herramienta de digitalización de caracteres (OCR)

3.2.2.1. Características.

3.2.2.2. Aplicación en el sistema.

3.2.2.3. Requerimientos.

3.2.3. Herramienta de procesamiento de imágenes.

3.2.3.1. Características.

3.2.3.2. Aplicación en el sistema.

3.2.3.3. Requerimientos.

Capítulo 4: Análisis y diseño del sistema.

4.1. Recolección de información para el sistema.

4.1.1. Investigación de características de un sistema de seguridad.

4.1.1.1. Estudios de sistemas existentes.

4.1.1.2. Investigación de características a implementar

4.1.2. Obtención de elementos requeridos de cada herramienta definida

4.1.2.1. Obtención de imágenes.

4.1.2.2. Definición de estructura de imágenes.

4.1.2.2.1. Clasificación

4.1.2.2.2. Numeración

4.1.2.2.3. Posición interna

4.1.2.2.4. Ubicación.

4.2. Determinación de los requerimientos básicos de hardware

4.2.1. Equipos necesarios para desarrollo y pruebas

4.3. Identificación de los procesos que intervienen en el sistema

4.3.1. Diagrama de Casos de Uso

4.3.2. Diagrama de Actividades

4.3.3. Diagrama de Clases

4.4. Determinación los flujos de datos

4.5. Determinación la información para los almacenes de datos

4.6. Diseño de la base de datos

4.7. Definición de procedimientos en cada herramienta

4.7.1. Elementos de entrada

4.7.2. Procedimientos a realizar

4.7.2.1.Enumeración de pasos a seguir

4.7.2.2.Observaciones

4.7.2.3.Resultados Esperados

Capítulo 5: Desarrollo del Sistema

5.1. Herramientas de programación.

5.1.1. Gestor de Bases de Datos

5.1.2. Lenguajes de programación.

5.2. Evaluación y Selección de Componentes

5.2.1. Componentes externos necesarios

5.2.2. Evaluación de Componentes

5.3. Desarrollo del Sistema



5.3.1. Fase de Codificación

5.3.2. Desarrollo y Monitoreo de Progresos en el Sistema

5.3.3. Integración de resultados.

5.3.4. Resultados.

5.3.4.1. Resultados Obtenidos

5.3.4.2. Depuración de resultados.

5.3.4.3. Comprobación de resultados.

Capítulo 6: Pruebas y Simulaciones

6.2. Pruebas de Funcionamiento

6.2.1. Comprobación por herramienta

6.3. Pruebas de Rendimiento

6.4. Simulaciones del Sistema Completo.

Capítulo 7: Manual del Usuario

Conclusiones y recomendaciones

## 10 Bibliografía

### Libros

- Bradski, Gary y Adrian Kaehler. Learning OpenCV. Primera. Sebastopol: O'Really, 2008.
- Duda, Richard O; Hart E. Peter y David G. Stork. Pattern classification. New York: Wiley, 2001.
- Mohamed Cheriet, Nawwaf Kharma, Cheng-Lin Liu, Ching Y. Suen. Character recognition Systems: A Guide for Students and Practitioners. Primera. Hoboken: Wiley-Interscience , 2007.

### Fuentes en Internet

- Anxel. Reconocimiento de patrones con OpenCV. 20 de Enero de 2012. 18 de Octubre de 2012  
<<http://ifnotisnull.blogspot.com/2012/01/reconocimiento-de-patrones-con-opencv.html>>.
- Bradski, Gary y Adrian Kaehler. Learning OpenCV. Primera. Sebastopol: O'Really, 2008.
- Cherkasov, Alex. Creating Optical Character Recognition (OCR) applications using Neural Networks. 1 de Septiembre de 2004. Octubre de 15 de 2012  
<<http://www.codeproject.com/Articles/3907/Creating-Optical-Character-Recognition-OCR-applica>>.
- Daniel, Leidert. OpenCV create samples. 17 de Mayo de 2008. 17 de Octubre de 2012 <<http://www.makelinux.net/man/1/O/opencv-createsamples>>.
- Duda, Richard O, Hart E. Peter y David G. Stork. Pattern classification. New York: Wiley, 2001.
- Entrenamiento OPENCV. 8 de Mayo de 2010 . 18 de Octubre de 2012  
<<http://coplec.org/2010/08/05/entrenar-opencv>>.
- JasVisio. Vision Artificial. 2010. 11 de Enero de 2013  
<<http://www.jasvisio.com/aplicaciones-vision-artificial-industria.html>>.
- Meyer, Ernesto A. Reconocimiento de patrones. 11 de Septiembre de 2012. 15 de Octubre de 2012  
<[http://es.wikipedia.org/wiki/Reconocimiento\\_de\\_patrones](http://es.wikipedia.org/wiki/Reconocimiento_de_patrones)>.
- Mohamed Cheriet, Nawwaf Kharma, Cheng-Lin Liu, Ching Y. Suen. Character recognition Systems: A Guide for Students and Practitioners. Primera. Hoboken: Wiley-Interscience , 2007.
- National Instruments. «IMAQ Vision Concepts Manual.» 2005. NI Vision. 11 de Enero de 2012 <ni.com>.
- Natoshi, Seo: «OpenCV haartraining.» 16 de Octubre de 2008. sonots.com: 18 de Octubre de 2012  
<<http://note.sonots.com/SciSoftware/haartraining.html>>.
- Open Dev. Team. OPENCV Installation in Windows. 14 de Febrero de 2011. 17 de Octubre de 2012



<[http://docs.opencv.org/doc/tutorials/introduction/windows\\_install/windows\\_install.html](http://docs.opencv.org/doc/tutorials/introduction/windows_install/windows_install.html)>.

Panichi. «Entrenamiento OPENCV.» 05 de Agosto de 2010. coplec.org. 18 de Octubre de 2012 <<http://coplec.org/2010/08/05/entrenar-opencv>>.

Reas, Casey y Ben Fry. Tutoriales Processing. 2012 de Septiembre de 2012. 20 de Octubre de 2012

<<http://www.anonimocolectivo.org/taller/tutoriales/Processing/>>.

Team, Google. Training Tesseract. 16 de Septiembre de 2010. 17 de Octubre de 2012 <<http://code.google.com/p/tesseract-ocr/wiki/TrainingTesseract3>>.

Viola, Paul y Michael Jones. Cascade Classifier Training. 3 de Noviembre de 2007. 15 de Octubre de 2012

<[http://docs.opencv.org/doc/user\\_guide/ug\\_traincascade.html](http://docs.opencv.org/doc/user_guide/ug_traincascade.html)>.

