



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
Facultad de Ingeniería en Electricidad y Computación

“EXTRACCIÓN Y EVALUACIÓN DE CARACTERÍSTICAS DE SECUENCIAS
DE VIDEO UTILIZANDO MÉTODOS ORIENTADOS A DISPOSITIVOS DE
BAJO PODER COMPUTACIONAL”

INFORME DE MATERIA DE GRADUACIÓN

Previa a la obtención del Título de:
**INGENIERO EN COMPUTACIÓN ESPECIALIZACIÓN
SISTEMAS TECNOLÓGICOS**

Presentado por:
Héctor José Astudillo Pino

GUAYAQUIL - ECUADOR

AÑO: 2015

AGRADECIMIENTO

Agradezco a Dios por ayudarme a esperar en El, por proveerme de fuerzas, salud, sabiduría y por rodearme de personas que directa o indirectamente contribuyeron para ayudarme a obtener este logro.

DEDICATORIA

Dedico este trabajo primeramente a Dios por su amor y sus bondades que son nuevas cada mañana, a mi familia por su constante e incondicional apoyo, a mis maestros por sus conocimientos y experiencias compartidas, y a mis compañeros por los momentos gratos vividos en las aulas de clases. Dios los bendiga a todos ustedes.

TRIBUNAL DE SUSTENTACIÓN

Ph.D. DENNIS GUILLERMO ROMERO LÓPEZ
PROFESOR DE LA MATERIA DE GRADUACIÓN

Ph.D. BORIS XAVIER VINTIMILLA BURGOS
PROFESOR DELEGADO POR LA UNIDAD ACADÉMICA

DECLARACIÓN EXPRESA

“La responsabilidad por los hechos, ideas y doctrinas expuestas en este informe me corresponde exclusivamente; y, el patrimonio intelectual de la misma, a la ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL”

(Reglamento de Graduación de la ESPOL).

Héctor José Astudillo Pino

RESUMEN

El presente trabajo describe el desarrollo de un sistema de control de acceso vehicular utilizando un dispositivo BeagleBone, dentro de la temática del proyecto titulado “Extracción y evaluación de características de secuencias de video utilizando métodos orientados a dispositivos de bajo poder computacional.”. Este sistema permite la entrada de un vehículo mediante la identificación de una placa autorizada, esto activará una puerta de garaje motorizada que será automáticamente cerrada cuando el vehículo haya ingresado, mediante la medición de distancia del vehículo a través de sensores ultrasónicos.

Esto representa una ayuda para el dueño del vehículo debido a que evita salir del automóvil para abrir una puerta o depender de un control remoto para la activación de una puerta motorizada. Los detalles acerca de la integración de sensores en un dispositivo de bajo costo computacional (BeagleBone) son mostrados en este documento.

ÍNDICE GENERAL

RESUMEN	vi
ÍNDICE GENERAL	vii
ÍNDICE DE TABLAS	x
ÍNDICE DE FIGURAS	xi
INTRODUCCIÓN	xiii
1 Descripción y Justificación	1
1.1 Descripción del problema	1
1.2 Justificación del proyecto	2
1.3 Solución propuesta	3
1.4 Objetivos	4
1.4.1 Objetivo General	4
1.4.2 Objetivos Específicos	5
1.5 Organización del documento	6
2 Revisión Bibliográfica	8

2.1	Beaglebone Black Rev C	8
2.1.1	Beneficios	9
2.1.2	Características Generales	10
2.2	Sensor de Proximidad Ultrasónico HC-SR04	13
2.2.1	Descripción	13
2.2.2	Características	15
2.3	Reconocimiento Óptico de Caracteres (OCR)	16
2.3.1	Reconocimiento de Patrones OCR	16
2.4	Procesamiento de la Imagen de la Placa	18
2.4.1	Captura	21
2.4.2	Preproceso	21
2.4.3	Binarización	22
2.4.4	Análisis de Componentes Conectados	24
2.4.5	Segmentación	25
2.4.6	Reconocimiento de Caracteres	26
2.5	Herramientas para el desarrollo del sistema	26
2.5.1	Python	26
2.5.2	OpenCV	27
2.5.3	Tesseract	28
3	Instalación y Configuración de Hardware y Software	29
3.1	Introducción	29
3.2	Configuración del Motor 28BYJ-48	30

3.3	Configuración del Sensor de Proximidad HC-SR04	31
3.4	Configuración de la Cámara	33
3.5	Instalación de OpenCV y Tesseract	33
4	Sistema de Control de Acceso Vehicular Utilizando Beaglebone Black	36
4.1	Sistema ANPR	36
4.2	Módulos del Sistema	38
4.2.1	Módulo de Captura de la Placa del Vehículo	38
4.2.2	Módulo de Lectura del Sensor de Proximidad HC-SR04	39
4.2.3	Módulo de Apertura y Cerrado de Puerta	40
5	Análisis de Resultados	42
5.1	Resultados	42
	CONCLUSIONES Y RECOMENDACIONES	48
	BIBLIOGRAFÍA	51

ÍNDICE DE TABLAS

Tabla 2.1	Comparación de motores OCR	16
Tabla 2.2	Letras de las placas según el servicio	19
Tabla 2.3	Tipos de placas por servicios	20
Tabla 3.1	Configuración del motor 28BYJ-48.	31
Tabla 3.2	Configuración del sensor HC-SR04.	31

ÍNDICE DE FIGURAS

Fig. 1.1	Esquema reconocimiento de placa vehicular	2
Fig. 1.2	Ilustración de la solución	4
Fig. 2.1	Beaglebone Black	8
Fig. 2.2	Vista frontal y posterior del sensor ultrasónico HC-SR04 . . .	13
Fig. 2.3	Funcionamiento sensor HC-SR04	15
Fig. 2.4	Matrícula de vehículos en Ecuador	18
Fig. 2.5	Procesamiento de imagen para reconocimiento de caracteres	20
Fig. 2.6	Captura de la imagen de la placa [1]	21
Fig. 2.7	Localización de la placa [1]	23
Fig. 2.8	Componentes conectados [1]	24
Fig. 2.9	Segmentación [1]	25
Fig. 2.10	Regiones segmentadas [1]	25
Fig. 3.1	Señales de expansión de la tarjeta Beaglebone	30
Fig. 3.2	Conexión entre el HC-SR04 y el Beaglebone	32
Fig. 3.3	Conexión de una cámara USB y el Beaglebone	33
Fig. 4.1	Esquema ANPR	37

Fig. 4.2	Captura de la imagen de la placa del vehículo	39
Fig. 4.3	Proximidad del vehículo dentro del garaje	40
Fig. 4.4	Motor 28BYJ-48	40
Fig. 4.5	Organigrama del sistema	41
Fig. 5.1	Placa detectada en tiempo real.	44
Fig. 5.2	Resultados de las lecturas de distancia que se obtuvieron con el sensor de proximidad.	44
Fig. 5.3	Resultados de la captura de la placa del vehículo.	45
Fig. 5.4	Sistema integrado del proyecto desarrollado.	46
Fig. 5.5	Estructura donde estarán colocados cada uno de los com- ponentes del proyecto.	47

INTRODUCCIÓN

Con el pasar de los años y los grandes avances que se han dado en el desarrollo de nuevas tecnologías, el procesamiento de imágenes se ha convertido en un tema de gran relevancia en beneficio de la sociedad. Adicionalmente, posee un enorme valor científico-técnico por los innumerables campos en los que puede ser aplicado. Las ventajas del procesamiento digital de imágenes radican en la gran cantidad de información que puede ser obtenida, manipulada y procesada, pudiendo ser aplicada en diferentes campos como la industria, seguridad, etc.

El tratamiento digital de imágenes contempla el procesamiento y análisis de señales visuales y está relacionada con la transformación, restauración y mejoramiento de imágenes. El análisis consiste en la extracción de características de las imágenes, así como la clasificación, identificación y reconocimiento de patrones.

Considerando los adelantos tecnológicos actuales y la creciente popularidad del uso de técnicas de procesamiento de imágenes, el presente trabajo propone una solución de bajo costo, eficiente y de aplicación práctica, mediante el desarrollo de un sistema que controle el acceso vehicular a partir del reconocimiento de la placa de vehículos e integración de sensores ultrasónicos y motores de paso para la apertura de puertas. El presente trabajo comprende

tres módulos, uno de ellos se encarga de capturar y procesar la imagen de la placa del vehículo, el siguiente módulo, se encarga del control del motor que permite abrir y cerrar la puerta del garaje y finalmente el módulo de los sensores de proximidad que ayuda a detectar la distancia del vehículo, tanto fuera o dentro del garaje.

CAPÍTULO 1

1. Descripción y Justificación

1.1. Descripción del problema

En la actualidad las técnicas de procesamiento y análisis de imágenes han adquirido gran importancia en los sistemas de seguridad, monitoreo y reconocimiento de patrones, usándolo esencialmente en el control de acceso vehicular a entidades y establecimientos en general.

Una de las características únicas en un vehículo es su placa, la cual relaciona propiedades relevantes del mismo e información sobre su dueño. Por tal motivo, el reconocimiento visual de la placa de un vehículo es uno de los temas abordados en este trabajo.

En el presente proyecto se plantea el tema “*SISTEMA DE CONTROL DE ACCESO VEHICULAR UTILIZANDO BEAGLEBONE BLACK*”. Este proyecto consiste en la integración de hardware y software para la captura y procesamiento de imágenes de placas de vehículos, utilizando técnicas de segmentación y reconocimiento óptico de caracteres (OCR). Una vez reconocida la placa del vehículo se procede a accionar un motor que abrirá y cerrará la puerta de un garaje junto con la activación de un sensor HC-SR04, que indicará si el vehículo ha ingresado en su totalidad.



Figura 1.1: Esquema reconocimiento de placa vehicular

1.2. Justificación del proyecto

El propósito de la implementación de un sistema de reconocimiento de placas junto con la integración de un sensor, un motor, una cámara y un dispositivo de bajo costo (Beaglebone), es automatizar la apertura

y cierre de la puerta de un garaje, facilitando el ingreso de un vehículo sin que su dueño descienda del mismo.

Considerando además que, las técnicas y herramientas de software libre que se utilizan en este sistema, permiten tener un mayor control del mismo, facilitando la obtención de los resultados esperados.

Cabe resaltar que, el reconocimiento de placas vehiculares ha sido utilizado en diversas aplicaciones, tales como:

- Control de acceso a parqueaderos.
- Control de peaje.
- Tráfico vehicular.
- Sistemas de seguridad aeroportuaria, entre otros.

Las aplicaciones mencionadas han sido también integradas con otros dispositivos, lo cual demuestra la relevancia de la temática como un aporte a la sociedad y la ciudadanía en general.

1.3. Solución propuesta

Para el control de acceso vehicular propuesto en este proyecto, se utilizará una cámara USB para la captura de imágenes de la placa del

vehículo. Mediante un proceso de filtrado y segmentación se obtiene el número de la placa con el cual, una vez validado, se enviarán las señales respectivas desde la tarjeta Beaglebone al módulo que accionará el mecanismo que abrirá la puerta del garaje al que se desea tener acceso. De forma paralela, será activado un sensor de proximidad que indicará que el vehículo ha ingresado completamente, para proceder al cerrado automático de la puerta.

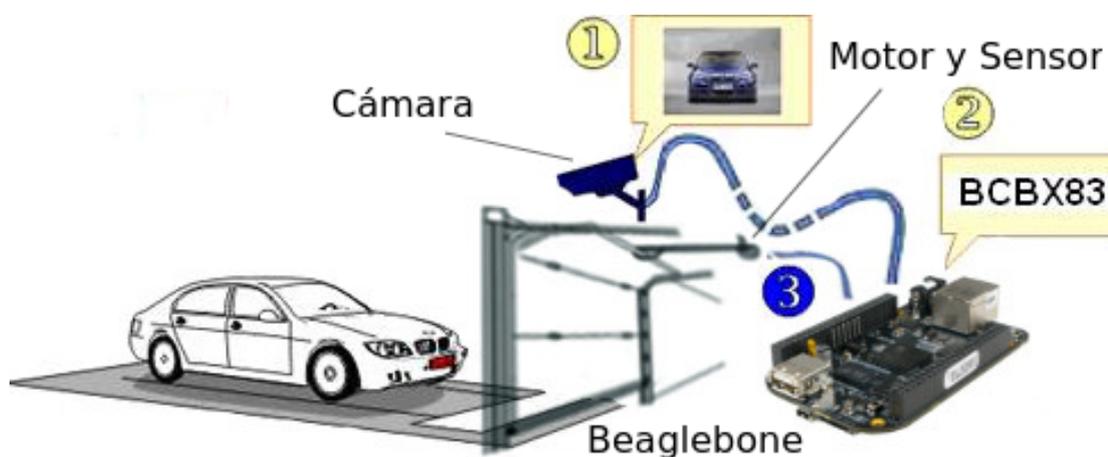


Figura 1.2: Ilustración de la solución

1.4. Objetivos

1.4.1. Objetivo General

Los objetivos generales del presente trabajo son los siguientes:

1. Desarrollar un sistema que permita detectar y segmentar la placa de un vehículo a partir de una imagen dada.
2. Implementar la integración de un sensor de proximidad y un motor de paso en un dispositivo Beaglebone.
3. Integrar los módulos de detección de placa y sensores en un único sistema de reconocimiento de placa.

1.4.2. Objetivos Específicos

Los objetivos específicos del proyecto son los siguientes:

1. Capturar la imagen previamente obtenida por medio de una cámara.
2. Seleccionar y extraer la placa del vehículo.
3. Segmentar la imagen en cada uno de sus dígitos mediante el método de crecimiento de regiones.
4. Realizar filtrado a la imagen segmentada tanto por área y tamaño.
5. Implementar la captura de datos de un sensor de proximidad en un dispositivo BeagleBone Black.
6. Realizar evaluación y comparación de los resultados obtenidos.

1.5. Organización del documento

En esta sección del documento describimos la organización del mismo, indicando de manera resumida el contenido de cada uno de los capítulos que lo componen.

- El capítulo 1 detalla el problema a abordar y la justificación de la solución propuesta, como también se establecen las condiciones generales y requerimientos necesarios para el diseño del proyecto.
- El capítulo 2 presenta el análisis y sustento bibliográfico correspondientes al sensor de proximidad HC-SR04, motor de paso, tarjeta Beaglebone y herramientas utilizadas para el desarrollo del presente trabajo.
- En el capítulo 3 se explica la configuración del hardware y software para el correcto funcionamiento de nuestro sistema.
- El capítulo 4 describe cada uno de los módulos que conforman el sistema de detección de la placa utilizando Beaglebone Black.
- En el capítulo 5 se realiza la evaluación de los resultados obtenidos luego de implementar el presente proyecto.
- Las conclusiones y recomendaciones que se generaron luego de las pruebas realizadas y experiencias que tuvimos durante la

implementación e integración de cada uno de los módulos con el dispositivo Beaglebone.

CAPÍTULO 2

2. Revisión Bibliográfica

2.1. Beaglebone Black Rev C

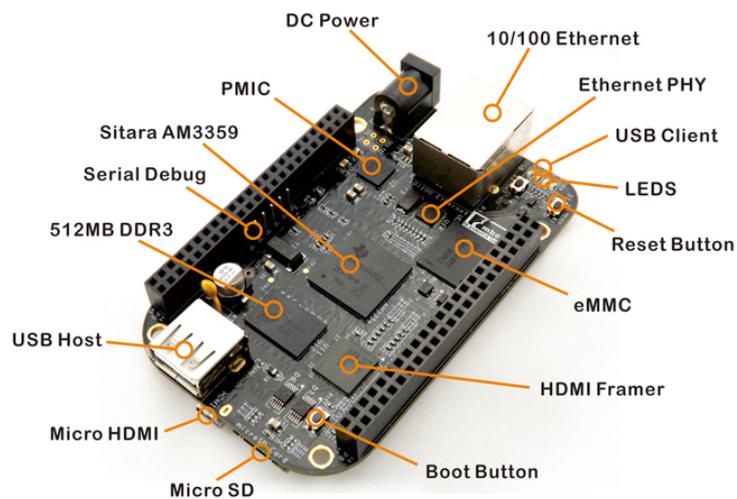


Figura 2.1: Beaglebone Black

Beaglebone Black es una tarjeta producida por *Texas Instruments (TI)* en asociación con *Digi-Key* y *Newark element14*, también cuenta con el mejor valor de desarrollo en código abierto para desarrolladores e ingenieros. Ofrece un hardware esqueleto con acceso a interfáz de señales para sensores y controles, mientras que elimina la necesidad de equipos adicionales.

2.1.1. Beneficios

A continuación se presentan algunos de los beneficios del dispositivo Beaglebone:

- Se ejecuta el sistema operativo Linux completo con servidores web, compiladores nativos y lenguajes de scripting, bibliotecas de análisis de vídeo y mucho más.
- Permite a los desarrolladores añadir fácilmente conexiones de hardware de expansión o directamente un protoboard a innumerables periféricos fácilmente disponibles.
- Los desarrolladores pueden conectar un solo cable de alimentación, depuración y la interfaz de sus aplicaciones y no necesitar un emulador JTAG, ahorrando costo adicional.

- Logra gráficos pixel-shaded o foto-realistas, en tiempo real para juegos debido a su acelerador gráfico 3D.
- Puede ser utilizado para periféricos USB como teclado, mouse, WiFi, Bluetooth, cámaras web o hubs USB para la expansión adicional a través del chip PHY USB.
- Añade varios gigabytes de almacenamiento para la imagen de arranque y datos.

2.1.2. Características Generales

El Hardware:

- Procesador: Sitara AM3359AZCZ100 1GHz, 2000 MIPS
- Procesador Gráfico: SGX530 3D, 20M Polygons/S
- Memoria SDRAM: 512MB DDR3L 606MHZ
- Flash en la Placa: 2GB, 8bit Embedded MMC
- PMIC: TPS65217C PMIC regulador y un LDO adicional.
- Soporte Para Depuración: CTI JTAG de 20 pines opcional en la placa, Cabezal Serial

- Fuente de Alimentación: miniUSB USB o Jack DC, 5VDC Externos via Cabezal de Expansión
- LEDs Indicadores: 1-Alimentación, 2-Ethernet, 4-LEDs de Usuario
- HiSpeed USB 2.0 Client Port: Acceso a USB0, Client mode via miniUSB
- HiSpeed USB 2.0 Host Port Acceso a USB1, Socket Type A, 500mA LS/FS/HS
- Acceso a Puerto Serial UART0 via Cabezal de 6 pin 3.3V TTL.
- Ethernet 10/100, RJ45
- Conector SD/MMC microSD , 3.3V
- Entradas de Usuario: Botón de Reset, Botón de Arranque (Boot), Botón de Activación (Power)
- Salida de Video: 16b HDMI, 1280x1024 (MAX), 1024x768,1280x720,1440x900 con soporte EDID
- Audio Mediante HDMI, Stereo
- Alimentación 5V, 3.3V , VDD ADC(1.8V)
- E/S 3.3V en todas las señales

- McASP0, SPI1, I2C
- GPIO(65)
- LCD
- GPMC, MMC1, MMC2
- 7 AIN(1.8V MAX)
- 4 Temporizadores
- 3 Puertos Seriales
- EHRPWM(0,2)
- XDMA Interrupt, botón de encendido
- Tarjeta de Expansión ID (Hasta 4 pueden ser acoplados).
- Peso 1.4 oz (39.80 grams)

En cuanto al **Software**, Beaglebone corre varios sistemas operativos:

- Angstrom Linux.
- Android.
- Ubuntu.

- Cloud9 IDE en Node.js con librería BoneScript.
- Otros.

2.2. Sensor de Proximidad Ultrasónico HC-SR04

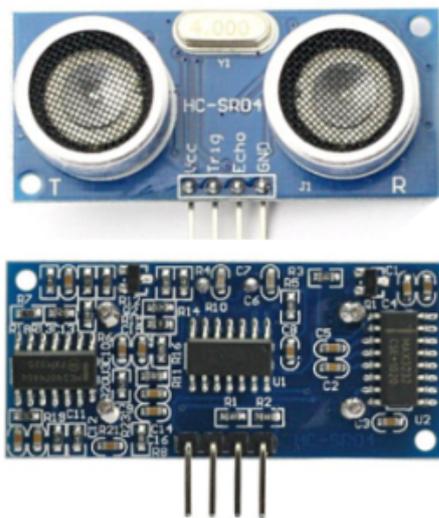


Figura 2.2: Vista frontal y posterior del sensor ultrasónico HC-SR04

2.2.1. Descripción

El sensor de distancia ultrasónico HC-SR04 tiene la capacidad de sensar la distancia por medio de la diferencia en tiempo entre la transmisión y recepción de una serie de pulsos que el modulo envía y captura. Este sensor es capaz de medir distancias de entre 2 cm y 400 cm de manera no

invasiva o sin contacto. El funcionamiento básico del sensor es el siguiente:

1. El módulo recibe un pulso de entrada de por lo menos 10 us de duración.
2. Una vez recibido el pulso de entrada en el *trigger*, el módulo envía ocho pulsos a una frecuencia de 40 KHz y espera a que regresen esos pulsos para sensarlos.
3. Cuando el módulo lee todos los pulsos este convierte la diferencia de tiempo en un pulso de salida que tiene un ancho de pulso determinado, en función del tiempo y la distancia en que los pulsos fueron transmitidos y recibidos.
4. Para calcular la distancia es utilizada la siguiente expresión: $\text{Distancia} = (\text{Ancho del pulso} * 170)$.

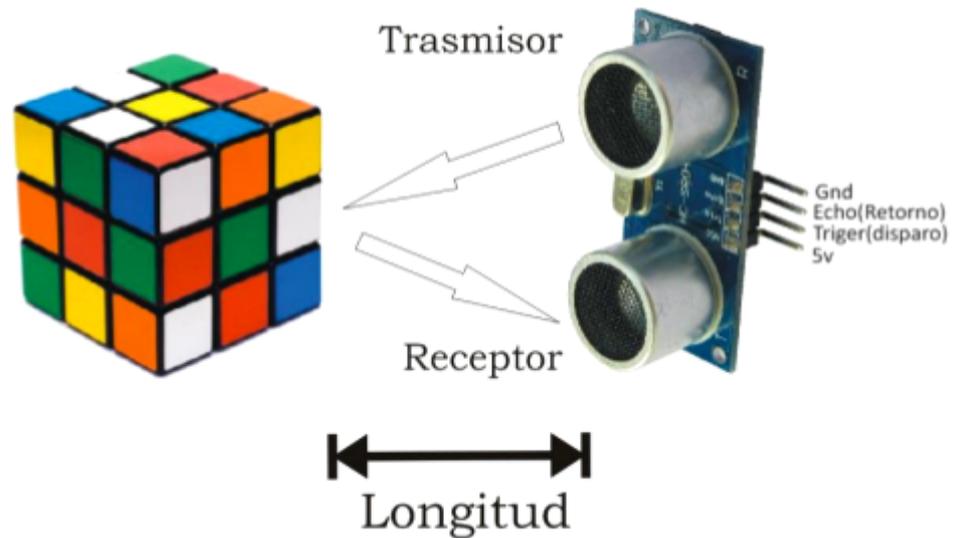


Figura 2.3: Funcionamiento sensor HC-SR04

2.2.2. Características

Las principales características del sensor ultrasónico son las siguientes:

1. Fuente de alimentación: 5V DC.
2. Angulo efectivo: menor 15 grados
3. Distancia efectiva de medición: 2 cm – 500 cm
4. Resolución: 0.3 cm

2.3. Reconocimiento Óptico de Caracteres (OCR)

El reconocimiento óptico de caracteres (*Optic Character Recognition - OCR*) realiza una simulación de la habilidad humana para el reconocimiento automático de patrones entre los diferentes caracteres alfanuméricos existentes mediante la creación y uso de modelos físicos y matemáticos.

	Ocrad	GOCR	Tesseract
Precisión en el reconocimiento	Buena	Buena	Excelente
Tiempo de Respuesta	Excelente	Mala	Buena
Sistemas operativos de desarrollo	Unix	Windows Linux	Windows Linux Mac OS
Lenguaje de desarrollo	C++	C	C y C++
Licencia	GNU Gneral Public License	GNU Gneral Public License	Licencia Apache
Formato de imagen de entrada	bmp, pgm, ppm	pnm, pbm, pgm y otros	TIFF
Calidad de imagen para análisis	Buena	Excelente	Buena

Tabla 2.1: Comparación de motores OCR

2.3.1. Reconocimiento de Patrones OCR

Existen diferentes enfoques a la hora de generar modelos matemáticos para el reconocimiento de patrones. Los más populares son:

- *Redes Neuronales*: Es un sistema inspirado en el funcionamiento del cerebro humano, que expresa la solución de un problema, no como una secuencia de pasos, sino como la evolución de un sistema dotado de cierta “inteligencia”.
- *Método Evolutivo*: En este método se define una estructura de datos que admite todas las posibles soluciones a un problema. Cada uno de los conjuntos de datos admitidos por esa estructura será un resultado. Las soluciones al problema son capaces de reproducirse entre sí, combinando sus características y generando nuevos resultados.
- *Método Probabilístico*: Se basan en la teoría de la probabilidad, utiliza análisis de varianzas, covarianzas, dispersión, distribución, etc; para encontrar la solución que mejor resuelve un problema específico.
- *Método Geométrico (Clustering)*: Es un método de agrupación de una serie de vectores de acuerdo con un criterio de cercanía. Esta cercanía se define en términos de una determinada función de distancia, geometría de formas, vectores numéricos y puntos de atracción.

2.4. Procesamiento de la Imagen de la Placa



Figura 2.4: Matrícula de vehículos en Ecuador

Las placas vehiculares son el registro que usan los vehículos automotores para su identificación y circulación legal en todo el territorio nacional. Todas las placas de identificación vehicular deben ser de una lámina metálica rectangular de 30 cm x 15 cm y deben cumplir con las normas de seguridad, recubrimiento y reflectancia determinada por el Consejo Nacional de Tránsito.

Su diseño es único para todo el país, las letras y números están en relieve de 2 mm., en color negro mate, sobre el fondo reflectivo que indica el tipo de servicio. En la parte superior central lleva la palabra ECUADOR, adicionalmente contiene tres letras y 3 o 4 dígitos, que son la clave necesaria para la identificación de cada vehículo. También existen otras matrículas especiales para vehículos de organismos

internacionales o que estén temporalmente en el país, y se distinguen por usar solo dos letras *Tabla 2.2*.

Para el presente proyecto se ha tomado en consideración solo las placas vehiculares, que son con fondo blanco; sin embargo no habría ningún tipo de impedimento para utilizar placas con otro tipo de fondo, ya que el motor de OCR trabaja con imágenes previamente binarizadas, es decir en blanco y negro, por lo cual no afectaría en el análisis final.

1. Estado	E
2. Oficial	X
3. Consejo Provincial	S
4. Municipio	M
5. Cuerpo Consular	CC
6. Cuerpo Diplomático	CD
7. Organismos Internacionales	OI
8. Asistencia Técnica	AT
9. Internación Temporal	IT

Tabla 2.2: Letras de las placas según el servicio

Otra de las características que poseen las placas vehiculares en el Ecuador es que presentan un color de fondo o una franja de color en la parte superior que corresponde al tipo de servicio que presta el vehículo, la información al respecto se la presenta en la *Tabla 2.3*.

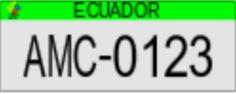
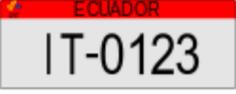
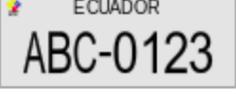
SERVICIO	COLOR	PLACA
Público o Comercial	Franja naranja	 AAC-0123
Organismos del Estado	Franja oro	 AEC-0123
Regionales, Provinciales, Municipales y Parroquiales	Franja verde limón	 AMC-0123
Diplomáticos y Organismos Internacionales	Franja azul	 CD-0123
Internación Temporal	Franja roja	 IT-0123
Particular	Color Blanco	 ABC-0123

Tabla 2.3: Tipos de placas por servicios

A continuación explicaremos cada una de las etapas que corresponden al procesamiento de la imagen de una placa vehicular como se muestra en la *Figura 2.5*.

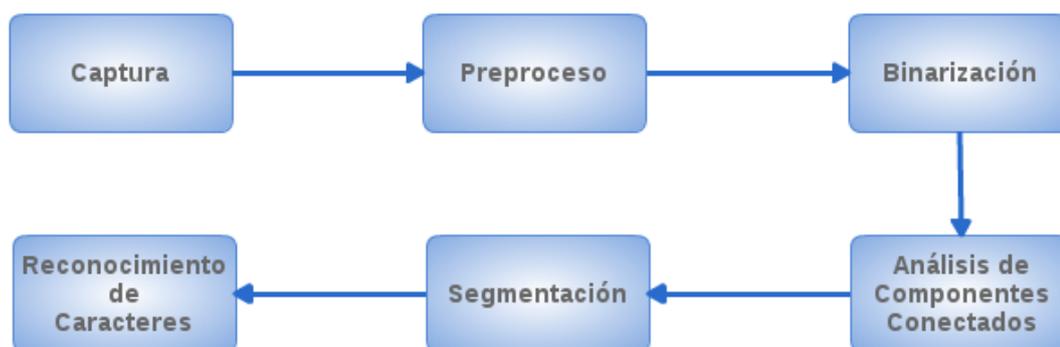


Figura 2.5: Procesamiento de imagen para reconocimiento de caracteres

2.4.1. Captura

Utilizando una cámara web o digital de alta definición se capturan imágenes de la parte frontal del vehículo.

El reconocimiento de caracteres es generalmente muy sensible a los cambios de perspectiva. La legibilidad de los caracteres puede verse afectada por la inclinación de la cámara, intensidad de la luz, entre otros factores; sin embargo, utilizar una cámara con alta resolución y definición reducirá el porcentaje de error de detección [1].



Figura 2.6: Captura de la imagen de la placa [1]

2.4.2. Preproceso

El preproceso es el conjunto de algoritmos aplicados en la imagen para mejorar su calidad. Esta es una etapa

muy importante y común en los sistemas de visión por computadora. Para el presente trabajo, el preprocesamiento involucra una etapa:

- *Redimensionar*.- El tamaño de la imagen capturada por la cámara puede ser grande y esto puede llevar a que el sistema se torne lento. Por lo que se procede a redimensionar la imagen para un desempeño más factible del sistema.

El número de canales define la información en cuanto a la cantidad de colores que se encuentran disponibles en la imagen, la misma que debe ser convertida a escala de grises.

2.4.3. Binarización

La imagen a obtener debe corresponder a la parte frontal del vehículo, ciertamente la imagen contendrá otras partes del vehículo las cuales no son necesarias para el sistema.

El área de la imagen que es de interés para el sistema, corresponde exactamente a la placa del vehículo como se muestra en la *Figura 2.7* [2].



Figura 2.7: Localización de la placa [1]

Existen dos motivos para dicha operación:

1. El resaltado de los caracteres.
2. La eliminación del color de fondo.

La binarización es realizada utilizando una técnica de procesamiento llamada *Thresholding* [3]. Los píxeles de la imagen son truncados a dos valores dependiendo del valor del umbral o *threshold*. El *threshold* o umbral requiere de un análisis previo para identificar el umbral adecuado. La técnica del umbral adaptable determina un valor óptimo de umbral para cada píxel de la imagen para evitar el inconveniente de la iluminación no uniforme.

2.4.4. Análisis de Componentes Conectados

A fin de eliminar las áreas no deseadas de la imagen, se aplica primero un algoritmo de componentes conectados a la imagen capturada de la placa. El análisis de componentes conectados se realiza para identificar los caracteres de la imagen.

La idea básica consiste en recorrer la imagen y encontrar los pixeles conectados. Cada uno de los componentes es etiquetado y extraído. La *Figura 2.8* muestra las manchas filtradas en la imagen.



Figura 2.8: Componentes conectados [1]

2.4.5. Segmentación

La segmentación es el proceso de recorte de las regiones resaltadas. Se espera que aquellas regiones sean la parte requerida de la placa. Para esto se utiliza un algoritmo especial llamado *Recorte de Imagen* [4]. En este algoritmo la imagen de la placa es escaneada verticalmente y recortada en la fila donde no hay píxeles en blanco y el área recortada es copiada en una nueva matriz *Figura 2.9*.



Figura 2.9: Segmentación [1]

Hay regiones no deseadas, incluso luego de la segmentación. Estas regiones son clasificadas mediante algoritmos con enfoque estadístico o inteligencia artificial.



Figura 2.10: Regiones segmentadas [1]

2.4.6. Reconocimiento de Caracteres

Para finalizar, las regiones seleccionadas se envían a un *Reconocedor óptico de caracteres (OCR)*, que retorna el código ASCII del número de la placa.

2.5. Herramientas para el desarrollo del sistema

2.5.1. Python

Python es un lenguaje potencialmente dinámico orientado a objetos ha sido utilizado para una amplia variedad de aplicaciones. Posee un fuerte apoyo para la integración con otros lenguajes y herramientas, y viene con bibliotecas estándar. Las siguientes son algunas de las características de Python:

- Muy claro y legible en la sintaxis.
- Las capacidades de introspección fuertes.
- Modularidad total.
- Control de errores basado en excepciones.
- Tipos de datos dinámicos de alto nivel.
- Soporta un estilo de programación orientada a objetos.

- Exportable.
- Escalable.

2.5.2. OpenCV

OpenCV es una librería de programación para visión por computadora, fue desarrollado originalmente por Intel y ahora apoyado por Willogarage. Es gratuito para uso bajo la licencia de código abierto BSD. Cuenta con más de medio millar de algoritmos optimizados.

OpenCV se utiliza en todo el mundo, su uso va desde el arte interactivo, hasta robótica avanzada. Esta librería fue escrita en C, lo que hace que sea portable para diferentes plataformas . Contenedores para lenguajes como C, Python, Ruby y Java (usando JavaCV) han sido desarrollado para promover la adopción de un público más amplio. Las últimas versiones tienen interfaces para C++. OpenCV es una librería multiplataforma, que puede ejecutarse en Linux, Mac OS y Windows. Hasta la actualidad, OpenCV es la mejor librería de visión por computadora de código abierto que los desarrolladores y los investigadores pueden utilizar.

2.5.3. Tesseract

Tesseract es un motor OCR de software libre que se desarrolló en HP entre 1984 y 1994. HP lanzó a la comunidad en 2005. Tesseract se introdujo en el 1995 [5] y se desarrolla actualmente por Google publicado bajo la Licencia de Apache. En la actualidad, puede reconocer 6 idiomas y es totalmente compatible con UTF8. Los desarrolladores pueden entrenar Tesseract con sus propias fuentes y asignación de caracteres para obtener eficientes resultados.

CAPÍTULO 3

3. Instalación y Configuración de Hardware y Software

3.1. Introducción

En el presente capítulo explicamos cómo debemos proceder al momento de configurar los componentes que se integran con el dispositivo Beaglebone Black y que son considerados para el desarrollo e implementación de nuestro sistema. Para ello debemos conocer la manera en que se encuentran distribuidos los diferentes pines o señales de la tarjeta. Para propósitos del desarrollo de nuestro proyecto utilizamos el esquema de configuración mostrado en la *Figura 3.1*.

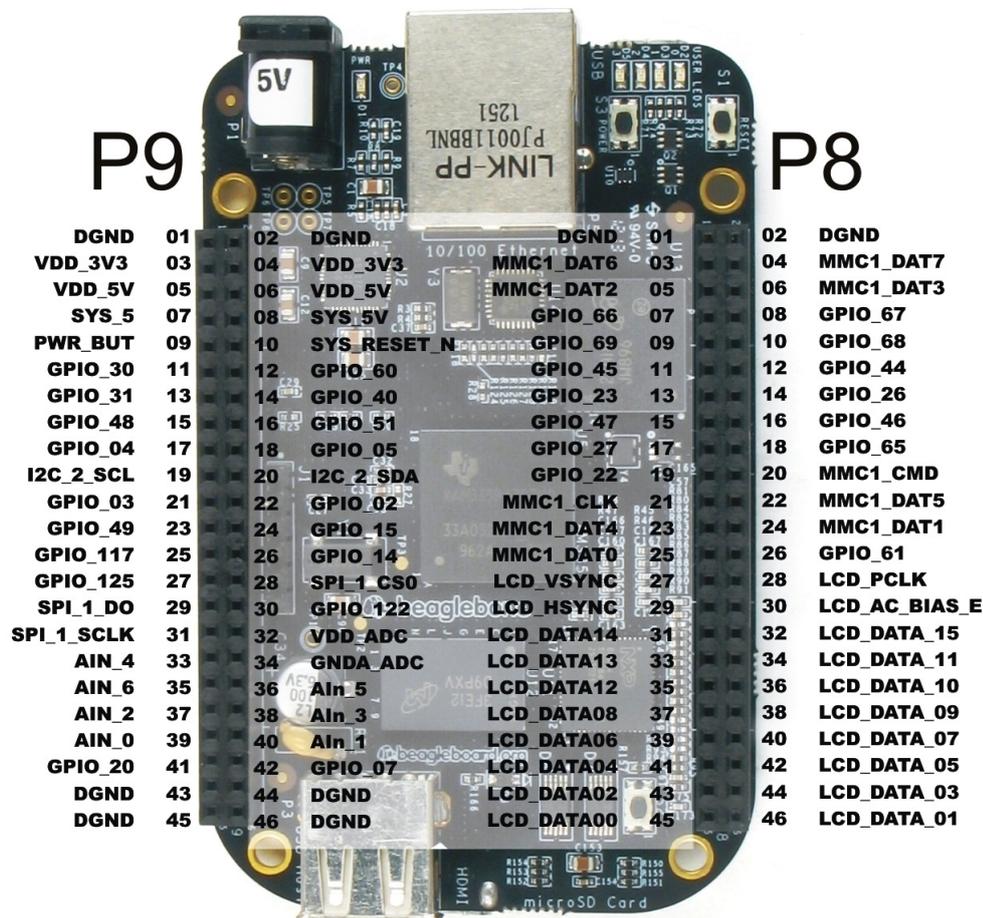


Figura 3.1: Señales de expansión de la tarjeta Beaglebone

3.2. Configuración del Motor 28BYJ-48

Para integrar un motor de paso a un dispositivo Beaglebone, necesitamos configurar los pines que enviarán las señales desde la tarjeta Beaglebone al motor. Para efectos de desarrollo de nuestro proyecto implementamos la configuración que se muestra en la *Tabla 3.1*.

MOTOR COLOR CABLE	BEAGLEBONE BLACK PIN
NARANJA	P8_13
AMARILLO	P8_14
ROSADO	P8_15
AZUL	P8_16
ROJO	P9_7

Tabla 3.1: Configuración del motor 28BYJ-48.

3.3. Configuración del Sensor de Proximidad HC-SR04

Para el correcto funcionamiento del sensor de proximidad realizamos la configuración que se muestra en la *Tabla 3.2*.

HC-SR04 PIN	BEAGLEBONE BLACK PIN
TRIGGER	P8_12
ECHO	P8_11
GND	P9_1 o P9_2
VCC	P9_5 o P9_6

Tabla 3.2: Configuración del sensor HC-SR04.

Todas las señales de expansión del dispositivo Beaglebone son configuradas por defecto como señales de salida. Una vez identificados los pines que envían o reciben las señales entre el sensor y el dispositivo, procedemos a configurar en la tarjeta Beaglebone los pines de entrada *IN* o salida *OUT* mediante el siguiente script:

```
echo 44 >/sys/class/gpio/export
echo 45 >/sys/class/gpio/export
echo out >/sys/class/gpio/gpio44/direction
echo in >/sys/class/gpio/gpio45/direction
```

También se debe conectar una resistencia de 1Kohm entre el pin ECHO del sensor y el pin de entrada del dispositivo Beaglebone ya que éste trabajan con un voltaje de 3.3V y el sensor emite 5V como se muestra en el esquema de la *Figura 3.2*

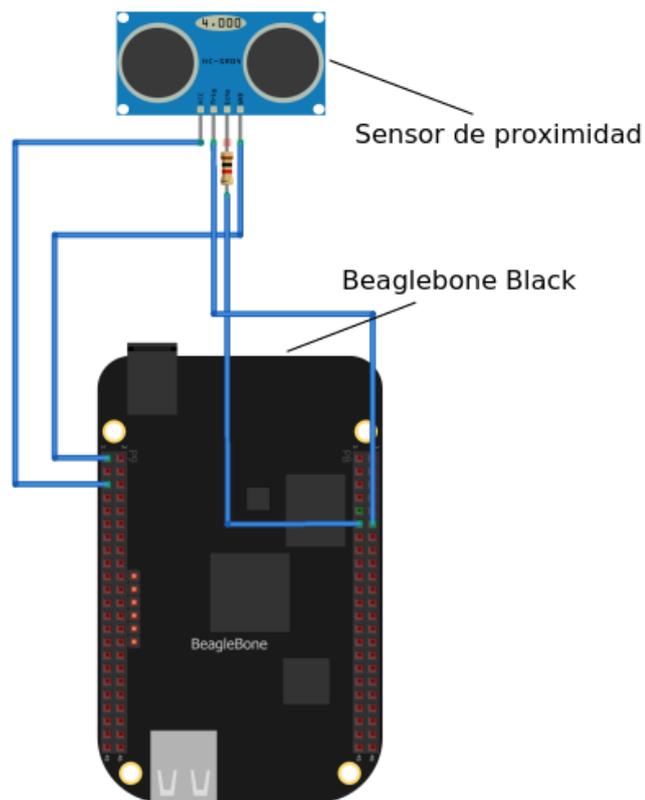


Figura 3.2: Conexión entre el HC-SR04 y el Beaglebone

3.4. Configuración de la Cámara



Figura 3.3: Conexión de una cámara USB y el Beaglebone

Se deben instalar las librerías necesarias para el reconocimiento y funcionamiento de la cámara con un dispositivo Beaglebone. *Open Source Computer Vision (OpenCV)*, es una de las librerías de software libre que se utilizan en el presente proyecto teniendo en cuenta la compatibilidad entre las librerías utilizadas para los otros módulos.

3.5. Instalación de OpenCV y Tesseract

Instrucciones para la instalación de OpenCV en el Beaglebone:

1. Descargar la última versión de OpenCV desde la página oficial o desde el repositorio *GIT*.

2. Para este caso descargamos OpenCV desde el repositorio: *git clone https://github.com/Itseez/opencv.git*, utilizando la terminal de Linux.
3. Luego se debe crear una carpeta dentro del directorio opencv:
 - `mkdir build`
 - `cd build`
 - `cmake -D CMAKE_BUILD_TYPE=RELEASE -D CMAKE_INSTALL_PREFIX=/usr/local -D WITH_CUDA=OFF -D WITH_CUFFT=OFF -D WITH_CUBLAS=OFF -D WITH_NVCUVID=OFF -D WITH_OPENCL=OFF -D WITH_OPENCLAMDFFT=OFF -D WITH_OPENCLAMDBLAS=OFF -D BUILD_opencv_apps=OFF -D BUILD_DOCS=OFF -D BUILD_PERF_TESTS=OFF -D BUILD_TESTS=OFF -D ENABLE_NEON=on ..`
 - `make`
 - `sudo make install`
4. Ahora, configurar el OpenCV con el siguiente comando: `sudo ldconfig`

Instrucciones de instalación de Tesseract en el Beaglebone:

1. Descargamos el Tesseract.
2. Descomprimos la carpeta, luego nos aseguramos de ingresar al directorio para ejecutar los siguientes comandos:
 - ./configure
 - make
 - sudo make install

CAPÍTULO 4

4. Sistema de Control de Acceso Vehicular Utilizando Beaglebone Black

4.1. Sistema ANPR

Los sistemas de Reconocimiento Automático de Placas Vehiculares ANPR, son aplicaciones de visión por computador que están compuestos de hardware y software que permiten la lectura de la placa de un vehículo *Figura 4.1*, su proceso básico radica en que una cámara captura la imagen de la placa y luego con la ayuda de algunas técnicas especializadas se procede a la binarización, segmentación y adelgazamiento de la misma, para finalmente ser reconocida por

medio de un algoritmo de Reconocimiento Óptico de Caracteres OCR [6].



Figura 4.1: Esquema ANPR

A diferencia de otras aplicaciones de visión por computador en el entorno industrial, este posee la dificultad de trabajar en un entorno *no controlado*, como por ejemplo la iluminación que varía a cada hora del día, la inclinación y estado de conservación de la placa, por lo que a menudo se tiende a utilizar iluminación infrarroja para hacer posible que la cámara capture imágenes en cualquier momento del día.

Los sistemas ANPR's pueden ser utilizados para:

- *La gestión de aparcamiento de abonados*: usando la matrícula a modo de “clave” o “comando” para acceder al estacionamiento.
- *Control de fraude en autopistas*: para poder determinar si un vehículo fue plagiado o no se encuentra autorizado para circular.
- *Control de velocidad media en autopistas*: situando lectores en varios accesos y salidas de la red vial.
- *Control de camiones*: situando un lector de matrículas junto a la báscula que mide la carga del camión.
- *Inventario de vehículos*: además de capturar la imagen de la matrícula se podrían adquirir imágenes adicionales del vehículo, para poder determinar el estado del mismo en el instante de ingreso al aparcamiento o en caso de sufrir algún siniestro.

4.2. Módulos del Sistema

4.2.1. Módulo de Captura de la Placa del Vehículo

El Módulo de Captura de la Placa del Vehículo comprende en la proyección de una cámara hacia la parte frontal del vehículo, obteniendo una imagen en la que se resalta el área de la placa como se ilustra en la *Figura 4.2*.

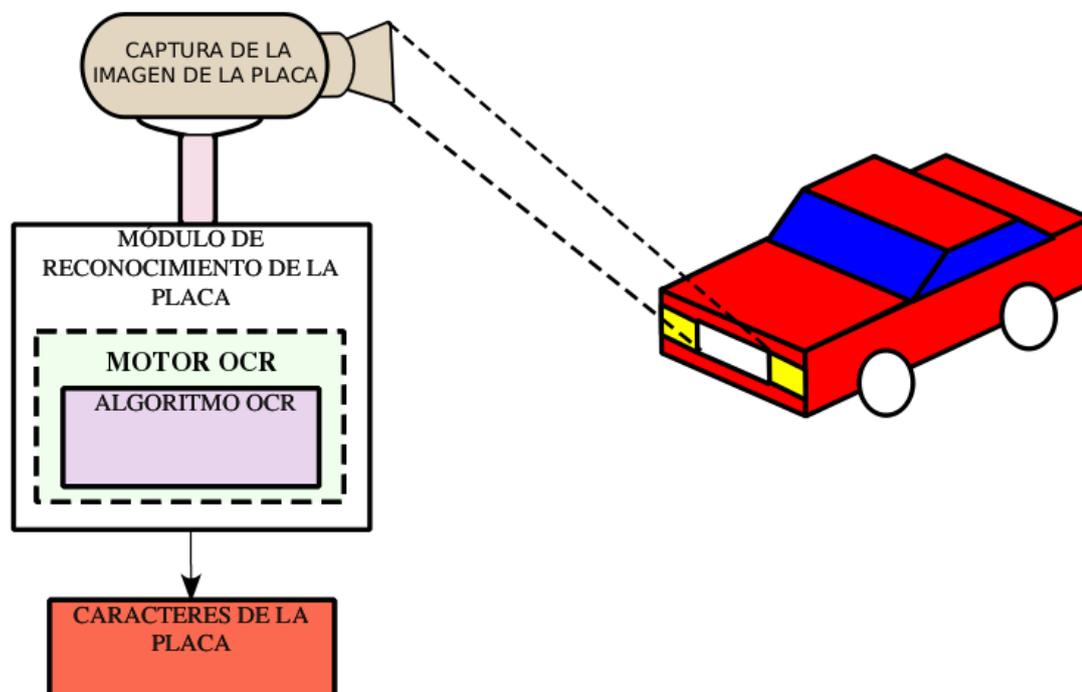


Figura 4.2: Captura de la imagen de la placa del vehículo

4.2.2. Módulo de Lectura del Sensor de Proximidad HC-SR04

El módulo de lectura de proximidad utilizando un sensor *HC-SR04*, fué implementado con el objetivo de controlar la distancia existente entre el vehículo y el sensor de proximidad. Una vez que ha ingresado al garaje, el módulo de lectura del sistema le indica el momento en que debe emitir la señal al motor para ejecutar el cierre automático de la puerta del garaje.

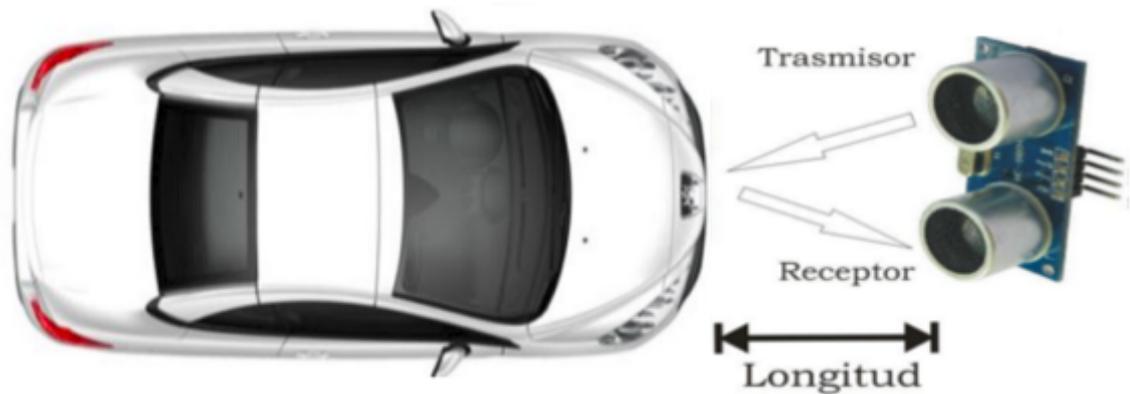


Figura 4.3: Proximidad del vehículo dentro del garaje

4.2.3. Módulo de Apertura y Cerrado de Puerta

En el presente módulo y para efectos del proyecto utilizamos un motor de paso, como el que se muestra en la *Figura 4.3*. Decidimos utilizar éste tipo de motor ya que nos permite tener un mayor control en cuanto al sentido del giro y número de vueltas que debe dar el eje del motor para abrir o cerrar la puerta de un garaje.

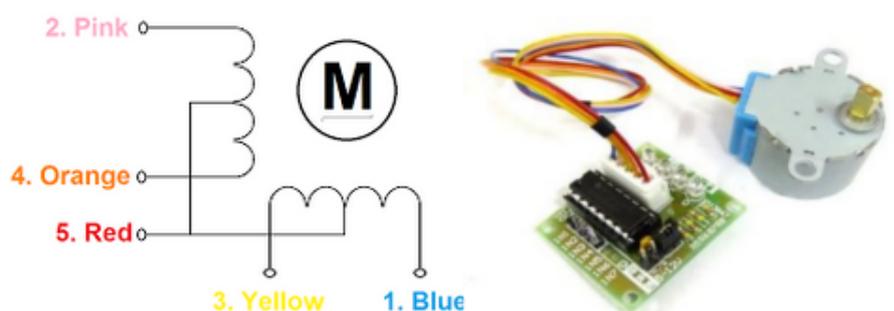


Figura 4.4: Motor 28BYJ-48

El módulo fue implementado con el objetivo de abrir o cerrar la puerta de un garaje, una vez que:

- La imagen de la placa del vehículo haya sido capturada, procesada y debidamente validada (abrir).
- El sensor de proximidad haya detectado que el vehículo está totalmente dentro del garaje (cerrar).

En cada uno de los escenarios anteriormente expuestos; se emite una señal desde el dispositivo Beaglebone hacia el motor de paso.

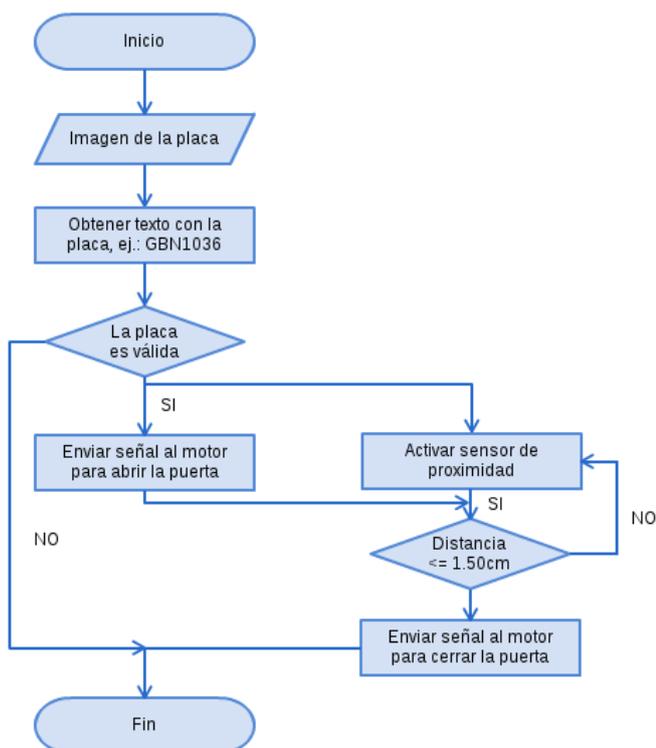


Figura 4.5: Organigrama del sistema

CAPÍTULO 5

5. Análisis de Resultados

5.1. Resultados

En este trabajo se implementó un sistema de control de acceso vehicular utilizando un dispositivo Beaglebone. Al analizar las diferentes herramientas de software libre disponibles para la implementación del proyecto, fueron escogidos OpenCV, Python y Tesseract, debido a que son compatibles y flexibles al momento de trabajar sobre la tarjeta Beaglebone Black. Adicionalmente, este dispositivo funciona bajo cualquier distribución del sistema operativo linux, resultando en un dispositivo de bajo costo computacional con una alta gama de opciones para integrar diferentes componentes como los que se

utilizaron para el desarrollo de este proyecto: un sensor de proximidad, un motor de paso y una cámara web.

También pudimos observar durante el desarrollo de nuestro proyecto que existen recursos implementados en trabajos similares, cómo es el caso del reconocimiento de placas vehiculares en tiempo real, lo que contribuyó con el desarrollo de algoritmos que representen trabajo innecesario y que de alguna forma hubiera comprometido la capacidad de cómputo de nuestro sistema.

Otro punto a considerar, es que para la mejora del sistema, se lograron implementar técnicas de corrección de imágenes capturadas en tiempo real, además de la integración de otras funcionalidades que hacen del sistema una herramienta más práctica.

A continuación se presenta una imagen de la detección en tiempo real de la placa de un vehículo, utilizando un ANPR básico.

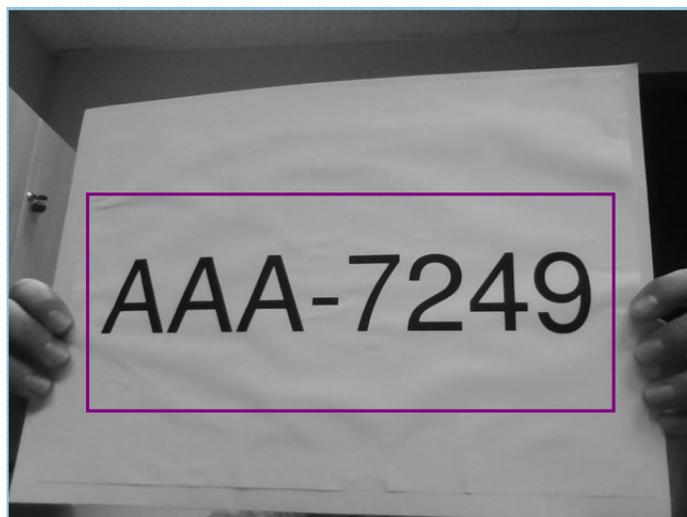


Figura 5.1: Placa detectada en tiempo real.

Para el módulo del sensor de proximidad y el módulo de la cámara se realizaron pruebas para verificar el correcto funcionamiento de cada uno de ellos, a continuación se muestran las imágenes donde se puede observar las mediciones realizadas por el sensor de proximidad y el análisis de la imagen de la placa del vehículo para obtener el texto de la misma.

```
pi@pi2cidls ~/Downloads/hcsr04 $ sudo python ultrasonic_2.py
Prueba Sensor de Proximidad HCSR04
Distancia : 120.3 cm
Distancia : 166.9 cm
Distancia : 166.2 cm
Distancia : 119.9 cm
Distancia : 166.2 cm
Distancia : 165.5 cm
Distancia : 166.5 cm
Distancia : 166.4 cm
Distancia : 121.1 cm
Distancia : 166.1 cm
Distancia : 167.2 cm
Distancia : 121.3 cm
Distancia : 120.4 cm
Distancia : 212.7 cm
```

Figura 5.2: Resultados de las lecturas de distancia que se obtuvieron con el sensor de proximidad.

Para realizar la prueba del módulo de la cámara se necesita compilar el código respectivo y luego ejecutar el comando `./proyecto`, obteniendo los resultados que se muestran en la *Figura 5.3*.

```

Archivo  Editar  Ver  Buscar  Terminal  Ayuda
Placa:
{AAA-7249
Placa:
[AAA-7249
Placa:
[AAA-7249
Placa:
'AAA-7249
Placa:
IAAA-7249'
Placa:
IAAA-7249!
Placa:
IAAA-7249
Placa:
[AAA-7249
Placa:
[AAA-7249
Placa:
VAAA-7249
Placa:
[AAA-7249
Placa:
"AAA-7249
Placa:
"AAA-7249
Placa:
AAA-7249
Placa autorizada!

```

Figura 5.3: Resultados de la captura de la placa del vehículo.

A continuación se presenta una imagen del sistema integrado (Beagle-Bone Black, cámara, motor y sensor de proximidad). Donde se muestra la conexión de cada componente con la tarjeta Beaglebone Black, dicha conexión puede variar de acuerdo a la configuración que se haya

hecho en los pines GPIO.

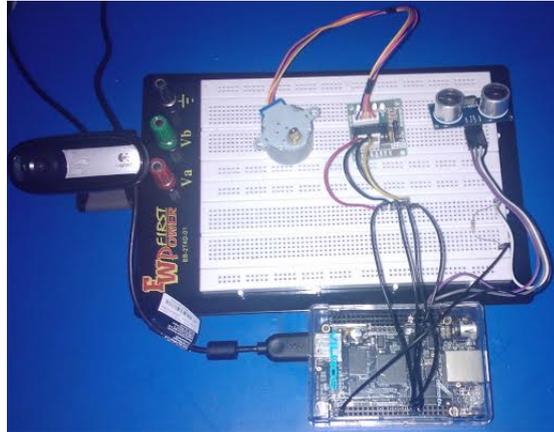


Figura 5.4: Sistema integrado del proyecto desarrollado.

En la *Figura 5.5* se muestra la maqueta donde se colocarán el motor, la cámara y los sensores de proximidad, que corresponden a cada uno de los componentes del proyecto.

La presente estructura fue diseñada con el objetivo de mostrar el funcionamiento del proyecto, lo que ayuda a tener una mejor perspectiva al momento de implementar el presente proyecto en un nivel macro.



Figura 5.5: Estructura donde estarán colocados cada uno de los componentes del proyecto.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

1. Un sistema de reconocimiento de placas puede actuar junto con otros dispositivos como los utilizados para el desarrollo del sistema y a su vez pueden ser integrados a un dispositivo de bajo costo como lo es Beaglebone.
2. La solución propuesta permite ampliar el funcionamiento del reconocimiento de placas vehiculares (ANPR) y aunque para el presente trabajo se ha optado por una técnica específica, siempre podrán haber mejores recursos o procedimientos tecnológicos que mejoren sus resultados.
3. El presente proyecto puede ser utilizado como un respaldo para futuros trabajos, donde la tarjeta *Beaglebone Black* actúa como el controlador del sistema e interactúa con diferentes dispositivos electrónicos, como: sensores, motores, cámaras, leds, entre otros.

4. Las herramientas de código abierto pudieron ser satisfactoriamente adaptadas en dominios de cálculos numéricos. Python y OpenCV son buenos puntos de partida para los investigadores y estudiantes de visión por computador.
5. En cuanto a los tiempos de respuesta de cada uno de los módulos que forman parte del presente trabajo oscilan en valores menores a 1 seg.
6. Durante las pruebas del módulo del motor se observó que al incrementar la velocidad de giro, el motor no realizaba el giro de manera correcta, debido a que se encontraba conectado de manera directa a los pines GPIO del Beaglebone.

Recomendaciones

1. Para un óptimo funcionamiento del sistema es necesario contar con una cámara que posea una buena resolución en exteriores, como una cámara IR para capturar la mayor cantidad de detalles de la placa.
2. La estandarización de las placas vehiculares y una revisión periódica del estado de las mismas, permitirá la adquisición de una imagen más nítida, lo que hará también más viable la implementación de proyectos similares al que se ha desarrollado.
3. Para evitar inconvenientes de desconfiguración de drivers que se pueden presentar al momento de integrar los módulos en el dispositivo Beaglebone,

se recomienda primeramente probar de manera individual cada uno de los módulos con todas sus librerías de dependencia.

4. Para la integración de los módulos necesarios, se debe tener presente la disponibilidad de las librerías para el correcto funcionamiento del sistema.

Futuros trabajos

1. Como futuro trabajo se mejorará el módulo de procesamiento de imágenes para reducir el tiempo de extracción de la placa. Actualmente el módulo de procesamiento tiene limitaciones en cuanto a la proximidad de la placa a la cámara. Mediante la mejora de este módulo el tamaño de la región de interés (ROI) será variable y se ajustará a la proximidad de la placa del carro a la cámara, permitiendo realizar la detección mucho antes de que el vehículo se encuentre dentro de un área de proximidad actualmente definido de forma manual.
2. Para tener una mayor precisión en cuanto a la proximidad del vehículo se implementará una solución con sensores más sofisticados, como por ejemplo dispositivos láser.
3. En el futuro se implementará nuevos módulos, uno de ellos será el que controle el encendido y apagado de la luz del interior del garaje y el otro módulo tendrá la función de indicar utilizando señales de sonido (*timbre*) la distancia que existe entre el vehículo y el sensor de proximidad.

BIBLIOGRAFÍA

- [1] KM Sajjad. Automatic license plate recognition using python and opencv. *Department of Computer Science and Engineering MES College of Engineering, Kuttippuram, Kerala, me@ sajjad. in.*
- [2] Aura Conci, JER De Carvalho, and TW Rauber. A complete system for vehicle plate localization, segmentation and recognition in real life scene. *Latin America Transactions, IEEE (Revista IEEE America Latina)*, 7(5):497–506, 2009.
- [3] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *Automatica*, 11(285-296):23–27, 1975.
- [4] Prathamesh Kulkarni, Ashish Khatri, Prateek Banga, and Kushal Shah. Automatic number plate recognition (anpr) system for indian conditions. In *Radioelektronika, 2009. RADIOELEKTRONIKA'09. 19th International Conference*, pages 111–114. IEEE, 2009.
- [5] Stephen V Rice, Frank R Jenkins, and Thomas A Nartker. The fourth annual test of ocr accuracy. Technical report, Technical Report 95, 1995.

- [6] Richard Gutierrez, Ma Fernanda Frydson, and Boris Vintimilla. Aplicación de visión por computador para el reconocimiento automático de placas vehiculares utilizando ocr's convencionales. *Tesis de pregrado FIEC-ESPOL*, 2011.