



**“Desarrollo de un prototipo para lectura y registro automático de información de visitas
en puntos de control de acceso a un establecimiento”**

Loja Paucar, Mauricio Manuel

Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Ingeniería en Electrónica, Automatización y Control

Trabajo de titulación, previo a la obtención del título de Ingeniero en Electrónica,
Automatización y Control

Ing. Silva Tapia, Rodrigo

09 de agosto del 2023

Trabajo_Titulacion_Mauricio_Loja.pdf

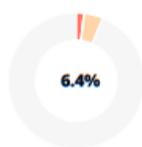
Scan details

Scan time:
August 7th, 2023 at 15:7 UTC

Total Pages:
62

Total Words:
15453

Plagiarism Detection



Types of plagiarism		Words
Identical	1.7%	260
Minor Changes	0.2%	29
Paraphrased	4.2%	646
Omitted Words	5%	775

AI Content Detection



Text coverage
 ● AI text
 ● Human text

Ing. Rodrigo Silva T.
C.I 0602199523



RODRIGO SILVA TAPIA

i Alerts: (1)

Cross Language: Same Document Language

Submitted language and cross-language text are the same language. No credits were used.

2/5 Severity



🔍 Plagiarism Results: (48)

🌐 ¿Qué es el OCR (Reconocimiento Óptico de Caracter... 0.8%

<https://blog.truora.com/es/ocr>

Ivan de Souza

...

🌐 Aprendizaje automático para la detección y el reco... 0.6%

<https://ts2.space/es/aprendizaje-automatico-para-la-deteccio...>

Saltar al contenido LIM Center, Aleje Jerozolimskie 65/79, 00-697 Warsaw, Poland +48 (22) 364 58 00 ...



Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Ingeniería en Electrónica, Automatización y Control

Certificación

Certifico que el trabajo de titulación: **"Desarrollo de un prototipo para lectura y registro automático de información de visitas en puntos de control de acceso a un establecimiento"** fue realizado por el señor **Loja Paucar, Mauricio Manuel**; el mismo que cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, además fue revisado y analizado en su totalidad por la herramienta de prevención y/o verificación de similitud de contenidos; razón por la cual me permito acreditar y autorizar para que se lo sustente públicamente.

Sangolquí, 09 de agosto de 2023

Firma:



El medio electrónico para
RODRIGO SILVA TAPIA

Ing. Silva Tapia, Rodrigo

C. C. 0602199523



Departamento de Eléctrica, Electrónica y Telecomunicaciones
Carrera de Ingeniería en Electrónica, Automatización y Control

Responsabilidad de Autoría

Yo, **Loja Paucar, Mauricio Manuel**, con cédula de ciudadanía n°0705547768, declaro que el contenido, ideas y criterios del trabajo de titulación: **Desarrollo de un prototipo para lectura y registro automático de información de visitas en puntos de control de acceso a un establecimiento** es de mi autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Sangolquí, 07 de agosto de 2023

Firma


Loja Paucar, Mauricio Manuel
C.C.: 0705547768



Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Ingeniería en Electrónica, Automatización y Control

Autorización de Publicación

Yo **Loja Paucar, Mauricio Manuel**, con cédula de ciudadanía n°0705547768, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: **Desarrollo de un prototipo para lectura y registro automático de información de visitas en puntos de control de acceso a un establecimiento** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

Sangolquí, 07 de agosto de 2023

Firma

Loja Paucar, Mauricio Manuel

C.C.: 0705547768

Dedicatoria

*A mi familia que siempre me brindó un apoyo incondicional para lograr mis objetivos.
Gracias por estar siempre pendientes de mí, por orientarme y alentarme a seguir siempre
adelante.*

Mauricio Manuel Loja Paucar

Agradecimiento

A Dios por darme fortaleza en los momentos difíciles y guiarme a tomar las decisiones correctas.

A mis padres Luz y Manuel, gracias por enseñarme el valor del esfuerzo y la perseverancia. A mis hermanos, en especial a Germania gracias por creer en mí, apoyarme y motivarme a cumplir mis sueños. A Mercedes por darme un segundo hogar y apoyarme incondicionalmente en todo momento.

A mis amigos, por estar a mi lado en los momentos buenos y difíciles. Su amistad ha sido un pilar fundamental para superar los retos que se presentaron en este proceso.

A mi tutor Ing. Rodrigo Silva, al departamento de Eléctrica, Electrónica y Telecomunicaciones de la Universidad de las Fuerzas Armadas “ESPE” y los docentes que lo integran, gracias por la guía, el conocimiento y formación que me otorgaron en la formación de esta profesión.

Mauricio Manuel Loja Paucar

Índice de Contenido

Capítulo I: Introducción.....	16
Antecedentes.....	16
Justificación e Importancia.....	18
Alcance del Proyecto	20
Objetivos.....	21
Objetivo General	21
Objetivos Específicos	21
Capítulo II: Marco conceptual	22
Internet de las cosas.....	22
Visión Artificial	25
Características	25
Visión artificial inteligente y los sistemas autónomos.....	27
Reconocimiento Óptico de Caracteres (OCR).....	29
Estructura básica de un algoritmo OCR.....	29
Métodos y Técnicas para OCR.....	30
Servicios de seguridad AAA.....	30
Autenticación.....	31
Autorización	31
Contabilización	31
Arquitectura Web	31

Arquitectura Back end	31
Arquitectura Front End	31
Edge Computing	32
Arquitectura	32
Capítulo III: Diseño e Implementación	33
Descripción general de la solución.....	33
Procesos en el prototipo del dispositivo.....	35
Procesos en el terminal del administrador	36
Desarrollo Hardware	36
Prototipo IOT	36
Carcasa del prototipo	39
Desarrollo Software	40
Captura de imagen.....	41
Preprocesamiento	42
Validar documento	53
Lectura de información	56
Diseño de la aplicación Web	60
Capítulo IV: Montaje del Prototipo	63
Dispositivo Edge Computing	63
Instalación del sistema operativo.....	64
Configurar software y librerías necesarias.....	65

	10
Acople del sistema y funcionamiento general	65
Capítulo V: Evaluación del prototipo	69
Desempeño general.....	69
Pruebas y resultados	70
Evaluación de rendimiento del prototipo	72
Tiempo de respuesta del prototipo	73
Análisis de resultados	75
Errores de lectura debido a factores externos.	76
Capítulo VI: Conclusiones y Recomendaciones	77
Trabajo Futuro	79
Bibliografía	80
Apéndices	83

Índice de Tablas

Tabla 1 Caracteres eliminados o reemplazados en el texto final de nombres	58
Tabla 2 Caracteres eliminados o reemplazados en el texto final de numero	58
Tabla 3 Matriz de confusión	72
Tabla 4 Resultados obtenidos de la lectura de información.....	73

Índice de Figuras

Figura 1 Arquitectura general del sistema.....	21
Figura 2 Arquitectura para sistemas IoT	24
Figura 3 Imagen a) original b) segmentada	26
Figura 4 Detección de bordes con algoritmo Canny.....	27
Figura 5 Detección de vehículos utilizando visión artificial	28
Figura 6 Estructura básica de un algoritmo OCR.....	29
Figura 7 Diagrama de bloques de procesos realizados para desarrollo del prototipo....	34
Figura 8 Diagrama de bloques del sistema implementado en el prototipo	35
Figura 9 Diagrama de conexiones del prototipo.....	38
Figura 10 Circuito de acondicionamiento para LEDs indicadores y sensor.	39
Figura 11 Carcasa diseñada en Fusion 360.....	39
Figura 12 Medidas internas de la carcasa	40
Figura 13 Diagrama de flujo para realizar la lectura y envío de información.	41
Figura 14 Imagen obtenida de cámara a) cedula nueva b) cedula antigua	42
Figura 15 Banco de imágenes para calibración de cámara.....	43
Figura 16 Resultado de esquinas identificadas del tablero de ajedrez.	45
Figura 17 Imagen sin distorsión efecto de barril a) cédula nueva b) cédula antigua.....	47
Figura 18 Imagen en escala de grises	48
Figura 19 Imagen con filtro de umbral adaptativo	49
Figura 20 Imagen con filtro dilatación	50
Figura 21 Imagen con filtro erosión.....	51
Figura 22 Imagen con filtro desenfoque mediano	51
Figura 23 Imagen con filtro de negación de bits.....	52
Figura 24 Imagen con filtro dilatación	52
Figura 25 Resultado de preprocesamiento de imagen.....	53

Figura 26	Diagrama de flujo para validación de documento.....	53
Figura 27	Identificación de documento ecuatoriano en cédula nueva	54
Figura 28	Identificación de cédula de identidad en cedula nueva	54
Figura 29	Identificación de documento ecuatoriano en cedula antigua	55
Figura 30	Identificación de documento ecuatoriano en cedula antigua	55
Figura 31	Identificación de cédula de identidad en cédula antigua	56
Figura 32	Etapas para la extracción de información en cédula antigua.....	57
Figura 33	Etapas para la extracción de información en cédula nueva.....	59
Figura 34	Extracción de foto en a) cédula nueva b) cedula antigua	59
Figura 35	Flujo peticiones/respuestas HTTP	60
Figura 36	Diagrama de flujo front-end.....	61
Figura 37	Visualización de información en página web.....	62
Figura 38	Acople de Jetson Nano e iluminación LED	66
Figura 39	Conexión de sensores e indicadores	66
Figura 40	Acople de cámara y dispositivo de sonido en pared izquierda	67
Figura 41	Adaptador de audio y Hub 4 puertos colocados en pared inferior	67
Figura 42	Integración de todos los componentes necesarios para el funcionamiento. ..	68
Figura 43	Desempeño general del prototipo	70
Figura 44	Lecturas realizadas en prototipo de lectura.....	71
Figura 45	Clasificación de errores por tipo de información leída	71
Figura 46	Gráfico de tiempos de respuesta del prototipo	74
Figura 47	Tiempo de respuesta del prototipo.....	74
Figura 48	Error de impresión presentado en cédula ecuatoriana antigua	76
Figura 49	Holograma reflejado por iluminación en cedula ecuatoriana antigua.....	76

Resumen

El desafío de desarrollar sistemas que permitan agilizar y mejorar los procesos de identificación y registro en puntos de control de acceso ha impulsado la implementación de soluciones tecnológicas avanzadas cubriendo las demandas crecientes de seguridad en diversos entornos. El presente proyecto plantea implementar un prototipo basado en conceptos de IoT y Edge Computing que sea capaz de extraer información del nombre, número de identificación y fotografía de una credencial como la cédula de identidad ecuatoriana utilizando algoritmos de detección facial y reconocimiento óptico de caracteres (OCR) tales como *Haar Cascade* y *Tesseract OCR* implementados en una placa de desarrollo edge computing Nvidia, para luego enviar automáticamente el registro a una base de datos de un servidor web y finalmente mostrar esta información en un navegador web. Se realizaron pruebas de funcionamiento para evaluar el desempeño del prototipo y garantizar su confiabilidad en un entorno real. Este proyecto busca aportar una solución innovadora para la automatización de la identificación y registro de personas en puntos de acceso a establecimientos, lo cual tiene aplicaciones en diversos sectores, como control de fronteras, seguridad física en empresas, instituciones gubernamentales y otros espacios de gran afluencia de personas. Se espera que los resultados obtenidos de este proyecto sirvan como una valiosa contribución para futuras implementaciones y desarrollos tecnológicos en el campo del reconocimiento de documentos en tiempo real.

Palabras clave: Control de acceso, algoritmos de detección, edge computing, seguridad física

Abstract

The challenge of developing systems to improve identification and registration processes at access control points has driven the implementation of advanced technological solutions to meet the growing demands of security in various environments. This project proposes to implement a prototype based on IoT and Edge Computing concepts that is capable of extracting information from the name, identification number and photograph of a credential such as the Ecuadorian identity card using face detection and optical character recognition (OCR) algorithms such as Haar Cascade and Tesseract OCR implemented on a Nvidia edge computing development board, and then automatically send the record to a web server database and finally display this information in a web browser. Functional tests were conducted to evaluate the performance of the prototype and ensure its reliability in a real environment. This project seeks to provide an innovative solution for the automation of the identification and registration of people at access points to establishments, which has applications in various sectors, such as border control, physical security in companies, government institutions and other sites with agglomeration of people. It is expected that the results obtained from this project will serve as a valuable contribution for future implementations and technological developments in the field of real-time document recognition.

Keywords: Access control, detection algorithms, edge computing, physical security.

Capítulo I: Introducción

Antecedentes

El registro de ingreso en puntos de acceso a establecimientos ya sea del sector comercial, industrial o residencial se ha convertido en un tema de interés como política de seguridad y servicio de atención a los usuarios en cualquier tipo de organizaciones. Según un reporte de la Organización del Trabajo (OIT), la inseguridad es una de las cinco principales preocupaciones en todos los sectores del Ecuador. Después de Guatemala y Nicaragua, Ecuador es el tercer país con mayor tasa de inseguridad durante los primeros cuatro meses del año, según una encuesta de la consultora CID Gallup elaborada entre el 10 y 19 de mayo de 2022 (Universo, 2022) . Ante esta situación, especialistas en temas de seguridad han propuesto, entre otras cosas, que se fortalezcan los controles y vigilancia en los establecimientos.

El Internet de las cosas (IoT) utiliza dispositivos inteligentes e Internet para brindar soluciones innovadoras a diversos desafíos y problemas relacionados con diversas industrias comerciales, gubernamentales y públicas/privadas en todo el mundo. Las principales aplicaciones que se desarrollan con IoT y Visión Artificial pueden ser implementadas en educación, medicina, edificios inteligentes, sistemas de vigilancia de personas y vehículos, entre otros.

La visión artificial es un campo de la Inteligencia Artificial que puede emular la visión humana. Se utiliza el software y hardware de un computador para analizar y procesar información y datos visuales, esto incluye el proceso de adquisición, transmisión, procesamiento, proyección, almacenamiento y comprensión de la información visual del objeto o el entorno (Xin & YiLiang, 2018). La detección de objetos es un campo de aplicación basado en la visión artificial que estudia cómo detectar la presencia de objetos en una imagen sobre la base de su apariencia visual, y se han desarrollado una variedad de algoritmos para lograr la detección de diferentes objetos.

La detección de objetos mediante clasificadores en cascada basados en algoritmos *Haar Cascade* es un método eficaz propuesto por Paul Viola y Michael Jones (Viola & Jones, 2001). *Haar Cascade* es un algoritmo que se utiliza para identificar rostros en una imagen o un video en tiempo real, el algoritmo recibe muchas imágenes positivas que consisten en rostros y muchas imágenes negativas que no constan de ningún rostro para entrenar sobre ellas, luego se realiza una extracción de características donde cada característica que es un valor único obtenido al restar la suma de píxeles debajo de un rectángulo blanco de la suma de píxeles debajo de un rectángulo negro (Purwa, Gusti, & Soeleman, 2018). También se puede combinar con otros algoritmos y métodos para mejorar el rendimiento, en (Cuimei, Zhiliang, Nan, & Jianhua, 2017) se propone un nuevo algoritmo de detección de rostros humanos mediante el algoritmo primitivo Haar Cascade combinado con tres clasificadores débiles adicionales. Los tres clasificadores débiles se basan en la coincidencia del histograma del tono de la piel, la detección de los ojos y la detección de la boca con un incremento del rendimiento de detección desde un valor de predicción de falsos positivos de 78,18 % a 98,01 %.

El reconocimiento óptico de caracteres (OCR) es una rama importante en el campo de la visión artificial. Implica reconocimiento de patrones, procesamiento de imágenes, procesamiento de señales digitales, inteligencia artificial y otras disciplinas. Tiene un valor de uso importante y un significado teórico en campos de alta tecnología como el procesamiento de información de palabras, la traducción automática y el sistema de monitoreo en tiempo real (He, 2020). En el pasado, los algoritmos de detección y reconocimiento de texto se basaban básicamente en características diseñadas artificialmente y métodos tradicionales de procesamiento de imágenes. Estas funciones y algoritmos eran difíciles de diseñar y requerían mucho conocimiento profesional y soporte de experiencia, por lo que la precisión no era alta y no se generalizaron. En los últimos años, con el rápido desarrollo de la tecnología de deep learning, se han logrado avances en los campos de la visión artificial, como la clasificación de imágenes, la detección de objetos y la segmentación semántica. El rendimiento de los modelos OCR se basa en redes

neuronales artificiales multicapa. Para la visión por computadora, los tipos más comunes son las redes neuronales recurrentes (RNN) o, más precisamente, la memoria a corto plazo (LSTM) y las redes neuronales convolucionales (CNN).

Tesseract es un motor de reconocimiento óptico de caracteres para varios sistemas operativos. En su arquitectura presenta una red de memoria a corto plazo (LSTM, Long short-term memory) la cual es una red neuronal recurrente (RNN) que integra ciclos de retroalimentación que permiten retener la información en ciertos pasos del entrenamiento a través de la conexión de la salida de cada capa. (Martinez & Araujo)

La visión artificial incorporada en los sistemas IoT pueden ser muy útiles cuando se trata de proyectos que den una solución efectiva a procesos cotidianos como el registro de datos en puntos de control de accesos a establecimientos. En (Al Ghaithi & Eaganathan, 2016) se presenta un ejemplo de sistema de registro automático de visitas, reemplazando el registro manual de la información del visitante mediante el uso de una tarjeta de identificación multipropósito que contiene un chip de identificación para el acceso a las áreas restringidas del campus de Pekan de la Universidad de Malasia Pahang. De este estudio, muestra que el porcentaje de mejora en tiempos para culminar el proceso de registro para los visitantes nuevos mediante el uso del sistema es entre un 26 % y un 54 % superior en comparación con el método de registro manual, mientras que el porcentaje de mejora para los visitantes existentes en la base de datos es entre un 86% y un 91%.

Justificación e Importancia

Mantener el control sobre el ingreso de personas visitantes a lugares concurridos, es de primordial importancia para fortalecer la seguridad de las organizaciones. En nuestro medio, la gestión de visitantes sigue siendo un problema debido a que los sistemas de control de acceso disponibles no ofrecen funcionalidades prácticas y verdaderamente seguras. Los sistemas más utilizados adolecen problemas como la dificultad para mantener un registro histórico de visitas, la falta de funciones específicas para apoyar el movimiento de visitantes y la integración de

sistemas con control de acceso, lo que efectivamente dificulta el proceso de registro e identificación.

Comúnmente, el registro de visitantes se realiza de forma manual, donde una persona solicita la cédula de identidad del visitante y transcribe manualmente los datos del documento de identidad en una hoja de registro. Este proceso puede resultar tedioso, consume tiempo e incluso pueden existir errores de transcripción, cuando este proceso se realiza en lugares donde existe alta concurrencia de visitantes como: hospitales, centros educativos, entidades gubernamentales, edificios administrativos, etc. (Oktaviadri & Kah Keat, 2019)

Con el uso de herramientas computacionales para el reconocimiento de caracteres OCR se puede extraer cierta información de un documento, así como también mediante algoritmos de detección de objetos se puede también extraer la fotografía del documento de una forma rápida, para luego, toda esta información almacenarla en una base de datos. También el uso de una placa edge computing es muy útil cuando se trata de diseñar y construir desde pequeñas aplicaciones inteligentes IoT, hasta robots más complejos, sistemas de visión artificial y reconocimiento de objetos, dispositivos que reaccionen de forma inteligente evaluando una serie de parámetros de sensores, pequeños vehículos autónomos, etc. (ohms, 2020)

Un estudio realizado en (Bhavana & Suvarna, 2021) determinó que el 94% de los rostros se identificaron correctamente utilizando 1000 imágenes de prueba para la detección mediante la técnica *Haar Cascade*, concluyendo que el detector de rostros basado en esta técnica tiene una alta precisión. Además, se evidenció que el nivel de identificación depende de la forma de las imágenes en la base de datos.

Una comparativa realizada en (Chincheró) analizan los diferentes motores OCR (Tesseract, Gocr, Ocrad, Tocr y Leadtools OCR SDK) en términos de eficacia, adaptabilidad, eficacia, licencia y evolución. Donde Tesseract presentó una capacidad de acierto por encima del 90%, como también de adaptarse a nuevos caracteres, nuevas tipografías, palabras frecuentes, caracteres permitidos y caracteres prohibidos.

Teniendo en cuenta lo antes mencionado, este trabajo pretende dar solución al problema de registro automático de información, que funcione con solo insertar una credencial tipo cédula de identidad en un elemento con ranura, se lea los datos de la persona como su nombre, número de identificación y fotografía. Luego, estos datos se envían a una base de datos en un servidor web en la cual se registra dicha información incluyendo fecha y hora de ingreso y abandono del lugar. Esta herramienta agilizará el trabajo que normalmente toma unos minutos al personal de guardia en tareas manuales de transcripción de datos del visitante.

Alcance del Proyecto

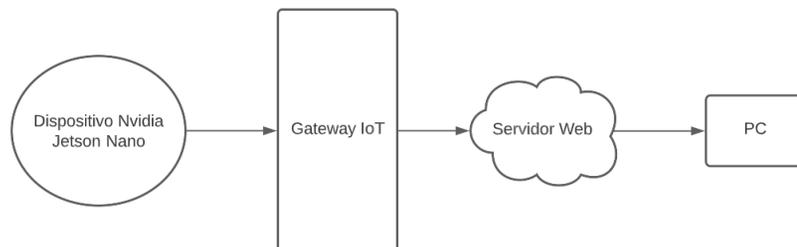
En el presente proyecto se plantea implementar un prototipo basado en el concepto de IoT que sea capaz de extraer información de nombre, número de identificación y fotografía de una credencial como la cédula de identidad utilizando algoritmos de detección facial y detección de caracteres tales como *Haar Cascade* y *Tesseract OCR* respectivamente, para luego enviar automáticamente esta información a un registro de una base de datos en un servidor web y mostrarla en una pantalla de monitor.

Se implementará un dispositivo con zócalo en donde se inserta la cédula de identidad por medio de ranura y con una cámara fotográfica en el interior se captura la imagen frontal de la cédula. Luego se realiza un procesamiento local de la imagen obtenida en una placa de desarrollo tipo Raspberry Pi o Nvidia Jetson Nano para extraer la información textual y de la fotografía, con el uso de herramientas de desarrollo basados en OpenCV, Python y Tensorflow.

Para la extracción de la información textual se realizará mediante el motor de reconocimiento óptico de caracteres Tesseract OCR, para la detección de la fotografía se realizará mediante el uso de un algoritmo de detección de objetos basado en el modelo Haar Cascade y algoritmos de aprendizaje AdaBoost, todo esto implementado en lenguaje de programación Python. En base a lo expuesto en el apartado anterior, se espera obtener una tasa de errores permitida igual o menor al 10% en el proceso de lectura del texto y una tasa igual o menor al 6% en la detección de la fotografía.

Figura 1

Arquitectura general del sistema



La placa de desarrollo que está conectada vía WiFi envía información a un servidor web enlazado a una base de datos remota que recibirá y registrará los datos de identificación del usuario, fecha y hora del ingreso o salida.

Objetivos

Objetivo General

- Desarrollar un prototipo para realizar la lectura y registro automático de información extraída de una credencial presentada por visitantes en puntos de control de acceso a un establecimiento.

Objetivos Específicos

- Implementar un subsistema electrónico de captura de imagen de la cédula de identidad de la persona visitante.
- Implementar en una tarjeta de desarrollo los algoritmos de OCR y detección de objetos, para extraer la información de la imagen de la cédula.
- Implementar un servidor web para la recepción, registro y consulta de datos de la persona visitante, enviados por la tarjeta de desarrollo.
- Realizar pruebas de funcionamiento para evaluar el desempeño del prototipo.

Capítulo II: Marco conceptual

La transformación digital y la automatización de procesos han impulsado el desarrollo de tecnologías avanzadas para la extracción de información a partir de documentos físicos. Uno de los desafíos más importantes en este contexto es la lectura automática y precisa de datos contenidos en documentos de identidad, esenciales para la identificación de las personas en diversos ámbitos sociales y gubernamentales.

El objetivo principal de este proyecto es desarrollar y evaluar un prototipo capaz de leer la información de cédulas de identidad ecuatoriana con mayor rapidez y precisión.

A continuación, se explorarán los fundamentos teóricos y conceptuales que sirvan de respaldo para el desarrollo de este proyecto, junto con principios clave que permitan un mayor enfoque en el transcurso del desarrollo de este. Se examinarán aspectos relevantes como el preprocesamiento de imágenes, la segmentación de texto, la seguridad en el manejo de información personal, entre otros.

Internet de las cosas

El Internet de las cosas (IoT) permite conectar una variedad de objetos, dispositivos, sensores mediante una conexión a internet.

Existe una amplia variedad de áreas y sectores donde se desarrollan aplicaciones IoT como:

- **Salud:** estos se centran más en monitorear la salud del individuo ya sea para controlar enfermedades o para mejorar el bienestar de las personas.
- **Seguridad:** Esta tecnología permite implementar sistemas de seguridad inteligentes en hogares, fábricas, oficinas, instituciones y otras. Al estar los dispositivos de seguridad conectados a internet permiten el monitoreo remoto constante además de la recopilación de información útil para mejorar el nivel de seguridad.

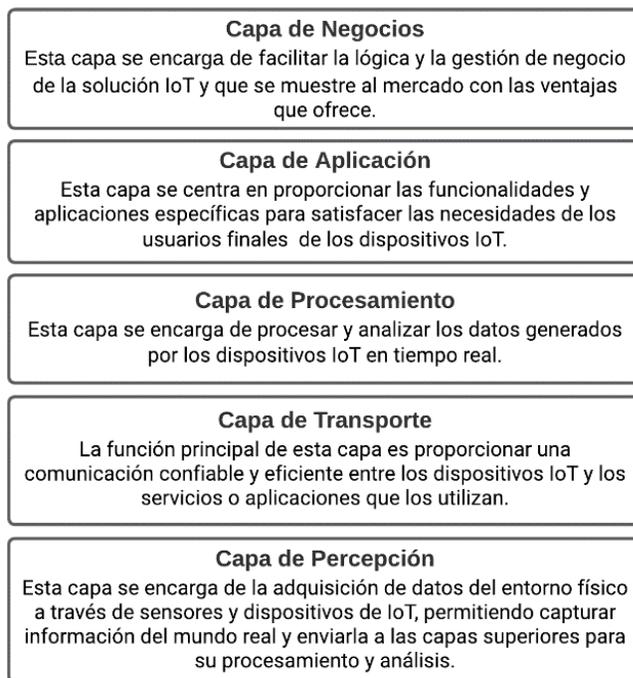
- **Energía:** En fábricas y establecimientos de estructura mediana o grande se utilizan para el monitoreo, control y optimización del uso de la energía de una manera más inteligente.
- **Gestión:** El IoT permite la integración de sistemas de iluminación, climatización, seguridad y gestión de energía en diversos lugares como edificios inteligentes, hogares, establecimientos, etc. Los sensores pueden recopilar datos en tiempo real sobre el consumo de energía, la ocupación de espacios y las condiciones ambientales.

En definitiva, existen un gran número de áreas y aplicaciones, como se describen en (Rose, Eldridge, & Chapin, 2022) en los que la tecnología IoT tiene presencia que, debido a las ventajas que se presentan por los avances en la potencia de cálculo, componentes electrónicos reducidos y las interconexiones de red estas ofrecen nuevas capacidades a soluciones que antes no eran posibles.

Para el desarrollo de cualquier sistema IoT se tiene en cuenta el funcionamiento de la arquitectura. Actualmente, esta arquitectura se compone principalmente de 3 capas: aplicación, red y percepción. Sin embargo, la arquitectura se complementa con el modelo OSI consolidando una estructura combinada. (Miao, Ting-Jie, Fei-Yang, Jing, & Hui-Ying, 2010)

Figura 2

Arquitectura para sistemas IoT



Nota. Figura tomada de (Miao, Ting-Jie, Fei-Yang, Jing, & Hui-Ying, 2010)

Como resultado se obtiene toda la arquitectura del ecosistema de IoT mostrada en la Figura 2, estas capas se componen de sensores, redes de comunicación, enrutadores, infraestructura de la nube, analítica de datos y seguridad.

Como se mencionó anteriormente, una de las áreas de aplicación del IoT es en los sistemas de seguridad, esta tecnología ha revolucionado la industria brindando nuevas oportunidades y capacidades para proteger personas, propiedades y activos. El IoT también se utiliza en sistemas de control de acceso para restringir y supervisar el acceso a áreas sensibles o restringidas. Los dispositivos IoT, como lectores de tarjetas, cerraduras electrónicas y paneles de control, se conectan a una red segura y permiten la autenticación y autorización de usuarios de forma remota. Esto puede incluir el uso de tarjetas de acceso, códigos PIN o incluso tecnologías biométricas, como reconocimiento facial o de huellas dactilares.

Visión Artificial

La visión artificial o visión por computadora se basa en emular la percepción visual que genera un ser humano. Esto se logra mediante algoritmos, modelos y técnicas que permiten a un computador obtener información a partir de captura de imágenes. (Szeliski, 2021)

En los últimos años, el desarrollo de este campo ha sido significativo teniendo importantes avances y contribuyendo a diferentes áreas de investigación como salud, tecnología, medicina, etc. Sin embargo, aún la visión humana es superior ya que aún no se logra tener características con el mismo nivel de detalle que se logra con el ojo humano.

Características

Existen varias características que se pueden definir en el campo de la visión artificial.

A continuación, se detallan las principales:

Detección y reconocimiento de Objetos. La detección de objetos implica poder reconocer e identificar la presencia de un objeto en una imagen, esto haciendo uso de descriptores locales y aplicando algoritmos de aprendizaje automático. Las condiciones en las que se presenta la imagen es el principal problema para que se puedan reconocer determinados objetos, por esta razón se desarrollan métodos y algoritmos que puedan identificar características que sean invariantes al escalado, transformación, rotación e iluminación de la imagen. (Lowe, 2002)

Los enfoques modernos de detección de objetos se basan en redes neuronales convolucionales (CNN) y algoritmos de aprendizaje automático, como R-CNN, Fast R-CNN, Faster R-CNN, YOLO (You Only Look Once) y SSD (Single Shot MultiBox Detector). Estos algoritmos han logrado un rendimiento significativo en términos de precisión y velocidad de detección.

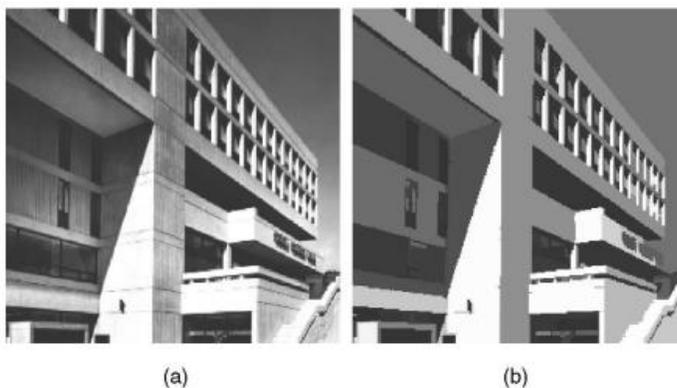
Segmentación de Imágenes. Para facilitar el análisis y comprensión del contenido visual, a la imagen se aplica divisiones en regiones o segmentos más pequeños que contengan características similares. Existen diferentes algoritmos para este proceso como el algoritmo de

Mean Shift (Comaniciu & Meer, 2002). Que agrupa píxeles o regiones que contengan características similares basándose en la similitud de la textura, el color u otras características que contengan estas regiones. En la Figura 3 se muestra el resultado de aplicar un proceso de segmentación.

Algunas de las principales ventajas del algoritmo son su capacidad para adaptarse a diversas distribuciones de datos, así como su resistencia al ruido y las variaciones de iluminación.

Figura 3

Imagen a) original b) segmentada



Nota. Imagen tomada de (Comaniciu & Meer, 2002)

Detección de bordes. Los bordes se presentan en los cambios abruptos de textura, color o intensidad de la imagen. La detección de bordes proporciona información sobre contornos y formas para luego ayudar a identificar y reconocer objetos presentes en la imagen.

Uno de los algoritmos ampliamente utilizado debido a su precisión y capacidad para reducir el ruido es el algoritmo de detección de bordes Canny (Figura 4) propuesto por (Canny, 1986) en 1986. Actualmente este algoritmo se fusiona con métodos y arquitecturas paralelas que mejoran la efectividad y la portabilidad del rendimiento del algoritmo, teniendo como resultado la reducción del tiempo de operación permitiendo aplicar estos algoritmos en el procesamiento de big data de imágenes. (Song, Li, & Zhou, 2023)

Figura 4

Detección de bordes con algoritmo Canny



Nota: Imagen tomada de (Song, Li, & Zhou, 2023)

Reconocimiento de patrones y características. El reconocimiento de patrones permite reconocer estructuras y patrones complejos con el objetivo de clasificar y asignar clases a las imágenes en diferentes categorías, esto se logra extrayendo información discriminativa aplicando algoritmos de aprendizaje automático. Este proceso es fundamental en aplicaciones como el reconocimiento facial, reconocimiento de caracteres, análisis de imágenes médicas y otros.

El reconocimiento de patrones y la clasificación son áreas de investigación y aplicación activas en la visión artificial y el aprendizaje automático. Con los avances en técnicas de extracción de características, aprendizaje profundo y computación de alto rendimiento, se han logrado grandes avances en el reconocimiento preciso y eficiente de patrones en una amplia variedad de aplicaciones. (O. Duda, E. Hart, & Stork, 2001)

Visión artificial inteligente y los sistemas autónomos

La visión artificial inteligente busca el desarrollo de algoritmos y técnicas que, mediante combinaciones de procesamiento de imágenes, aprendizaje automático, reconocimiento de patrones y otras técnicas logran extraer información significativa de las imágenes. Esta

información será utilizada por sistemas complejos para que puedan operar de forma autónoma y tomar decisiones de manera independiente sin necesidad de la intervención humana.

Al combinar la visión artificial inteligente en sistemas complejos se logra sistemas autónomos que pueden analizar situaciones y entornos para tomar decisiones de forma autónoma, lo que permite una amplia gama de aplicaciones en diferentes campos como la automoción, la robótica, la medicina, la manufactura, entre otros.

Una de las aplicaciones que está en auge en la actualidad, como menciona (Masmoudi, Ghazzai, & Frikha, 2019) son los vehículos autónomos, que tienen la atención de investigadores, la industria y el público en general. Estos vehículos combinan la visión artificial con el aprendizaje automático para simular a un conductor humano, esto hace que los investigadores desarrollen algoritmos de detección, reconocimiento y seguimiento de objetos que permitan al vehículo tomar decisiones para una correcta conducción y que no signifique un peligro en el entorno que se moviliza y no ocasione accidentes.

Figura 5

Detección de vehículos utilizando visión artificial usando (a) SVM, (b) YOLO.



Nota: Foto tomada de (Masmoudi, Ghazzai, & Frikha, 2019)

Otras de las aplicaciones que se presentan últimamente son la digitalización de datos, a inicios de este siglo la era de la digitalización empezó a surgir con mayor fuerza, es por esto por lo que en la actualidad la información está pasando de tener un impreso físico a digital. La digitalización de datos como facturas, documentos, o escritos físicos tienen la necesidad de desarrollar sistemas que permitan este traspaso con la mayor efectividad posible, es aquí donde la tecnología de visión artificial se extrapola al sistema para mejorar el texto dentro de la

imagen digitalizada para luego transmutar estas imágenes escaneadas y otras imágenes en texto. (Rakesh Kamisetty, y otros, 2022)

Reconocimiento Óptico de Caracteres (OCR)

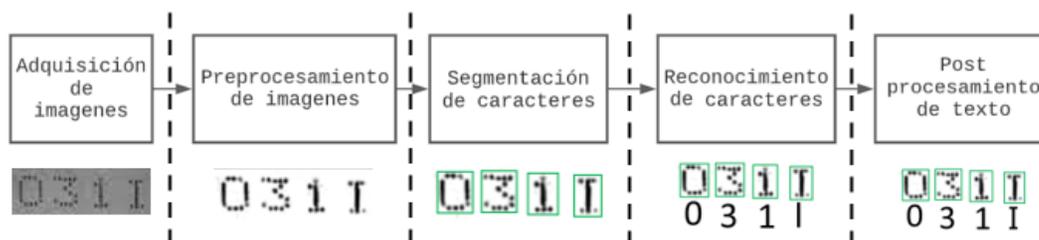
El reconocimiento óptico de caracteres (OCR) convierte imágenes que contienen texto en formato digital, reconoce cada carácter presente y extrae el texto para poder procesarlo o almacenarlo dependiendo de la aplicación que se quiera realizar.

Para identificar los caracteres, se aplican algoritmos de comparación de atributos como curvas, ángulos, tamaños y esquinas de caracteres presentes en la imagen con una base de datos conocida, estos algoritmos se han ido mejorando con los años para obtener mejores resultados, así también se han introducido nuevos métodos, pre y post procesos que eleven la probabilidad de éxito en la identificación de textos en imágenes. (Singh & Sachan, 2018)

Estructura básica de un algoritmo OCR

Figura 6

Estructura básica de un algoritmo OCR



El proceso de reconocimiento OCR implica varias etapas como se ilustra en la Figura 6 y son:

Adquisición de imágenes. Se obtiene una imagen o documento físico que contiene texto utilizando escáneres, cámaras o cualquier otro dispositivo de captura de imágenes.

Preprocesamiento de imágenes. Se aplican técnicas de procesamiento de imágenes para mejorar la calidad de la imagen, como el filtrado de ruido, la corrección de distorsiones y el ajuste del contraste.

Segmentación de caracteres. Se realiza la segmentación de la imagen para identificar y aislar los caracteres individuales. Esto puede implicar la detección y separación de líneas de texto, así como la segmentación de caracteres dentro de cada línea.

Reconocimiento de caracteres. Se aplica un algoritmo de reconocimiento para asignar una etiqueta o valor a cada símbolo de texto identificado en la imagen. Los métodos de reconocimiento de caracteres pueden incluir enfoques basados en reglas, redes neuronales, clasificadores estadísticos o técnicas de aprendizaje profundo.

Post procesamiento de texto. Se aplican técnicas adicionales para mejorar la precisión y coherencia del texto reconocido, como la corrección ortográfica y la verificación de contexto.

Métodos y Técnicas para OCR

Algunas técnicas y algoritmos comunes utilizados en el reconocimiento OCR incluyen motores OCR como Keras OCR, Easy OCR, y Tesseract OCR, métodos basados en aprendizaje profundo como Convolutional Neural Networks (CNN) y Recurrent Neural Networks (RNN), así como enfoques basados en características y clasificadores, como Support Vector Machines (SVM) y Hidden Markov Models (HMM). (Smith, 2007).

El reconocimiento OCR ha avanzado significativamente en los últimos años, especialmente con el uso de técnicas de aprendizaje profundo, lo que ha mejorado la precisión y la capacidad de reconocer diferentes tipos de fuentes y estilos de escritura.

Servicios de seguridad AAA

Los servicios de seguridad AAA (Authentication, Authorization and Accounting, por sus siglas en inglés) son fundamentales para el control y gestión del acceso a recursos y servicios de redes y sistemas, asegurando que solo usuarios autorizados puedan acceder a ellos.

Autenticación

Se utiliza para asegurarse de que el usuario es quien dice ser. Los métodos comunes de autenticación incluyen contraseñas, certificados digitales, tokens de seguridad o autenticación biométrica (como huellas dactilares o reconocimiento facial).

Autorización

Determina qué recursos o servicios puede acceder un usuario autenticado. La autorización se basa en políticas y reglas predefinidas que especifican los permisos y restricciones asociados a cada usuario o grupo de usuarios.

Contabilización

Implica el seguimiento y registro de actividades relacionadas con la seguridad, como los intentos de autenticación, las acciones de los usuarios y el acceso a recursos. Estos datos pueden ser utilizados para la gestión de red, la facturación del cliente y auditoría.

Arquitectura Web

La arquitectura web se refiere a la estructura y organización de un sistema o aplicación web. Se compone de diferentes componentes que trabajan juntos para ofrecer funcionalidades y servicios a los usuarios.

El desarrollo de aplicaciones web modernas involucra la colaboración entre dos componentes esenciales: el frontend y el backend.

Arquitectura Back end

Los componentes típicos de una arquitectura backend incluyen servidores, APIs, lógica de negocio, bases de datos, servicios de terceros y medidas de seguridad.

Arquitectura Front End

Es responsable de la interfaz de usuario y cómo se presenta la información al usuario. Utiliza tecnologías como HTML, CSS y JavaScript para crear la estructura, el diseño y la interactividad de la aplicación en el navegador.

Edge Computing

El Edge Computing (computación en el borde o computación perimetral) se centra en procesar y analizar datos más cerca de la ubicación donde se generan, o en el "borde" de la red, en lugar de enviarlos a un centro de datos centralizado o a la nube para su procesamiento.

En el modelo tradicional de computación, los dispositivos y sensores generan datos que luego se envían a servidores remotos o centros de datos para su procesamiento, análisis y almacenamiento. La computación de borde, por otro lado, tiene como objetivo reducir la latencia y el ancho de banda necesario procesando datos más cercanos al punto de generación. (Yu, y otros, 2018)

Arquitectura

En lugar de realizar el procesamiento de datos y la ejecución de aplicaciones en servidores distantes o en la nube, la arquitectura informática perimetral está diseñada para hacerlo más cerca del perímetro de la red. Esto implica distribuir la capacidad de computación y almacenamiento en dispositivos y servidores locales cercanos a la fuente de generación de datos.

Capa de Interfaz. En esta capa se encuentran todos los dispositivos finales (sensores, actuadores, etc) que se encargan de recopilar datos del entorno donde se encuentran.

Capa de Borde. En esta capa se ubican los servidores y dispositivos de borde que realizan el procesamiento de datos en el borde de la red. Estos servidores de borde también pueden tener numerosos requisitos de recursos, como el procesamiento de datos en tiempo real, el almacenamiento en caché de datos y la descarga de cómputo.

Capa de Nube. Los servidores se ubican en esta capa debido a que los servidores en la nube se implementan más alejados de los dispositivos finales, lo que provoca una importante latencia de transmisión en las redes. No obstante, los servidores en la nube en el entorno del cliente pueden proporcionar más poder de cómputo y más almacenamiento de datos.

Capítulo III: Diseño e Implementación

El presente capítulo describe el proceso de diseño e implementación de un prototipo que permita la lectura de información de las cédulas de identidad ecuatorianas, con el objetivo de automatizar el registro que se realiza actualmente en puntos de control de accesos a establecimientos. En este capítulo, se presentarán en detalle el diseño del prototipo en el cual se considera el procesamiento de imágenes, validación del documento, reconocimiento de información incluida en el documento de identidad, envío de información a la nube y posterior presentación de datos en el navegador.

La seguridad en los puntos de control de acceso es de vital importancia para garantizar la protección de establecimientos, bienes y personas. Al implementar medidas de seguridad adecuadas en los puntos de control de acceso, se contribuye a reducir el riesgo de incidentes y se protege de manera efectiva a las personas y los activos en estas áreas.

El registro manual de información en puntos de control de acceso puede ser un proceso lento, especialmente en lugares con un alto flujo de personas. Esto puede generar filas y congestionar los puntos de acceso, causando retrasos y molestias tanto para los usuarios como para el personal encargado del registro. Por otra parte, al ingresar la información de manera manual, existe una mayor probabilidad de cometer errores al transcribir nombres, números de identificación u otros datos relevantes. Estos errores pueden tener consecuencias negativas, como problemas de identificación y registros incorrectos. Además, el registro manual puede dificultar la implementación de controles y auditorías efectivas, ya que no se cuenta con una base de datos estructurada y confiable que permita un seguimiento preciso y remoto de los registros de acceso.

Descripción general de la solución

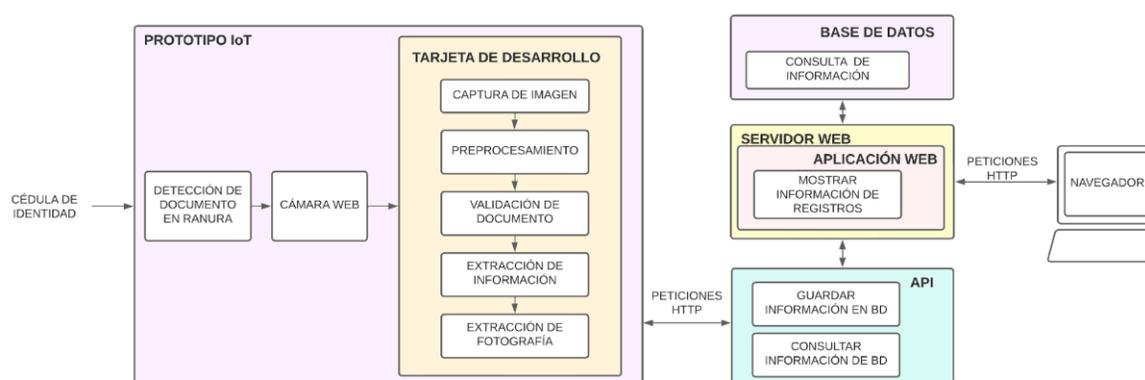
Para dar solución a esta problemática, se plantea un prototipo que realice la lectura automática de información contenida en las cédulas de identidad ecuatorianas mediante

reconocimiento de caracteres OCR y que dicha información se almacene en una base de datos remota con el fin de tener disponible en el lugar que se requiera.

En la Figura 7 se muestra, mediante un diagrama de bloques, en forma generalizada los procesos realizados para el desarrollo del prototipo, estos van desde que se introduce una cédula en la ranura hasta que se muestra la información en un navegador web.

Figura 7

Diagrama de bloques de procesos realizados para desarrollo del prototipo.



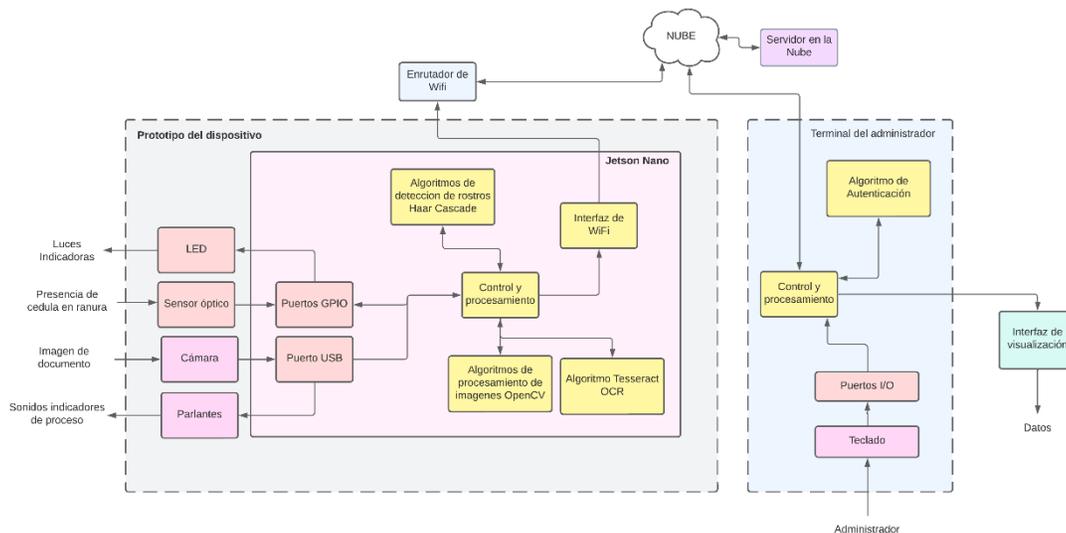
Para empezar, se detecta si se ingresó una cédula en la ranura, luego se realiza la captura de la imagen del documento mediante una cámara web y se envía a la tarjeta de desarrollo. Con la obtención de la imagen, se procede a realizar el preprocesamiento de imagen que permita mejorar la calidad de esta y tener mayor eficiencia en la lectura de la información.

Posteriormente, se realiza la validación del documento ingresado, esto consiste en identificar si el documento pertenece o no a un documento ecuatoriano y si es una cédula de identidad. Luego se clasifica por modelo: cédula con modelo actual y cédula con modelo anterior al actual, esto debido a que actualmente están en circulación dos modelos de cédulas ecuatorianas en el país. Realizados todos estos procesos, se procede a identificar y extraer la información presente en el documento utilizando reconocimiento óptico de caracteres OCR y por último obteniendo la fotografía utilizando algoritmos de detección de características.

Teniendo en cuenta todo lo descrito anteriormente, se elabora un diagrama de bloques con más detalle cómo se ilustra en la en la Figura 8 mostrando las diferentes conexiones y relaciones entre todo el prototipo tanto de hardware como de software.

Figura 8

Diagrama de bloques del sistema implementado en el prototipo



En la representación de bloques, los de color rojo se enfocan en elementos de hardware mientras los de color amarillo representan los elementos de software, todos estos a su vez están contenidos en bloques mayores representando a la tarjeta de desarrollo, prototipo del dispositivo de extracción de información y el terminal del administrador. A continuación, se describe a detalle el funcionamiento de cada bloque:

Procesos en el prototipo del dispositivo

En este proceso se detecta con un sensor óptico si ingresó una cédula en la ranura, luego se realiza la captura de la imagen del documento mediante una cámara web, se envía a la tarjeta de desarrollo para luego indicar mediante luces LED si el proceso está realizándose correctamente, adicionalmente se indica mediante voz el estado de los procesos realizados.

La tarjeta de desarrollo recibe la imagen que emite la cámara, se realiza la etapa de procesamiento de imagen para eliminar factores que pueden influir en la lectura de la

información y se procede a la validación del documento. Mediante el motor de reconocimiento de caracteres Tesseract OCR se extrae los nombres, apellidos y número de cédula del usuario y se realiza la detección y extracción de la imagen del rostro mediante el algoritmo para detección de rostros *Haar Cascade* que se explicaran a detalle más adelante.

Cuando se tiene toda la información necesaria se procede a enviar al servidor mediante protocolo HTTP y que se pueda almacenar en la base de datos.

Procesos en el terminal del administrador

El terminal del administrador recibe la información del servidor y está lista para ser mostrada, el administrador debe ingresar sus credenciales para acceder a la interfaz del sistema donde se muestra en una tabla los datos como nombres, apellidos, fotografía y fecha en la que se realizó el registro del ciudadano.

Finalmente, el prototipo cumple con características específicas como:

- **Conectividad:** El dispositivo está diseñado para operar bajo conexión WiFi o ethernet para permitir la comunicación y transferencia de datos con el servidor.
- **Portabilidad:** Su tamaño permite transportar e instalar fácilmente en cualquier lugar que tenga conexión WiFi.
- **Versatilidad de uso:** Se consolidan todos los instrumentos necesarios que permitan la lectura de información en una caja como única herramienta visible al usuario para que sea más fácil de manejar.
- **Bajo costo:** Dado que se desea un dispositivo accesible a cualquier empresa o institución, se utilizaron dispositivos básicos que cumplan con los requisitos mínimos para llevar a cabo este proyecto.

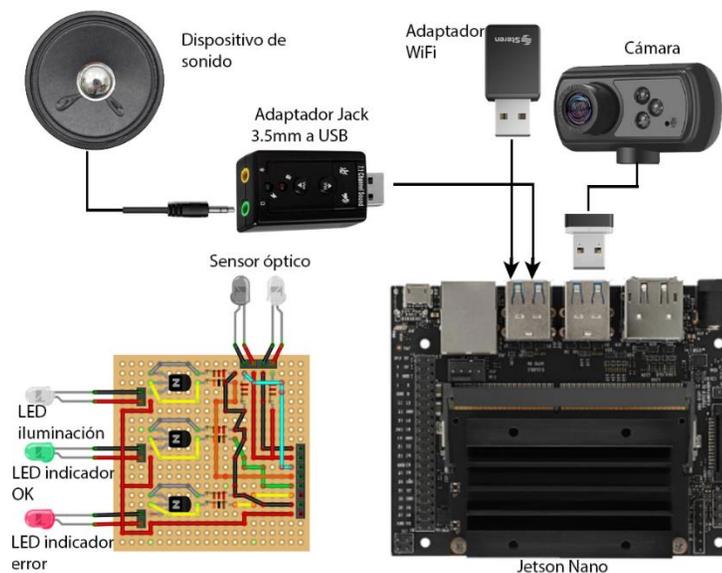
Desarrollo Hardware

Prototipo IOT

El prototipo IoT se compone de diferentes dispositivos como cámara, sensor óptico, tarjeta de desarrollo Jetson nano, LEDs indicadores, dispositivo de sonido y circuitos de

acondicionamiento. En la Figura 9 se observa cómo se distribuyen las conexiones entre cada dispositivo y hacen referencia a:

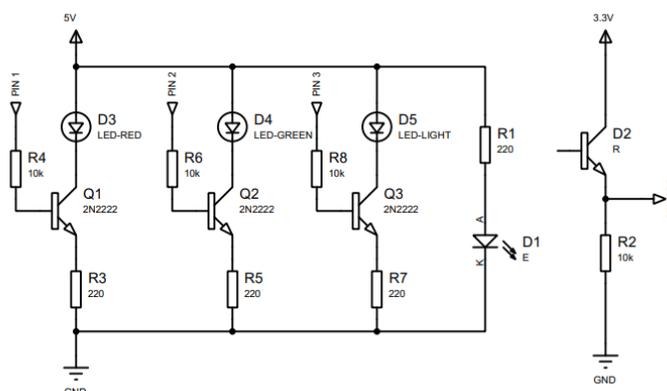
- La alimentación principal se conecta a la tarjeta de desarrollo con una alimentación de 12v, esta se encarga de distribuir la alimentación a los sensores y actuadores mediante sus salidas de 5v y 3.3v.
- Los LEDs indicadores se alimentan de 5v, estos son tomados de los pines de alimentación de la tarjeta de desarrollo. Para el control de encendido y apagado se realiza un circuito de acondicionamiento con un transistor NPN funcionando como interruptor debido a que la señal que se tiene a la salida de los pines GPIO un voltaje de 3.3V que en este caso si se conecta directamente se enciende el LED, pero no a su brillo normal. El LED verde es indicador de procesos correctos y el LED rojo indica si ocurrió un error durante el proceso. El LED de iluminación proporciona luz dentro de la caja para poder capturar la imagen.
- Los LED infrarrojos emisor y receptor del sensor óptico son alimentados con 3.3V y estos también se toman de los pines de alimentación de la tarjeta, se colocan resistencias de protección para cada uno.
- Se coloca un parlante de 8 ohmios como dispositivo de sonido, se agrega un adaptador Jack 3.5mm a USB debido a que no se tiene una salida de audio directamente de la tarjeta.
- La cámara está conectada directamente a la entrada USB.
- Entre los dispositivos también se agrega un adaptador WiFi ya que la tarjeta de desarrollo solo tiene conexión ethernet.

Figura 9*Diagrama de conexiones del prototipo.*

Circuito de acondicionamiento. Para los LED indicadores, de iluminación y para el sensor óptico se elaboró un circuito de acondicionamiento compuesto por transistores y resistencias como se muestra en la Figura 10. Este circuito fue necesario implementar debido a que en las salidas GPIO de las tarjetas de desarrollo se tiene 3.3V los cuales están por debajo del voltaje necesario para el encendido normal de un LED. En cuanto al sensor, está compuesto de un fotodiodo emisor con resistencia 220 ohmios en serie para protección y un receptor con una resistencia 10k en configuración pull-up, de este punto se toma la salida a la entrada GPIO de la tarjeta de desarrollo por lo que su alimentación es de 3.3V que es lo máximo permitido.

Figura 10

Circuito de acondicionamiento para LEDs indicadores y sensor.

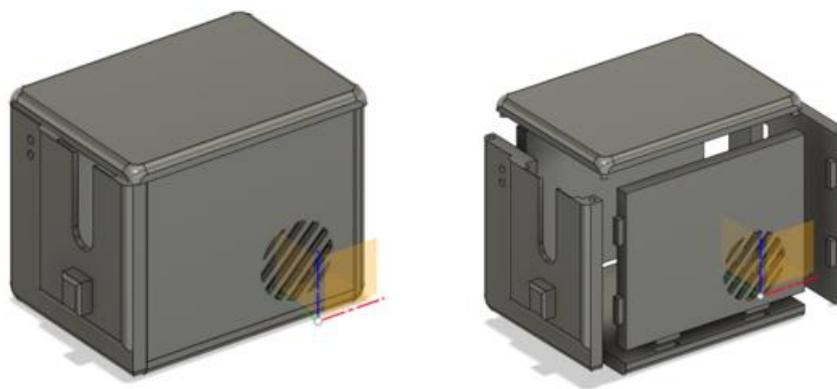


Carcasa del prototipo

Se realiza el diseño en 3D en el programa Fusion 360 que consiste en una caja rectangular con medidas internas de 10x12x15 cm. La carcasa está diseñada para ser funcional y proporcionar un espacio de almacenamiento seguro y protegido para los componentes que se alojarán en ella. Además, el diseño permite ensamblar la caja por partes al estar dividida en cada pared como se muestra en la Figura 11.

Figura 11

Carcasa diseñada en Fusion 360

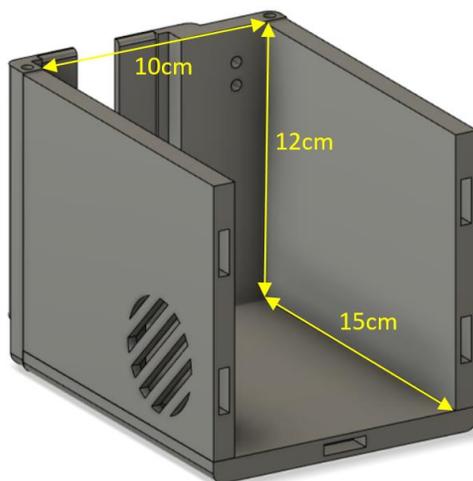


La carcasa presenta paredes sólidas de 1cm de grosor y 20% de relleno en su interior (Figura 12) que brindan resistencia y estabilidad al diseño. Cada lado de la caja tiene una

longitud de 12 cm de altura, 10 cm de largo y 15 cm de profundidad, se conecta de manera precisa con ángulos rectos para formar una estructura sólida.

Figura 12

Medidas internas de la carcasa



En uno de los lados de la carcasa, se ha incorporado una ranura de tamaño adecuado para permitir la inserción de la cédula de identidad ecuatoriana. La ranura está ubicada en la parte frontal de la caja y se extiende a lo largo de una de las paredes, también se agrega un corte de 2x6 cm proporcionando un acceso conveniente para insertar y retirar los documentos. En la parte trasera se realiza un corte en forma rectangular donde se van a colocar las salidas USB y de alimentación de la tarjeta de desarrollo.

La carcasa tiene una tapa desmontable que se ajusta perfectamente sobre la parte superior, asegurando que los objetos almacenados en su interior estén protegidos de la suciedad, el polvo y los daños externos. La tapa se mantiene en su lugar mediante un sistema de pines.

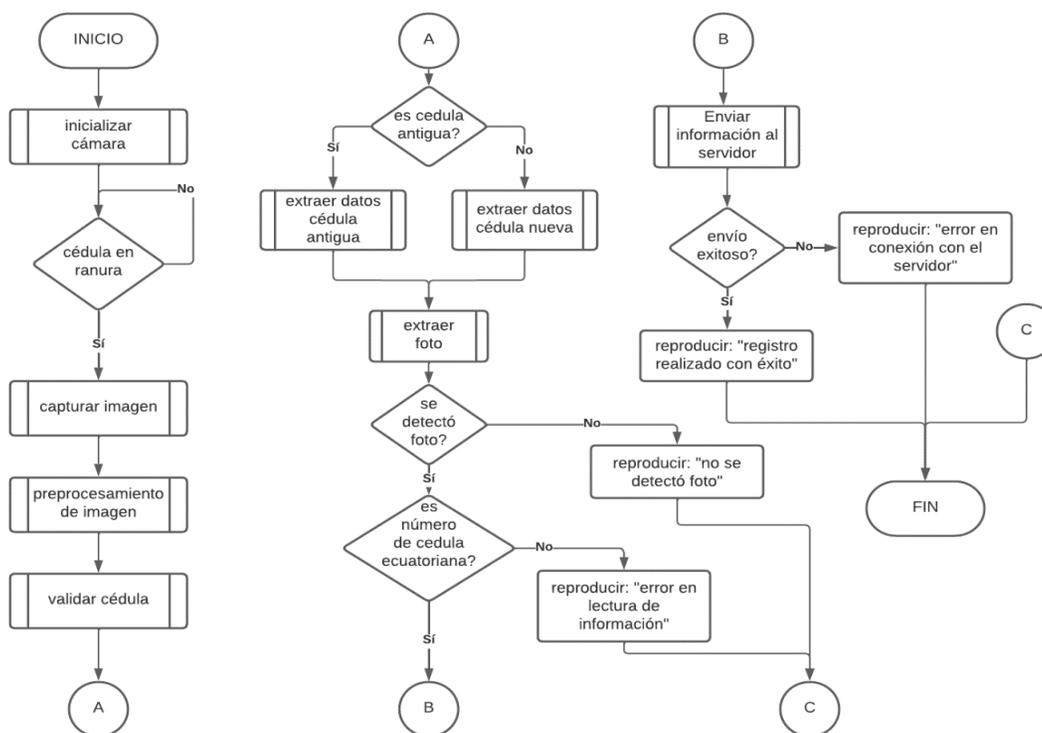
Desarrollo Software

A continuación, se presentan todos los procedimientos desarrollados a nivel de software para la lectura de la información de las cédulas de identidad ecuatoriana. Luego de obtener la imagen tomada por la cámara, se realizan etapas de preprocesamiento de imágenes, luego se

aplica el reconocimiento de caracteres para extraer la información del documento y posteriormente enviar al servidor. En la Figura 13 se muestra el diagrama de flujo del proceso que se realiza desde la detección de la cédula de identidad en la ranura hasta el envío de la información al servidor en la nube.

Figura 13

Diagrama de flujo para realizar la lectura y envío de información.



Captura de imagen

Al insertar el documento de identidad en la ranura, el sensor óptico detecta el objeto y se realiza la captura de la imagen con la cámara disponible. En este caso se tomó como ejemplo dos posibilidades: una cédula de identidad con diseño actual y con orientación normal de lectura (Figura 14a), y una cédula de identidad ecuatoriana con diseño antiguo que se encuentra en una posición invertida en relación con la orientación normal de lectura (Figura 14b).

Figura 14

Imagen obtenida de cámara a) cedula nueva b) cedula antigua



Preprocesamiento

Un adecuado preprocesamiento de imágenes nos permitirá obtener condiciones óptimas y maximizar la precisión y el rendimiento del algoritmo OCR. A continuación, se detalla las diferentes técnicas utilizadas en el preprocesamiento de imágenes, como la mejora de la calidad de imagen, la eliminación de ruido, el ajuste de contraste, la corrección de distorsiones geométricas, entre otras. Además, analizaremos cómo estas técnicas contribuyen a obtener condiciones óptimas para la aplicación del algoritmo OCR, asegurando una extracción precisa y confiable de la información contenida en los documentos.

Corrección de distorsión. El efecto barril es una distorsión óptica que curva las líneas rectas hacia afuera desde el centro de la imagen. Para eliminar esta distorsión se realiza los siguientes pasos:

- 1) Se realiza la calibración de la cámara para determinar la matriz de la cámara y los parámetros de distorsión, se realiza con las siguientes consideraciones:
 - Se usan múltiples imágenes.
 - Un objeto conocido
 - Diferentes posiciones
 - Girada

- Inclínada
- Desplazada

Figura 15

Banco de imágenes para calibración de cámara



- 2) Se importan las librerías y se define el tamaño del tablero de ajedrez y las dimensiones de las imágenes.

```
import numpy as np
import cv2 as cv
import glob

chessboardSize = (10,6) #Dimensiones del tablero de ajedrez
                           (número de esquinas internas)
frameSize = (720,480) #dimensiones de las imágenes

criterio = (cv.TERM_CRITERIA_EPS + cv.TERM_CRITERIA_MAX_ITER,
            30, 0.001)
```

Se define una variable “criterio” para especificar los criterios de terminación de una operación iterativa que utilizaremos más adelante, donde tiene los siguientes parámetros:

- *cv.TERM_CRITERIA_EPS*: Especifica el criterio de parada basado en la precisión deseada. La iteración se detendrá si la precisión alcanza el valor especificado en el tercer parámetro.
- *cv.TERM_CRITERIA_MAX_ITER*: Especifica el criterio de parada basado en el número máximo de iteraciones permitidas. La iteración se detendrá si el número de iteraciones alcanza el valor especificado en el segundo parámetro.

- 30: Es el número máximo de iteraciones permitidas. En este caso, se establece en 30 iteraciones.
 - 0.001: Es la precisión deseada. En este caso, se establece en 0.001, lo que significa que la iteración se detendrá si la precisión alcanza 0.001.
- 3) Se crean listas para almacenar los puntos de referencia 3D del mundo real y los puntos 2D de las esquinas detectadas en las imágenes:

```
objp = np.zeros((chessboardSize[0]
                *chessboardSize[1], 3), np.float32)

objp[:, :2]=np.mgrid[0:chessboardSize[0], 0:chessboardSize[1]].T
.reshape(-1, 2)

objPoints = []

imgPoints = []
```

- 4) Se lee las imágenes y detecta las esquinas del tablero de ajedrez en cada una:

```
images = glob.glob('calibracion/*.jpg')

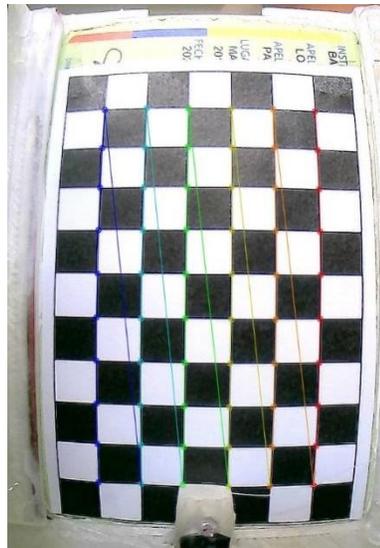
for image in images:
    print(image)
    img = cv.imread(image)
    gray = cv.cvtColor(img, cv.COLOR_BGR2GRAY)

    ret, corners = cv.findChessboardCorners(gray,
chessboardSize, None)
    print(ret)

    if ret == True:
        objPoints.append(objp)
        corners2 = cv.cornerSubPix(gray, corners, (11,11), (-
1,-1), criteria)
        imgPoints.append(corners)
        cv.drawChessboardCorners(img, chessboardSize,
corners2, ret)
        cv.imshow('img', img)
        cv.imwrite('img.png', img)
        cv.waitKey(1000)
```

Figura 16

Resultado de esquinas identificadas del tablero de ajedrez.



5) Se Calibra la cámara y se obtiene los parámetros de corrección:

```
ret, cameraMatrix, dist, rvecs, tvecs=
cv.calibrateCamera(objPoints, imgPoints, frameSize, None,
None)
```

Donde los parámetros devueltos son los siguientes:

- *ret*: Un valor booleano que indica si la calibración de la cámara fue exitosa o no. Si la calibración fue exitosa, *ret* será True; de lo contrario, será False.
- *cameraMatrix*: La matriz de la cámara, también conocida como matriz de proyección. Es una matriz 3x3 que contiene los parámetros intrínsecos de la cámara, como el punto principal, la distancia focal y los coeficientes de escala. Esta matriz se utiliza para transformar puntos 3D en coordenadas de imagen 2D.
- *dist*: Los coeficientes de distorsión de la cámara. Estos coeficientes representan las distorsiones radiales y tangenciales que pueden ocurrir en las imágenes capturadas por la cámara. Los coeficientes de distorsión son necesarios para corregir la distorsión en las imágenes.

- *rvecs*: Un arreglo de vectores de rotación. Cada vector representa los parámetros de rotación necesarios para alinear el sistema de coordenadas del mundo real con el sistema de coordenadas de la cámara en cada imagen calibrada.
- *tvecs*: Un arreglo de vectores de traslación. Cada vector representa los parámetros de traslación necesarios para alinear el sistema de coordenadas del mundo real con el sistema de coordenadas de la cámara en cada imagen calibrada.

Los valores obtenidos luego de la calibración son los siguientes:

Matriz de cámara:

$$(f_x \ 0 \ c_x \ 0 \ f_y \ c_y \ 0 \ 0 \ 1)$$

Donde:

- f_x y f_y son las longitudes focales en píxeles. Estos valores indican la relación entre la distancia focal de la cámara y el tamaño del píxel en la imagen.
- c_x y c_y son las coordenadas del punto principal, que representa el centro óptico de la cámara.

El valor de la matriz de cámara devuelta luego de la calibración es el siguiente:

$$(7.00025706 * 10^3 \ 0 \ 8.11629264 * 10^2 \ 0 \ 7.42261178 * 10^3 \ 4.44845279 * 10^2 \ 0 \ 0 \ 1)$$

Parámetros de distorsión:

$$(k_1 \ k_2 \ k_3 \ p_1 \ p_2 \)$$

Donde:

- k_1, k_2 y k_3 son los coeficientes de distorsión radial que controlan la curvatura de la imagen.
- p_1 y p_2 son los coeficientes de distorsión tangencial que controlan la deformación de la imagen.

$$(-17.2113686 - 4.06123836 * 10^2 - 1.89876262 * 10^{-1} - 2.58905805 * 10^{-1} 2.63725473 * 10^4)$$

Para realizar la prueba de eliminación de la distorsión se aplica el siguiente código:

```
img = cv.imread('imagen.jpg')
h, w = img.shape[:2]
newCameraMatrix, roi = cv.getOptimalNewCameraMatrix(cameraMatrix,
dist, (w,h),1,(w,h))

dst = cv.undistort(img, cameraMatrix, dist, None,
newCameraMatrix)
x, y, w, h = roi
dst = dst[y:y+h, x:x+w]
cv.imwrite('resultado.png', dst)
```

Como se mostró en la Figura 14, la imagen que se captura tiene un efecto de distorsión, característico de la construcción de los lentes de las cámaras. Para eliminar esta distorsión se aplica el algoritmo de corrección para el efecto de barril diseñado, obteniendo el resultado mostrado en la Figura 17. Como se observa, se tiene una imagen con los bordes rectos y las direcciones de los textos sin distorsión.

Figura 17

Imagen sin distorsión efecto de barril a) cédula nueva b) cédula antigua



Redimensionar. La imagen que se obtiene de la cámara es de 720x480 píxeles, se realiza un escalamiento para ampliar la imagen al doble de las medidas originales. Esto es para mejorar la resolución de la imagen, mejorar y suavizar mejor los bordes debido a que los caracteres presentes en el documento de identidad son pequeños, al no aplicar este procedimiento no obtendremos buenos resultados al aplicar el algoritmo OCR recordando que este algoritmo se basa en la detección de los bordes de cada carácter para el reconocimiento.

Para redimensionar la imagen se utiliza la función `cv2.resize()`, esta recibe como parámetros: una imagen, un nuevo ancho y un nuevo alto para luego redimensionar la imagen a las nuevas dimensiones especificadas.

Escala de grises. Al convertir la imagen a *escala de grises*, las características de contraste y textura se vuelven más prominentes como se observa en la Figura 18. Esto puede beneficiar el proceso de segmentación y extracción de texto, ya que los bordes y detalles importantes pueden ser más fácilmente detectados y resaltados por los procesamientos posteriores.

Figura 18

Imagen en escala de grises



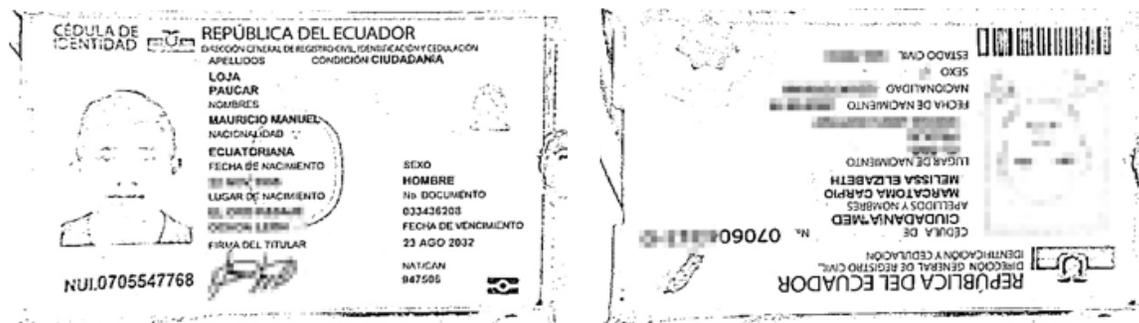
Para este caso se utiliza la función `cv2.cvtColor()` que recibe como parámetros la imagen original y convierte a escala de grises. Al trabajar en escala de grises, las características de contraste y textura se vuelven más prominentes. Esto puede beneficiar el

proceso de segmentación y extracción de texto, ya que los bordes y detalles importantes pueden ser más fácilmente detectados y resaltados

Segmentación. Se aplica un umbral adaptativo a la imagen en escala de grises con el objetivo de segmentar la imagen en regiones de interés o áreas de interés basadas en un valor de umbral, en la Figura 19 se observa que se obtiene la información de interés, en este caso el texto, pero aún se pueden observar áreas con ruido de imagen que pueden afectar el rendimiento del algoritmo OCR.

Figura 19

Imagen con filtro de umbral adaptativo



Se utiliza la función **cv2.adaptiveThreshold()** ya que este a diferencia de un umbral global fijo, se calcula localmente en cada región de la imagen, lo que permite obtener mejores resultados en imágenes con iluminación irregular o contrastes variables.

Este proceso permite eliminar objetos que no son de interés en la imagen, por ejemplo, fondos y hologramas presentes en la cédula de identidad ecuatoriana, dejando presente solo textos que es la información útil para la lectura.

Eliminación de ruido. Para eliminar el ruido presente en la imagen del documento se utilizan varios filtros en el mismo orden que se detallan a continuación:

- **cv2.dilate():** Esta función se utiliza para expandir los píxeles de los objetos en una imagen. La dilatación se realiza tomando el máximo valor de intensidad de los píxeles en una vecindad específica y asignándole al píxel central. Esto tiene el

Se realiza la operación de **erosión (Erode)** para eliminar pequeños detalles no deseados o ruido, lo que resulta en una imagen más limpia y simplificada como se presenta en la Figura 21. Este proceso de erosión también ayuda a separar caracteres cercanos cuando están conectados debido a artefactos o falta de calidad en la imagen.

Figura 21

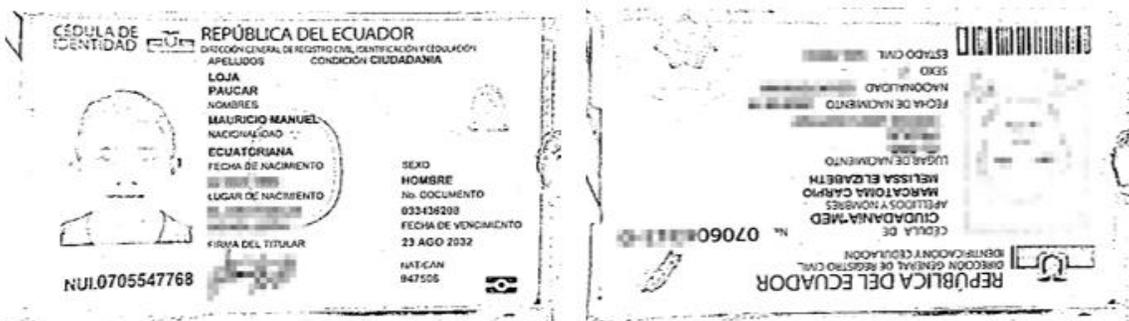
Imagen con filtro erosión



El **Desenfoque Mediano (MedianBlur)** aplicado permite mejorar la calidad de la imagen como se observa en la Figura 22, haciéndola más suave y eliminando detalles no relevantes, como consecuencia esto ayuda a reducir el ruido y las irregularidades en la imagen, preservando al mismo tiempo los bordes y detalles importantes.

Figura 22

Imagen con filtro desenfoque mediano



Se aplica la operación de **Negación de bits (bitwise_not)**, este proceso ayuda a resaltar y mejorar el contraste de los objetos de interés, como los caracteres, al convertir el

fondo en blanco y los caracteres en negro. En este caso se utiliza para luego realizar un proceso de dilatación y lograr que los caracteres tengan un grosor mayor al actual, el resultado se presenta en la Figura 23.

Figura 23

Imagen con filtro de negación de bits



Se realiza un nuevo proceso de **Dilatación** debido a que existen aún caracteres con el grosor insuficiente para poder obtener una buena identificación de parte del algoritmo OCR. En la Figura 24a se observa que la palabra “CÉDULA DE IDENTIDAD” se puede identificar fácilmente mientras que en la Figura 32a se logra leer con mayor dificultad.

Figura 24

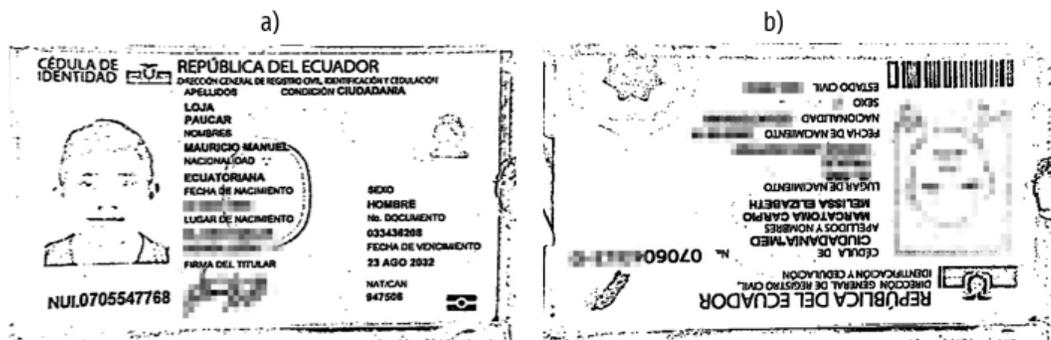
Imagen con filtro dilatación



Finalmente, se invierte los colores para obtener imagen con caracteres negros y el fondo blanco como se ilustra en la Figura 25 para luego realizar el proceso de clasificación y posterior lectura de información.

Figura 25

Resultado de preprocesamiento de imagen a) cédula nueva b) cédula antigua

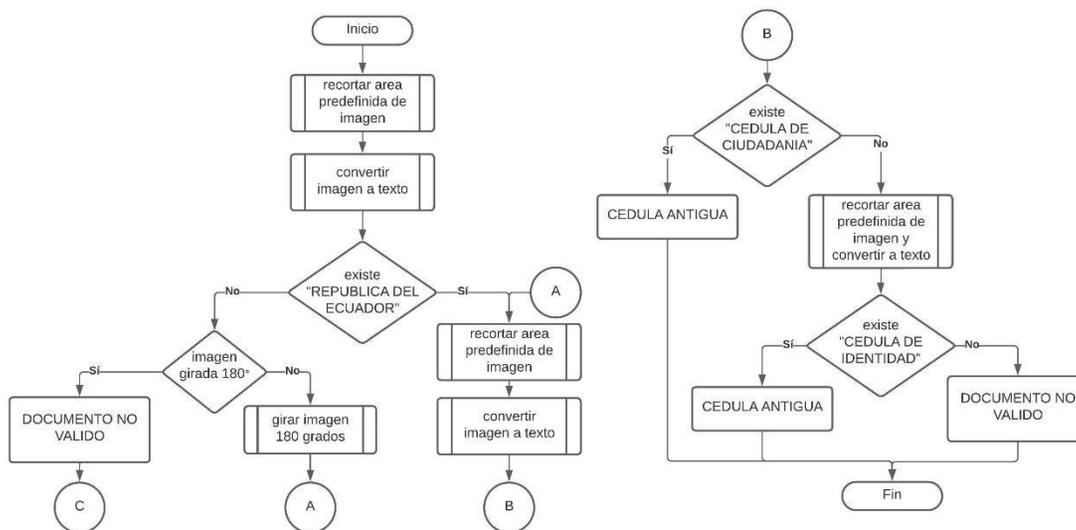


Validar documento

Esta etapa se encarga de validar el documento ingresado, en la Figura 26 se explica mediante un diagrama de flujo este proceso que es el siguiente: se valida si el documento pertenece o no a una cédula de identidad ecuatoriana, luego de validar se procede a corregir la orientación si esta no está correcta en relación con la posición de la cámara. Luego se procede a identificar si es una cédula de identidad con diseño antiguo o con diseño actual.

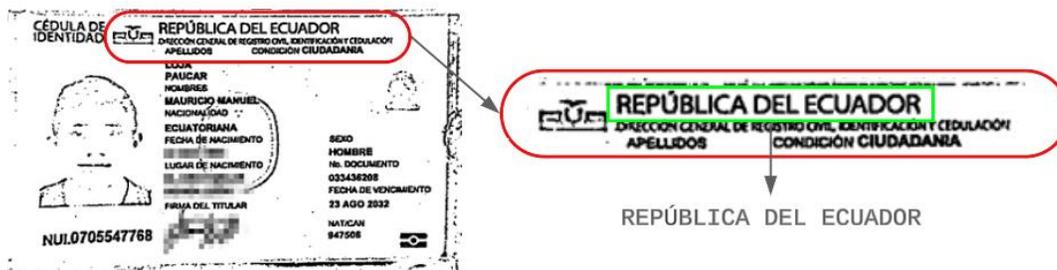
Figura 26

Diagrama de flujo para validación de documento



Caso 1: Cédula de identidad nueva y posición normal. En este caso la cédula está en posición normal, por lo que el algoritmo recorta la parte superior de la imagen, procesa la imagen con el algoritmo OCR y la convierte a texto, luego se realiza una búsqueda de las palabras “REPÚBLICA DEL ECUADOR”.

Figura 27 Identificación de documento ecuatoriano en cédula nueva



Se encuentra las palabras buscadas como se ilustra en la Figura 27 determinando que se trata de un documento ecuatoriano y se continúa a la identificación del tipo de cédula, para esto se buscan dos identificadores en dos áreas diferentes de la imagen: para el caso de la cédula actual, se busca las palabras “CÉDULA DE IDENTIDAD” que está presente en la parte superior izquierda y, para el caso de la cédula antigua se busca la palabra “CEDULA DE CIUDADANÍA” en la parte media de la imagen.

En este caso se encontró la palabra “CÉDULA DE IDENTIDAD” mostrado en la Figura 28 por lo que el algoritmo concluye que se trata de una cédula de ciudadanía ecuatoriana del modelo actual.

Figura 28

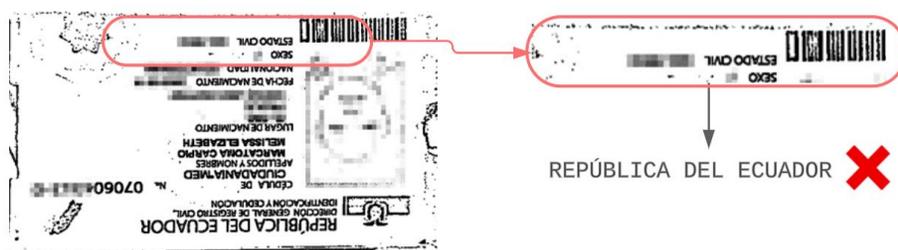
Identificación de cédula de identidad en cedula nueva



Caso 2: Cédula de identidad antigua y posición invertida. Para este caso la cédula de identidad está colocada en posición invertida con relación a la posición de la cámara, el algoritmo recorta la parte superior de la imagen, procesa la imagen con el algoritmo OCR y le convierte a texto, luego se realiza una búsqueda de las palabras “REPÚBLICA DEL ECUADOR” sin tener éxito como se muestra en la Figura 29.

Figura 29

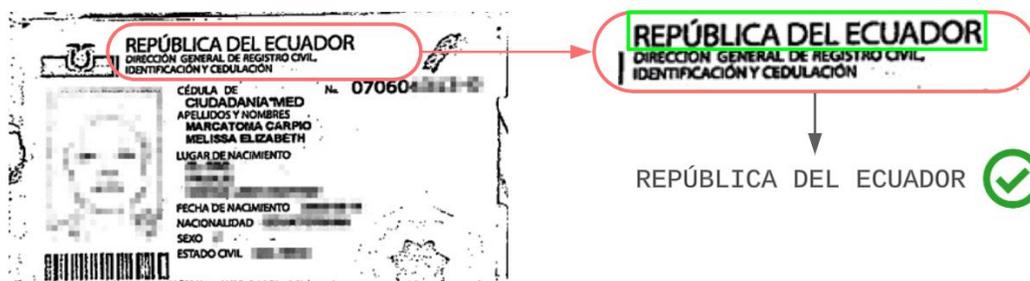
Identificación de documento ecuatoriano en cedula antigua (orientación incorrecta)



Al no tener resultado, se procede a girar 180 grados a la imagen para luego realizar el mismo proceso de búsqueda, esta vez se obtiene una respuesta positiva como se ilustra en la Figura 30 ya que si existe el identificador en la imagen y se valida que sea un documento ecuatoriano. Para rotar la imagen utilizando OpenCV en Python, se utiliza la función `cv2.warpAffine()`. Esta función aplica una transformación de afinidad a una imagen, lo que permite realizar rotaciones, traslaciones y escalados. Como parámetro para la función anterior se obtiene la matriz de rotación con la función `cv2.getRotationMatrix2D()`.

Figura 30

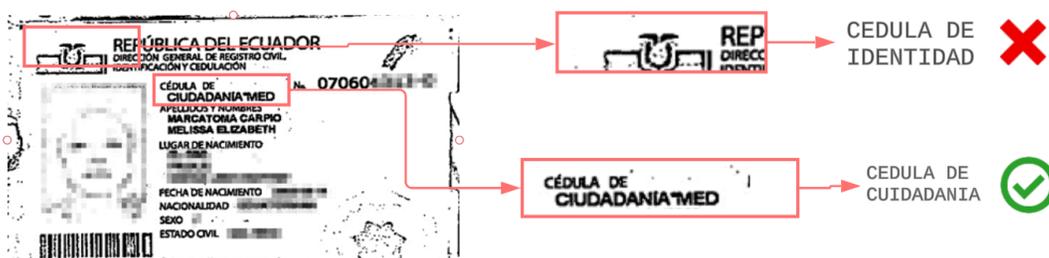
Identificación de documento ecuatoriano en cedula antigua



Se realiza el proceso de identificar si es una cédula nueva o antigua, para esto se recortan las dos áreas de interés de la imagen y se convierten a texto mediante el algoritmo Tesseract OCR. En este caso no se encuentra los identificadores “CÉDULA DE IDENTIDAD” pero si se obtiene un resultado positivo en la búsqueda de las palabras “CEDULA DE CIUDADANÍA” como se observa en la Figura 31.

Figura 31

Identificación de cédula de identidad en cédula antigua



Lectura de información

Extraer datos cédula antigua. Para extraer los datos como nombres, apellidos y número de cédula se toma el área donde se encuentra esta información, se recorta y se realiza un preprocesamiento de negación de bits a la imagen para luego aplicar la detección de bordes, esto permite identificar y segmentar el área que contiene caracteres del resto de la imagen.

Para encontrar y extraer los contornos de los objetos en la imagen se utiliza la función `cv2.findContours()` utilizando como parámetros de configuración los siguientes:

- **cv2.RETR_EXTERNAL:** Recupera solo los contornos externos, descartando los contornos internos contenidos dentro de ellos.
- **cv2.CHAIN_APPROX_SIMPLE:** Almacena sólo los puntos finales de los segmentos de línea que forman los contornos. Por ejemplo, si los contornos forman un cuadrado, solo se almacenan cuatro puntos.

Esta función nos devuelve dos listas de valores que son:

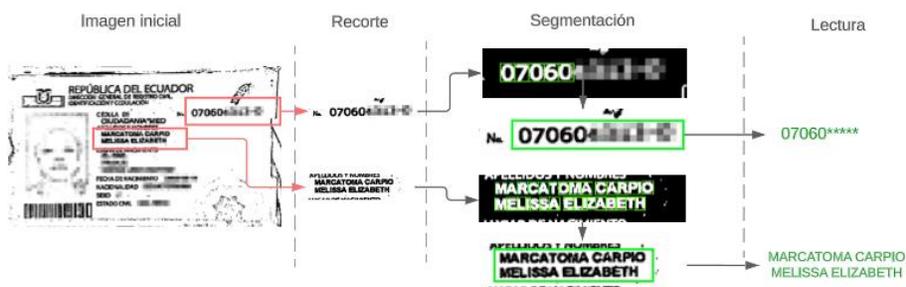
- **contours:** La lista de contornos encontrados en la imagen.
- **hierarchy:** La información de jerarquía de los contornos encontrados.

De estos dos valores de retorno, el que se utiliza es la lista de contornos, con esto se generan rectángulos contenedores o bounding boxes donde el área de cada rectángulo debe cubrir totalmente al segmento de píxeles donde existe un carácter cuya relación de aspecto (ancho vs alto) se determinan, estos rectángulos contienen las coordenadas específicas permitiendo extraer solo los caracteres de interés. Este proceso se realiza para excluir ruido como puntos o rayas que pueden interpretarse como texto válido.

Luego de detectar cada carácter se realiza un nuevo recorte al área segmentada donde se identificaron bordes que cumplan la condición de relación de aspecto que se establece entre unos rangos aproximándose al tamaño de las letras que se pueda encontrar en la imagen. Los rectángulos finales se observan en la Figura 32. Posterior a esto, ya teniendo los caracteres segmentados y definido el área donde se encuentran, se toma las coordenadas del área y se recorta de la imagen inicial donde el fondo es blanco y las letras de color negro. Cada recorte por separado se lee mediante Tesseract configurado para tratar la imagen de entrada como “un solo bloque uniforme de texto” ya que en este caso tenemos todo el contenido en un solo bloque sin divisiones y sin elementos gráficos o diseños complicados.

Figura 32

Etapas para la extracción de información en cédula antigua.



Cada recorte es leído por Tesseract y guardados en variables tipo string, estas lecturas pueden ser correctas o incorrectas por lo que se requiere realizar una limpieza antes que se retorne el resultado final.

Para los nombres y apellidos se eliminan: todos los caracteres especiales, letra minúsculas y números, mostrado en Tabla 1 ya que en el formato que se imprime en la cédula solo constan letras mayúsculas. Para el caso del número de cédula, se realiza una limpieza similar dejando solo caracteres que pertenezcan a números, adicional se reemplazan letras que tengan un parecido con los números como se muestra en la Tabla 2.

Tabla 1

Caracteres eliminados o reemplazados en el texto final de nombres y apellidos

Símbolo a letra	
Original	Reemplazo
Caracteres especiales (#\$%&*)	Vacío
Números	Vacío
Letras minúsculas	vacío

Tabla 2

Caracteres eliminados o reemplazados en el texto final de numero

Símbolo a letra	
Original	Reemplazo
O	0
S	5
A	4
I	1
G	6
T	7
Caracteres especiales (#\$%&*)	vacío

Lectura de datos cédula nueva. Para la lectura de los datos en la cédula nueva se realiza el mismo procedimiento anterior con: recorte, segmentación y lectura, en este caso se tiene dividido los nombres y los apellidos por lo que se debe procesar en diferentes áreas para

luego unirlos y así poder enviar al registro de la nube. En la Figura 33 se ilustra las etapas que se realizan para finalizar con la lectura de la información.

Figura 33

Etapas para la extracción de información en cédula nueva.



Extraer foto. Se utiliza el *clasificador Haar Cascade* para detectar la fotografía del documento en la imagen, esta función devuelve las coordenadas donde se detecta un rostro las mismas que se utilizan para recortar de la imagen original del documento.

Figura 34

Extracción de foto en a) cédula nueva b) cedula antigua



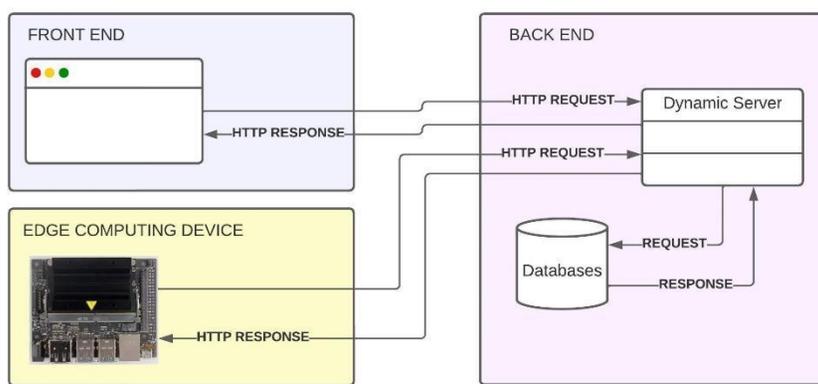
Diseño de la aplicación Web

La arquitectura de diseño de esta aplicación web se basa en una estructura de cliente-servidor, donde el front-end y el back-end desempeñan roles específicos para brindar una experiencia de usuario fluida y además, al mantener separados se obtiene comodidad y flexibilidad en el desarrollo de cada componente.

La comunicación entre el backend, frontend y la tarjeta jetson nano es mediante RESTful API, donde el flujo de peticiones y respuestas en HTTP (Figura 35) sigue un patrón básico en el que un cliente envía una solicitud al servidor y el servidor envía una respuesta con los datos solicitados o el estado de la operación.

Figura 35

Flujo peticiones/respuestas HTTP

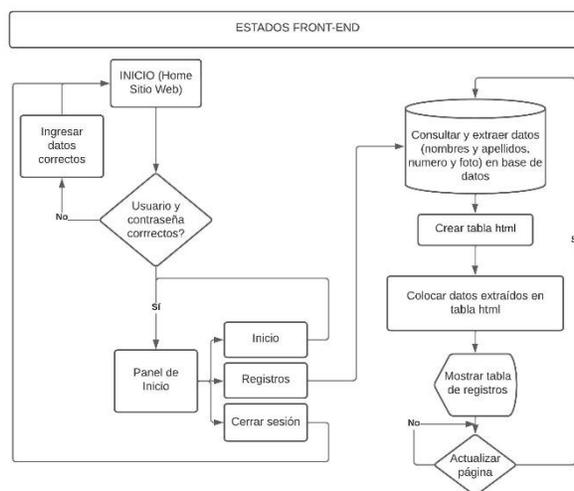


A continuación, se describe detalladamente la implementación tanto del front-end como del back-end:

Front End. Para definir la arquitectura que se ejerció para el desarrollo de la interfaz del usuario utilizando HTML, JavaScript y CSS, se establece el flujo de actividades sobre el comportamiento del sistema frente a posibles acciones que se tomen, esto representado en forma de diagrama de flujo.

Figura 36

Diagrama de flujo front-end



Los diferentes componentes de la Figura 36 cumplen con las funcionalidades básicas de recibir, comprobar y actualizar los diferentes datos del sistema de registro de información de visitas en puntos de control de acceso.

Back-end. El backend se desarrolla en el entorno de ejecución Node.js, utilizando el framework Express para manejar las rutas y crear APIs RESTful. Cuando se ejecute la aplicación, esta escuchará las peticiones realizadas y, en caso de existir esa ruta establecida resolverá respondiendo los datos solicitados o el estado de operación, caso contrario devolverá un mensaje de error.

Las rutas establecidas se detallan a continuación:

- **files:** esta ruta recibirá la petición referente a los datos enviados desde la tarjeta Jetson Nano, esta solicitud contiene los datos de nombres y apellidos, número de cédula y foto. Luego se procede a insertar toda la información recibida en la base de datos, devuelve el estado de la operación.

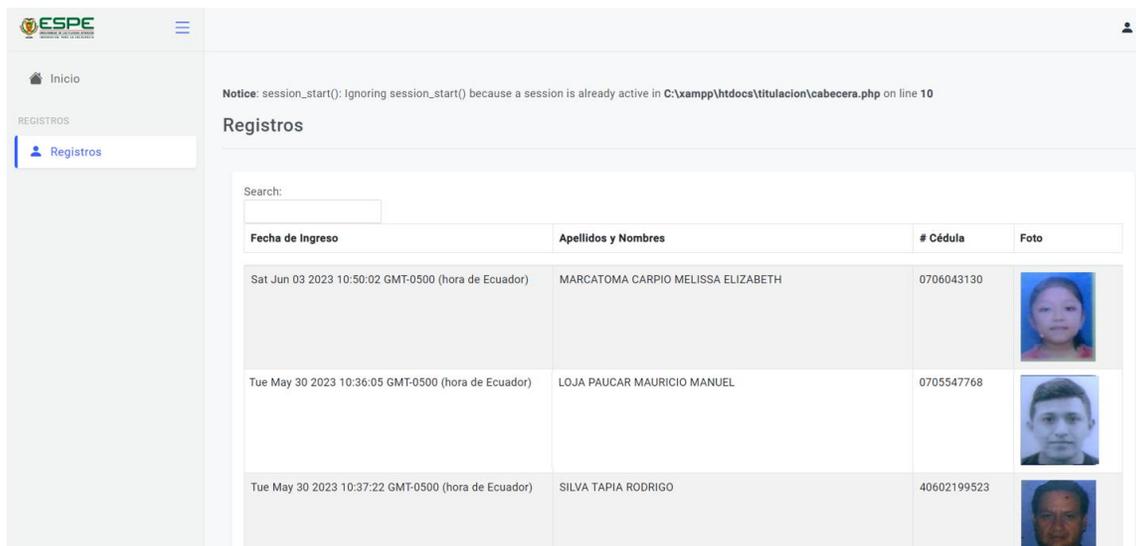
- **user:** esta ruta recibirá la solicitud que se encarga de validar los datos de ingreso de los usuarios al sistema. Se recibe un usuario y contraseña para luego verificar en la base de datos la existencia o no del usuario, devuelve un booleano.
- **register:** esta ruta procesa la petición referente a la información contenida en la base de datos

Teniendo la información necesaria para realizar un registro, se procede a enviar mediante solicitud http POST al servidor a través de un formulario web. Este formulario contiene: nombres y apellidos (cadena de texto), número (cadena de texto) y fotografía (archivo de imagen).

Para verificar que la información se ha subido correctamente, se ingresa al sitio web -> pestaña registros y se observa los datos insertados en una tabla. Estos datos están ordenados en columnas como: fecha de ingreso, apellidos y nombres, número de cédula y fotografía de la persona que ingresa como se muestra en la Figura 37.

Figura 37

Visualización de información en página web



Notice: session_start(): Ignoring session_start() because a session is already active in C:\xampp\htdocs\titulacion\cabecera.php on line 10

Registros

Search:

Fecha de Ingreso	Apellidos y Nombres	# Cédula	Foto
Sat Jun 03 2023 10:50:02 GMT-0500 (hora de Ecuador)	MARCATOMA CARPIO MELISSA ELIZABETH	0706043130	
Tue May 30 2023 10:36:05 GMT-0500 (hora de Ecuador)	LOJA PAUCAR MAURICIO MANUEL	0705547768	
Tue May 30 2023 10:37:22 GMT-0500 (hora de Ecuador)	SILVA TAPIA RODRIGO	40602199523	

El desarrollo de código implementado para este proyecto y explicado en esta sección se alojó en un repositorio digital, el enlace se ubica en los anexos de este documento.

Capítulo IV: Montaje del Prototipo

En el presente capítulo, se aborda el montaje del prototipo planteado en este proyecto de titulación, centrado en la integración de un dispositivo Edge Computing con otros elementos como sensores y actuadores que conforman el prototipo que permita la lectura de información de cédulas ecuatorianas.

El Edge Computing, un paradigma emergente en el campo de la computación, ha ganado relevancia en los últimos años debido a la necesidad de procesar datos en tiempo real y reducir la latencia en aplicaciones sensibles al tiempo. La capacidad de realizar cálculos y análisis en el extremo de la red, más cerca del origen de los datos, permite una toma de decisiones más rápida y eficiente, sin depender de una conexión constante a la nube.

En este sentido, NVIDIA ofrece una placa de desarrollo como Jetson Nano, una placa especialmente para aplicaciones de inteligencia artificial y aprendizaje automático ofrece una potencia de procesamiento excepcional y capacidades de inferencia de modelos entrenados. Su reducido tamaño y consumo de energía la convierten en una opción ideal para implementaciones en dispositivos compactos, como la caja de prototipo propuesta.

A lo largo de este capítulo, se describe en detalle el proceso de montaje, incluyendo la configuración del dispositivo Jetson Nano, la selección y conexión de los componentes adicionales necesarios para el funcionamiento del prototipo.

Dispositivo Edge Computing

Existen varios dispositivos Edge computing actualmente en el mercado que permiten un alto nivel de procesamiento de datos y a precios accesibles. Para la implementación de este proyecto, se optó por la tarjeta Nvidia Jetson Nano de 2GB. Este dispositivo ofrece un alto rendimiento para ejecutar cargas de trabajo de IA modernas gracias a su GPU integrada que la convierte en ideal para aplicaciones como procesamiento y segmentación de imágenes, detección de objetos, procesamiento de voz, entre otras orientadas al campo de visión artificial.

Instalación del sistema operativo

El sistema operativo está alojado en una tarjeta microSD. La Jetson Nano no tiene un almacenamiento interno incorporado, por lo que el sistema operativo y los archivos del sistema se instalan y ejecutan desde una tarjeta microSD.

Antes de iniciar con la instalación se debe tener dos componentes esenciales:

- *Tarjeta microSD:* Debe ser suficientemente rápida y tener el almacenamiento necesario dependiendo del proyecto, se recomienda una tarjeta UHS- de 32GB como mínimo.
- *Fuente de alimentación:* Se recomienda una fuente de alimentación micro USB que pueda entregar 5V- 2ª en el puerto de la tarjeta de desarrollo.

Teniendo listos estos componentes se procede a realizar la instalación con los siguientes pasos:

- *Escribir imagen en tarjeta microSD:* Se descarga la imagen SDK del sitio oficial de Nvidia. Se formatea la tarjeta microSD y luego, mediante el software Etcher se escribe la imagen del Jetson Nano Developer Kit en la tarjeta
- *Configuración y primer arranque:* Una vez que la tarjeta microSD esté lista, se procede a configurar el kit de desarrollador. Esta configuración se puede realizar de dos formas: con periféricos o en modo “sin interfaz” a través de una conexión desde otra computadora. Se coloca la tarjeta microSD en la ranura de la tarjeta de desarrollo, se alimenta mediante el cable micro USB y se enciende, se iluminará un LED verde indicando el encendido. Se selecciona el idioma, nombre de usuario, contraseña y nombre del dispositivo y el tamaño de partición de la aplicación: se recomienda seleccionar el tamaño máximo.

Configurar software y librerías necesarias

Luego de realizar el primer arranque se realiza la instalación y configuración del software necesario y librerías para poner en funcionamiento los algoritmos diseñados, se realiza lo siguiente:

- Mediante el terminal o consola se instala la librería *GPIO* que permite el acceso a los pines de entrada y salida de la tarjeta de desarrollo.
- Se instala la librería *OpenCV* que contiene las herramientas necesarias para el procesamiento de imágenes.
- Otra de las librerías necesarias es *pytesseract*, el contenedor de Tesseract para Python. Además, se debe descargar e instalar el motor de reconocimiento óptico de caracteres *Tesseract*.
- Adicional se descarga el archivo "*haarcascade_frontalface_default.xml*" que contiene los parámetros y las reglas de decisión aprendidas durante el proceso de entrenamiento del clasificador en cascada para la detección de rostros frontales.

Acople del sistema y funcionamiento general

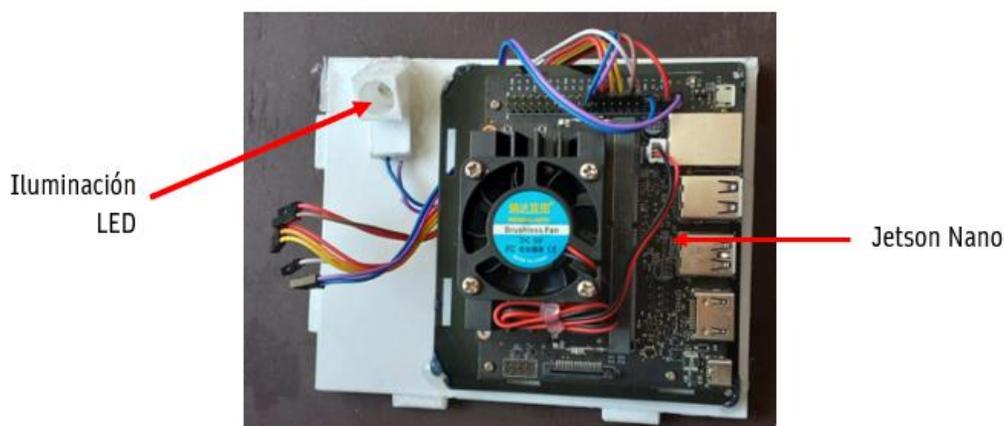
Se realiza la colocación y conexión de los diferentes elementos que conforman el sistema, esto con el objetivo de lograr una interconexión y coordinación entre los componentes para un correcto funcionamiento. En este contexto, el acople del sistema garantiza que: los sensores, actuadores, la cámara, el software y otros elementos clave trabajen en conjunto de manera efectiva, permitiendo la captura precisa de imágenes, el reconocimiento de caracteres y la extracción de información relevante de las cédulas de identidad.

En este apartado, explicaremos en detalle cómo se establece esta colocación y posterior conexión entre los componentes, cómo se implementa el algoritmo diseñado para la lectura de información y cómo es el funcionamiento en general del prototipo. El adecuado acople del sistema permitirá el correcto funcionamiento y la obtención de resultados confiables en el proceso de lectura de información mediante OCR.

Como se observa en la Figura 38, el primer componente se coloca la tarjeta Jetson nano, esta se ubica en una pared de la caja, adicional a esto también se añade la iluminación LED que va conectado a la misma tarjeta para su alimentación.

Figura 38

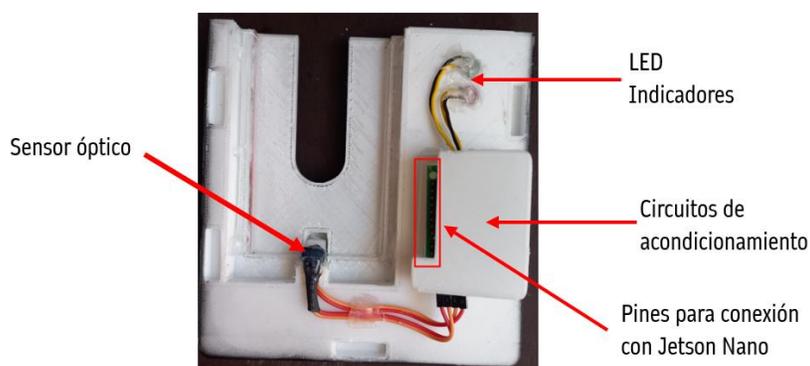
Acople de Jetson Nano e iluminación LED



En la pared frontal de la caja, como se ilustra en la Figura 39 se colocan: los indicadores LED verde y rojo, el sensor óptico de presencia y los circuitos de acondicionamiento. Estos últimos estarán conectados a los demás componentes y tendrán las salidas mediante pines para la conexión con la tarjeta Jetson Nano.

Figura 39

Conexión de sensores e indicadores



En la pared izquierda se coloca la cámara, el parlante de sonido que estará conectado a un circuito amplificador de sonido, como se observa en la Figura 40. Estos dispositivos se alimentan mediante conexión USB, y para la salida de audio desde la tarjeta se realiza mediante un Jack 3.5mm.

Figura 40

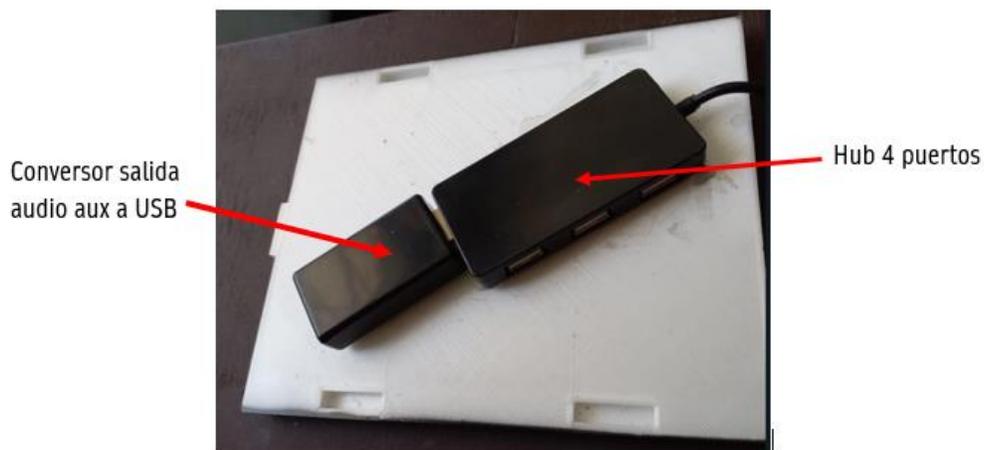
Acople de cámara y dispositivo de sonido en pared izquierda



Finalmente, en la pared inferior de la caja (Figura 41) se coloca el adaptador de audio Aux a USB y una extensión HUB de puertos USB para que la conexión se pueda realizar con la Jetson Nano debido a que esta solo dispone de 3 puertos USB.

Figura 41

Adaptador de audio y Hub 4 puertos colocados en pared inferior (piso de caja)



Luego de colocar todos los componentes, se procede a el armado de la caja prototipo. Esta estructura, como se mencionó anteriormente, al tener una construcción tipo lego permite trabajar por separado las paredes para luego unir y formar la estructura completa. En la Figura 42 se ilustra la estructura final, donde se puede observar la unificación de todos los componentes en una sola caja.

Figura 42

Integración de todos los componentes necesarios para el funcionamiento del prototipo.



Capítulo V: Evaluación del prototipo

En este capítulo, se presentan los detalles de las pruebas realizadas y se analizan los resultados obtenidos. Este proceso de evaluación es esencial para validar la confiabilidad y el rendimiento del prototipo en general.

Para este propósito se realizó la lectura de 99 cédulas de identidad ecuatoriana, de las cuales 45 pertenecen al diseño antiguo, 48 cédulas con diseño actual y 6 documentos que no pertenecen a una cédula de identidad. Esta prueba se realizó tomando como muestra a estudiantes simulando el ingreso a un establecimiento como la Universidad de las Fuerzas Armadas donde existe una gran afluencia de personas.

Desempeño general

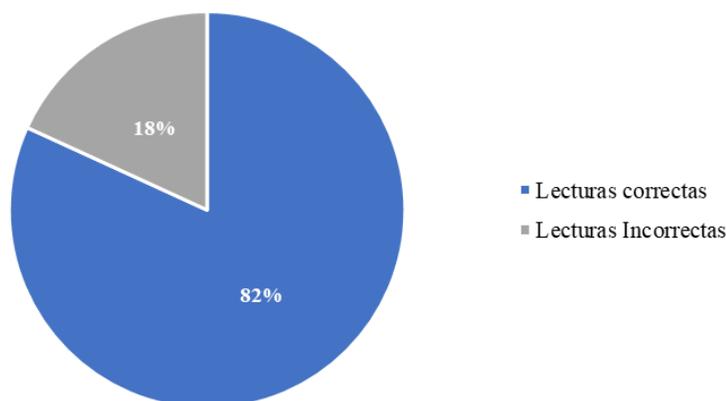
Esta métrica se define como la cantidad de aciertos o errores sobre la cantidad total de las mediciones. Partiendo de un conjunto de 99 cédulas de identidad se obtienen los siguientes resultados:

- **Total de lecturas:** 99
- **Lecturas Correctas:** 81
- **Lecturas Incorrectas:** 18

La Figura 43 expone gráficamente el desempeño general del sistema donde se muestra la clasificación de resultados: las lecturas correctas que son las que no presentan variación o errores con respecto a la información original impresa en el documento de identidad, las lecturas incorrectas que presentan errores ya sea en uno o más caracteres en los nombres, apellidos y número de cédula.

Figura 43

Desempeño general del prototipo



Pruebas y resultados

Con el fin de tener una comprensión más amplia del desempeño del prototipo, se muestran a detalle los resultados de las pruebas de funcionamiento del sistema en general, con el propósito de determinar en qué medida el sistema cumple con los requerimientos establecidos y cómo se comporta en diversas situaciones. A través de un análisis detallado de los resultados, se busca identificar las ventajas y desventajas del prototipo.

Como se ha expuesto en los capítulos anteriores, el objetivo de este trabajo ha sido desarrollar un prototipo para realizar la lectura y registro de información para credenciales de identidad de forma automática, disminuyendo los errores y el tiempo que pueden conllevar este proceso si se lo realiza manualmente.

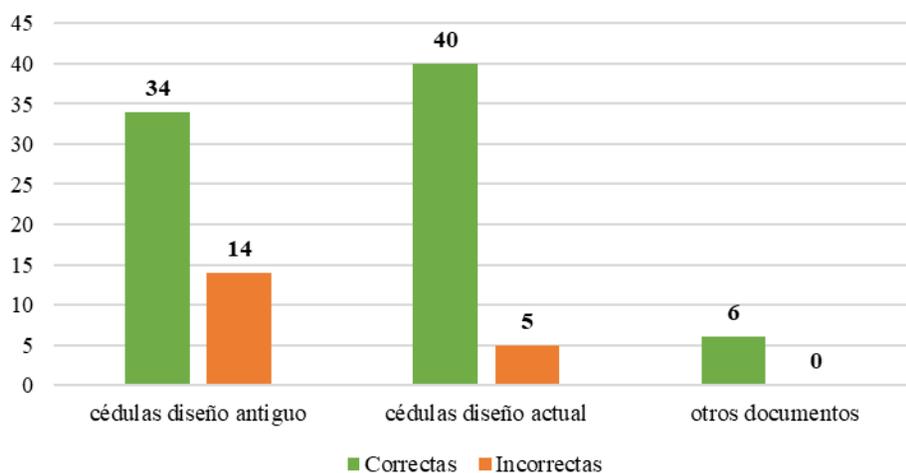
Para evaluar el prototipo de lectura de información en general, es importante considerar una combinación de métricas que abarque distintos aspectos del rendimiento y la confiabilidad del sistema. A continuación, se realiza una evaluación para determinar dichas métricas.

En la lectura de las cédulas antiguas se tuvo un total de 34 lecturas exitosas y 14 con información errónea, en el caso de las cédulas con diseño actual se tuvo un total de 40 lecturas correctas de las 45 que se utilizaron. También se utilizaron documentos que no eran cédulas de

identidad para comprobar si existe errores en la clasificación, en este caso no se tuvo errores dando 6 identificaciones correctas como se ilustra en la Figura 44.

Figura 44

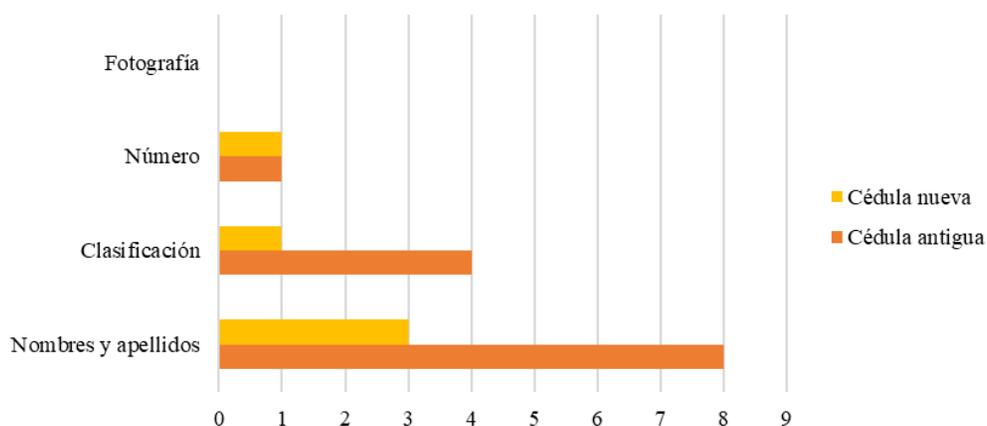
Lecturas realizadas en prototipo de lectura



De los errores observados, el mayor número de lecturas erróneas se da en la lectura de nombres y apellidos de las cédulas antiguas con 8 lecturas en total, en la Figura 45 también se presentan los errores los errores que existieron en la clasificación donde el prototipo no logró detectar como cédula de identidad al documento válido.

Figura 45

Clasificación de errores por tipo de información leída



Evaluación de rendimiento del prototipo

Partimos de una matriz de confusión (Tabla 3) que clasifica los resultados obtenidos en las pruebas de funcionamiento del prototipo.

Tabla 3

Matriz de confusión

		PREDICCIÓN	
		Positivo	Negativo
REAL	Positivo	Verdadero Positivo (TP)	Falso Negativo (FN)
	Negativo	Falso Positivo (FP)	Verdadero Negativo (TN)

Donde:

$$Precisión = \frac{TP}{TP + FP} \quad (1)$$

La precisión permitirá medir la proporción de información correctamente reconocidos respecto al total de reconocidos por el OCR. Una alta precisión indica que hay pocos errores de reconocimiento.

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

El Recall mide la proporción de cédulas correctamente reconocidos respecto al total de reales presentes en el conjunto de datos. Un alto recall indica que el OCR es capaz de identificar la mayoría de las cédulas reales.

$$F1 - Score = 2 * \left(\frac{Precisión * Recall}{Precisión + Recall} \right) \quad (3)$$

El F1-Score es una métrica que combina tanto la precisión como el recall en una sola medida. Es útil cuando hay un desbalance entre las clases de interés (por ejemplo, más caracteres negativos que positivos) y proporciona una visión equilibrada del rendimiento del sistema.

Estas métricas permiten evaluar el rendimiento que tiene el reconocimiento de información del prototipo de lectura con OCR en cédulas de identidad. Cuanto más altas sean

la *precisión* y el *recall*, mayor será la confiabilidad del sistema en reconocer correctamente los caracteres presentes en las imágenes de cédulas de identidad.

Tabla 4

Resultados obtenidos de la lectura de información

		PREDICCIÓN	
		Información reconocida	Información no reconocida
REAL	Información real	81	18
	Información no real	0	81

Precisión

Con esta métrica se mide la proporción de cédulas reconocidas correctamente entre todas las cédulas reconocidas. Aplicando la Ecuación 1 se obtiene el siguiente resultado:

$$Precision(\%) = \frac{81}{81} * 100\% \approx 100\%$$

Recall

Se define esta métrica como la proporción de cédulas reconocidas correctamente entre el total de cédulas reconocidas. Aplicando la ecuación 2 se obtiene el siguiente resultado:

$$Recall(\%) = \frac{81}{99} * 100\% \approx 81.81\%$$

F1-Score

Se realiza una combinación de las dos métricas anteriores para calcular el F1-score que nos permitirá obtener una medida ponderada, esta considera tanto la precisión como la recuperación del motor OCR

$$F1 - Score \approx 2 * \frac{1 * 0.8181}{1 + 0.8181} \approx 0.8999 \approx 89.99\%$$

Tiempo de respuesta del prototipo

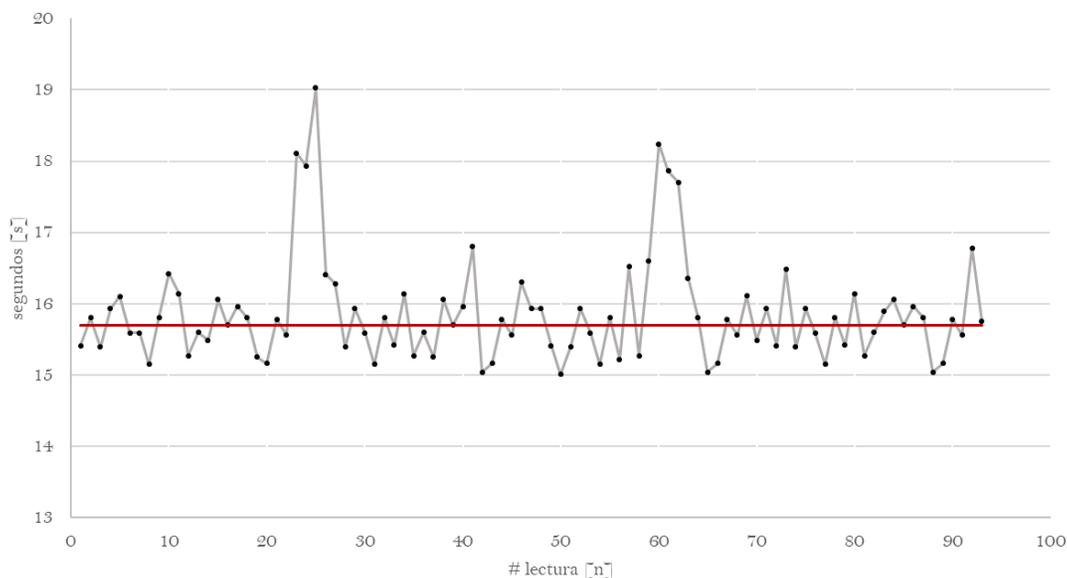
El objetivo de este proyecto es automatizar la lectura de información de visitantes en puntos de control de acceso a establecimientos, por esta razón el tiempo que se toma en

realizar este proceso debería ser lo más corto posible para que no existan complicaciones como la acumulación de visitantes en los puntos de control.

En la Figura 46 se muestran los tiempos obtenidos en cada registro de la muestra, dando como resultado un tiempo promedio de 15.69 segundos, esto puede dar como resultado de que el prototipo sea capaz de registrar aproximadamente 4 personas por minuto.

Figura 46

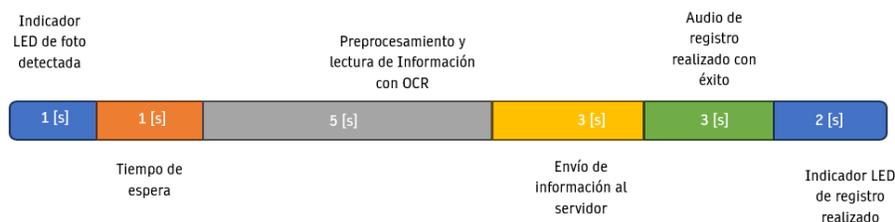
Gráfico de tiempos de respuesta del prototipo



Para esto se analiza los tiempos de cada proceso realizado por el software y se identifica que: existen tiempos utilizados por los indicadores tanto visuales como los de audio, tiempos de procesamiento para lectura de información y posterior envío al servidor como se ilustra en la Figura 47.

Figura 47

Tiempo de respuesta del prototipo



Análisis de resultados

Los resultados muestran que aproximadamente el 82% de las lecturas realizadas son correctas como se muestra en la Figura 43, estas no reflejan ninguna alteración ni error en la información y que se registraron correctamente en el servidor ubicado en la nube.

Existe un 18% que pertenece a las lecturas incorrectas, estas incluyen errores de 1 hasta n caracteres y los errores presentados en la clasificación del documento. Se considera información no válida y se clasifica en un solo grupo ya que si existe al menos un carácter erróneo se alteraría toda la información del documento del visitante y esta quedaría sin utilidad. El mayor número de errores se presentan en las cédulas con modelo antiguo según se observa en la Figura 44, estos errores se pueden ocasionar debido a baja o alta iluminación, marcas de agua, errores de impresión o ruido de imagen que alteran los bordes del carácter y el motor OCR tiende a identificar de forma errónea como se muestra en la Figura 47.

Al calcular las métricas de rendimiento del prototipo, se obtuvo un 100% de precisión en relación con la información detectada correctamente y un recall de 81.81%, lo que indica que el sistema pudo recuperar más del 80% de la información real de la muestra utilizada. Estos dos valores permitieron calcular el F1-score con valor de 89.99%, esta métrica se toma como la confiabilidad que el sistema presenta al leer información de las cédulas de identidad.

En cuanto al tiempo de respuesta, un parámetro de gran relevancia en este proyecto y que debe ser el menor posible. Luego de obtener todas las mediciones se calcula un tiempo promedio de 15.69 segundos que el prototipo se toma desde que se ingresa la cedula hasta que se registra la información en la base de datos ubicada en la nube, este tiempo puede considerarse en el límite del rango esperado por un sistema automático de registro de información. En la Figura 46 se observan picos máximos en algunas mediciones, esto se presenta debido a que la conexión entre el prototipo y la red no es estable y en algunos intervalos de tiempo se obtiene una intensidad de señal baja.

Errores de lectura debido a factores externos.

En las pruebas de funcionamiento se obtuvo casos donde no se pudo realizar el registro automático, el sistema marcó como error la identificación del documento que si pertenecía a una cédula de identidad ecuatoriana. Los errores se produjeron debido a factores externos que influyeron en la identificación de la lectura, en la figura 48 se muestra un error de impresión en la parte de un identificador, al estar sobrepuesto el texto el motor OCR no logra leer correctamente la información, también se presentan errores como los ilustrados en la Figura 49, las cédulas antiguas se fabrican con un holograma de seguridad en el fondo de los datos del ciudadano, estas al entrar en contacto con luz se emite un reflejo que en ocasiones afecta tanto en la identificación del documento como en la lectura de la información.

Figura 48

Error de impresión presentado en cédula ecuatoriana antigua

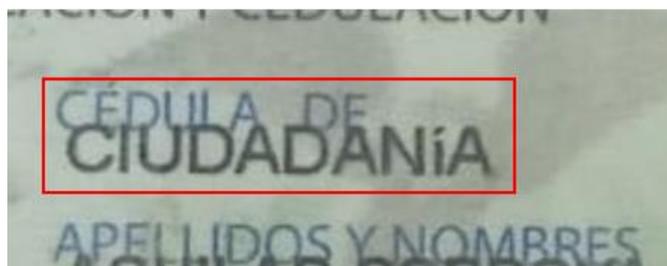


Figura 49

Holograma reflejado por iluminación en cedula ecuatoriana antigua



Capítulo VI: Conclusiones y Recomendaciones

El prototipo desarrollado en el presente proyecto de titulación tiene como finalidad automatizar el proceso de registro de información de visitantes en puntos de control de accesos, este proceso actualmente por lo general se realiza manualmente y con este sistema se pretende realizar de una manera rápida y sin errores.

La estructura elaborada permite tener portabilidad al prototipo, este puede ser trasladado e instalado en cualquier lugar que tenga conexión a internet, ya sea vía ethernet o wifi. Además, el prototipo al tener iluminación interior dentro de la caja permite que este pueda operar en cualquier ambiente independientemente de la luz exterior, lo que le hace idóneo para funcionar tanto en lugares con iluminación como en lugares con escasa iluminación.

El prototipo permite la lectura de información de cédulas de identidad ecuatorianas de dos tipos, tanto de cédulas con el modelo actual como el anterior modelo a este. Esta lectura se realiza bajo la condición de que la información contenida en el documento de identidad sea clara y legible, el estado del documento no debe presentar deformaciones que impidan que se introduzca en la ranura debido a que se construyó a medida.

Mediante las pruebas realizadas al prototipo de lectura se determinó que presenta una confiabilidad mayor al 85% y, aunque el motor OCR utilizado rinda una alta precisión en la lectura de caracteres y el porcentaje de recuperación (recall) obtenida del prototipo sea alta, el principal problema que se presenta es que en este tipo de aplicaciones no debe existir error en la lectura. Es decir, si un carácter presenta error entonces la información final resultado de la lectura perdería total validez al ser alterada y por consecuencia afectan drásticamente al rendimiento del sistema en general. Sin embargo, es casi imposible lograr una confiabilidad del 100% ya que no existe metodología que permita alcanzar este porcentaje.

La información final obtenida de la lectura del documento de identidad se puede observar en una página web donde se detallan en una tabla todos los registros guardados, esta información al estar alojada en la nube está disponible para ser mostrada en cualquier parte del

mundo a diferencia de los registros actuales que en su mayoría se los realiza manualmente en una hoja de papel. Esta información se puede utilizar para fines estadísticos, de control y supervisión, entre otros.

En cuanto al tiempo de respuesta, en las pruebas de funcionamiento se obtuvo un tiempo promedio de 15.69 segundos aproximadamente lo que significaría un total de 4 personas registradas por minuto, este tiempo puede considerarse aceptable o no dependiendo del lugar donde se requiera implementar. Si se requiere optimizar el tiempo de respuesta se recomienda eliminar los indicadores y reemplazar por un solo indicador audible, además el LED indicador se puede encender al inicio del proceso y apagarse cuando se produzca un error en el proceso y de paso al encendido del LED rojo o cuando se finalice con éxito el proceso. Reemplazando estos tiempos se puede obtener un tiempo de respuesta aproximado menor a los 11 segundos, esto permitiría que se realicen registros en mayor volumen por minuto y, en caso de implementarse en lugares con mayor afluencia el prototipo sea capaz de responder rápidamente y evitar aglomeraciones.

La iluminación que se utilice debe ser la adecuada, un exceso o escasez de esta puede afectar al rendimiento del sistema. Además, esta debe ser indirecta debido a que se pueden producir reflejos frente al lente de la cámara que ocasionen pérdidas de información al momento de aplicar el preprocesamiento de imágenes.

Para obtener imágenes claras, el enfoque de la cámara debe ser lo más nítido posible y la resolución debe sobrepasar los 5Mpx, esto debido a que la información requerida en el documento es de tamaño pequeño y que dificulta en el momento de realizar el reconocimiento de caracteres. La cámara debe colocarse en una distancia prudente y que no existan interferencia de objetos entre el lente y el documento.

Para la implementación del sistema se debe tener disponible una conexión WiFi estable y sin interrupciones, una falla en la conexión puede ocasionar que no se complete el proceso y la información no pueda ser enviada al servidor.

Trabajo Futuro

- *Validación de la información:* Implementar mecanismos de validación para asegurar la precisión de la información leída. Se podría comparar los datos obtenidos con bases de datos oficiales del registro civil ecuatoriano mediante API's.
- *Integración con otros sistemas de seguridad:* Integrar a sistemas de seguridad que utilicen otros tipos de validación como reconocimiento facial para validar la identidad de la persona a partir de la imagen capturada por la webcam y compararla con la foto de la cédula.
- *Definir otro propósito de uso:* Utilizar la información obtenida para verificar si el ciudadano tiene antecedentes penales, deudas al estado, etc.

Bibliografía

- Al Ghaithi, H., & Eaganathan, U. (2016). A Brief study and implementation of visitor management system for asia pacific university, Malasya. *International Journal of Advance Research in Science and Engineering*, 317-326.
- Bhavana, R., & Suvarna, N. (2021). Face Detection Using Haar Cascade Classifiers. *International Journal of Science*.
- Canny, J. (1986). A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 679-698.
- Chincheró, J. L. (s.f.). *Desarrollo de un dispositivo que mediante visión artificial permita adquirir imágenes con palabras para la conversión a audio, orientado a la ayuda de personas invidentes*. Universidad Tecnológica Israel, Quito.
- Comaniciu, D., & Meer, P. (2002). Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence vol. 24*, 603-619.
- Cuimei, L., Zhiliang, J., Nan, J., & Jianhua, W. (2017). Human face detection algorithm via Haar cascade classifier combined with three additional classifiers. *2017 IEEE 13th International Conference on Electronic Measurement & Instruments*, 46-49.
- Espinel, J. ... (s.f.). *Diseño e implementación de seguridad a.a.a(authentication, authorization and accounting) en las redes wi-fi del gad municipal del cantón Mejía*. *Seguridad a.a.a*. Universidad Técnica de Cotopaxi, Latacunga.
- He, Y. (2020). Research on Text Detection and Recognition Based on OCR Recognition Technology. *Proceedings of the IEEE 3rd International conference on information systems and computer aided education ICISCAE, 2020*, 132-140.
- Lowe, D. (2002). Object Recognition from Local Scale-Invariant Features. *Proceedings of the Seventh IEEE International Conference on Computer Vision*, 1150-1157.

- Martinez, E., & Araujo, A. (s.f.). Algoritmo para la Lectura por medio de Reconocimiento Óptico de Caracteres (OCR) de Etiquetas Nutricionales y la Generación de un Tipo de Sellos Frontales. *Proyecto de grado*. Universidad Antonio de Nariño, Neiva.
- Masmoudi, M., Ghazzai, H., & Frikha, M. (2019). Object Detection Learning Techniques for Autonomous Vehicle Applications. *2019 IEEE International Conference on Vehicular Electronics and Safety (ICVES)*, 1-5.
- Miao, W., Ting-Jie, L., Fei-Yang, L., Jing, S., & Hui-Ying, D. (2010). Research on the architecture of Internet of Things. *2010 3rd International Conference on Advanced Computer Theory and Engineering(ICACTE)*, (págs. V5-484-V5-487). Chengdu.
- O. Duda, R., E. Hart, P., & Stork, D. G. (2001). *Pattern Classification, 2nd Edition*. New York: Jhon Wiley & Sons Inc.
- ohms, 3. (17 de Mayo de 2020). *Que puedo hacer con una Jetson Nano?* Obtenido de <https://blog.330ohms.com/2020/05/17/que-puedo-hacer-con-una-jetson-nano/>.
- Oktaviadri, M., & Kah Keat, F. (2019). Design and Development of Visitor Management System. *Journal of Intelligent Manufacturing & Mechatronics*, 73-79.
- Purwa, D., Gusti, W., & Soeleman, A. (2018). Face Detection Using Haar Cascade in Difference Illumination. *2018 International Seminar on Application for Technology of Information and Communication* , 555-559.
- Rakesh Kamisetty, V. N., Chidvilas, S., Revathy, S., Jeyanthi, P., Anu, V. M., & Gladence, L. M. (2022). Digitization of Data from Invoice using OCR. *2022 6th International Conference on Computing Methodologies and Communication (ICCMC)*, 1-10.
- Rose, K., Eldridge, S., & Chapin, L. (15 de 11 de 2022). *Internet Society*. Obtenido de <https://www.internetsociety.org/wp-content/uploads/2017/09/report-InternetOfThings-20160817-es-1.pdf>

- Singh, H., & Sachan, A. (2018). A proposed approach for character recognition using Document Analysis with OCR. *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*, 190-195.
- Smith, R. (2007). An Overview of the Tesseract OCR Engine. *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, 629-633.
- Song, Y., Li, C., & Zhou, Q. (2023). A parallel Canny edge detection algorithm based on OpenCL acceleration. *Research Square*.
- Szeliski, R. (2021). *Computer Vision: Algorithms and Applications*. Washington: Springer.
- Universo, E. (16 de Agosto de 2022). *El Universo Noticias*. Obtenido de <https://www.eluniverso.com/noticias/seguridad/asi-estan-las-cifras-de-la-inseguridad-en-el-primer-semester-del-2022-comparadas-con-los-tres-anos-previos-nota/>.
- Viola, P., & Jones, M. (2001). Rapid Object Detection using a Boosted Cascade of Simple Features. *Accepted Conference on Computer Vision and Pattern Recognition 2001*, 25-29.
- Xin, L., & YiLiang, S. (2018). Computer Vision Imaging Based on Artificial Intelligence. *2018 International Conference on Virtual Reality and Intelligent Systems*, 22-25.
- Yu, W., Liang, F., He, X., Grant Hatcher, W., Lu, C., Lin, J., & Yang, X. (2018). A Survey on the Edge Computing for the Internet of Things. *IEEE*, 6900-6919.

Apéndices

- Enlace repositorio GitHub de código implementado:

<https://github.com/MauricioLojaP/Lectura-ID-EC-OCR.git>