

UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería



**Optimización de la herramienta de procesamiento de imágenes
para el sistema Brainlab de HUMANA - Fase III**

Protocolo de trabajo de graduación presentado por Francisco Estephan
Portales Okrassa, estudiante de Ingeniería Mecatrónica

Guatemala,

2023

Resumen

La visión por computadora es una herramienta que permite a las computadoras y a los sistemas informáticos obtener información útil contenida en imágenes digitales, videos y algunas otras entradas digitales. Esta herramienta ha sido utilizada en diversos proyectos siendo estos enfocados a la obtención de caracteres correspondientes a ángulos de configuración para el posicionamiento de un sistema robótico.

Con el proyecto que se propone se busca la optimización de los algoritmos ya desarrollados junto con una interfaz con la que se le pueda facilitar al usuario el uso de las herramientas de procesamiento he identificación de los ángulos de configuración. También se busca evaluar la migración de las herramientas que se han desarrollado a un sistema embebido. Además, se busca verificar el desempeño de los algoritmos usando capturas de pantalla en lugar de fotografías tomadas. El objetivo es optimizar recursos, mejorar la portabilidad e instalación del sistema y minimizar el error en la obtención de los datos. Finalmente se busca implementar un protocolo para el envío de datos hacia un brazo robótico de asistencia para cirugías que se está desarrollando para HUMANA.

Antecedentes

El reconocimiento de imágenes ha sido una rama que ha llevado muchos años de estudio y perfeccionamiento. Se ha buscado implementar numerosos algoritmos tanto para el preprocesar de las imágenes como para el reconocimiento de características específicas contenidas en estas para diversas aplicaciones como lo son los sistemas de control.

En la Universidad del Valle de Guatemala se han realizado diversos estudios relacionados con la visión por computadora y *machine learning*. A lo largo de los estudios realizados se ha buscado poder optimizar y perfeccionar todo el proceso que conlleva realizar el procesamiento de imágenes. Un ejemplo de estos estudios que se han realizado es el trabajo realizado por S. Galicia [1] el cual complementa de gran manera el detectar características de calibración para sistemas robóticos en HUMANA [2], el sistema robótico con el que se ha trabajado es el Neuromate el cuál se puede observar en la Figura 1.



Figura 1: Robot NeuroMate. [2]

HUMANA

El centro de epilepsia HUMANA [2] se especializa en el diagnóstico y tratamiento de la epilepsia el cuál se enfoca en utilizar tecnología avanzada para de esta manera poder mejorar la atención de los pacientes. Una de estas tecnología que es clave en estos procesos es la inteligencia artificial la cuál les permite analizar grandes cantidades de datos para la identificación de patrones y posteriormente predecir los resultados. La visión por computadora también es utilizada en este centro para poder analizar las imágenes cerebrales y brindar información detallada sobre la ubicación y la extensión de las lesiones cerebrales.

Brainlab

Brainlab [3] es una de las empresas líderes en tecnología en el ámbito médico quienes combinan la tecnología de los sistemas robóticos actuales con inteligencia artificial para poder mejorar los procesos de diagnóstico y tratamiento de los pacientes. Estos ofrecen robots quirúrgicos junto con soluciones de inteligencia artificial para poder proporcionar una mayor precisión y control de las intervenciones quirúrgicas. Realizan procesamiento de imágenes en tiempo real con herramientas como ExacTrac.

Investigaciones externas a la Universidad del Valle de Guatemala

En el trabajo de graduación de Andrea Builes y Esteban Palacio [4] se presenta una simulación de la integración entre un sistema de visión artificial y un manipulador robótico serial con el objetivo que este logre posicionar cuerpos con formas geométricas simples los cuales se encuentran sobre una banda transportadora en movimiento. Se utilizó el software de MATLAB para extraer el centroide de los objetos mediante el procesamiento de imágenes mediante el *toolbox* de *image processing*. Se presenta una forma de poder calibrar la cámara que se desea utilizar mediante la herramienta llamada *Computer Vision System*. Para este trabajo se utilizó la cámara Logitech HD C615 la cuál cumplió con los requerimientos para la aplicación deseada. A pesar de presentar un proceso bastante completo para el correcto procesamiento de imágenes no se realizaron pruebas con diferentes dispositivos y software para poder brindar una comparación que pueda ser útil para posteriores investigaciones.

En la tesis de Enrique Palafox [5] se presenta principalmente algoritmos y técnicas para un procesamiento de imágenes. Se muestran métodos espaciales en los que se incluyen diferentes tipos de filtros tanto para mejorar la nitidez de las imágenes como para la detección de bordes. Se presentan métodos frecuenciales con los cuales se es capaz de realizar compresiones y expansiones de las imágenes con el objetivo de tener archivos que sean más livianos y sean mucho más sencillos de procesar, esto mediante herramientas matemáticas. Se buscó poder implementar las diferentes herramientas mencionadas en un DSK 6211, se presenta el proceso con el cuál se puede realizar la primera comunicación con este y cómo transferirle todos los datos que son necesarios para la correcta implementación.

En la tesis de Jonathan González [6] se presentan los resultados que se obtuvieron de un estudio sobre el uso de la librería OpenCV en Python para el reconocimiento de imágenes. El procesamiento de imágenes se realizó incluyendo la carga de imágenes, el filtrado y la

segmentación de objetos, como resultado de este procesamiento se implementó una detección de objetos utilizando algoritmos de aprendizaje automático como SVM y redes neuronales. Como resultado de la investigación se determinó que OpenCV y Python son muy útiles para el procesamiento de imágenes y la detección de objetos en estas. Se encontró que la precisión que se puede obtener en la detección depende mucho del conjunto de datos utilizado y de la complejidad de los objetos que se intentan detectar.

Investigaciones en Universidad del Valle de Guatemala

La Universidad del Valle de Guatemala a lo largo de los años ha mantenido la relación de investigación con el Centro de Epilepsia y Neurocirugía Funcional HUMANA [2] en la cual se han desarrollado numerosos proyectos que aportan de gran manera a la optimización de todos los procesos que se llevan a cabo en este centro, desde el control mecánico de los instrumentos hasta procesamiento de resultados de los diferentes estudios que ahí se realizan.

En el trabajo de graduación de Juan González [7] se presentan los resultados de un sistema de detección de ángulos de configuración del sistema VarioGuide de Brainlab [3]. Los datos que se detectan tienen como propósito final que un sistema robótico pueda replicarlos y así maximizar la precisión que se pueda tener en las cirugías. Este es una fase del proyecto del brazo de HUMANA [2] las cuales se dividen en software y hardware. La implementación realizada por González cuenta con 7 fases de procesamiento de las imágenes, tuvo como resultados reconocer satisfactoriamente los ángulos de configuración, esto dentro de los márgenes de error establecidos por Brainlab. Se hicieron pruebas con diferentes sensores ópticos, el primero fue el que posee la cámara web Logitech 1080p, el segundo fue con la cámara Gopro Hero 8 con la cual se obtuvieron los mejores resultados debido a su mejor resolución. En la Figura 2 se observa la fotografía tomada por la Gopro Hero 8, en la Figura 3 se observa el resultado de la aplicación de los filtros para finalmente recortar y someter a binarización el ángulo de configuración deseado que se observa en la Figura 4.

En el trabajo de graduación de Santiago Galicia [1] se presenta una investigación bastante amplia en el procesamiento de imágenes mediante distintos algoritmos los cuales algunos se encargan en preparar dichas imágenes para su posterior análisis y procesamiento, esto aplicando diferentes filtros para mejorar la implementación y resultados con cada uno de estos llegando a la conclusión del software más óptimo para el procesamiento de imágenes. Se investigó sobre el mejor hardware para la implementación tanto como dispositivos que capturen las imágenes como las bases utilizadas para colocar los dispositivos para realizar las capturas. Los resultados de la investigación indican que, en cuanto al dispositivo de captura, lo más importante entre muchos otros es el ángulo de visión ya que entre más amplio, el error de procesamiento será mayor. Se recomienda realizar pruebas con mejores cámaras y buscar alternativas de procesamiento para poder implementar en HUMANA [2].

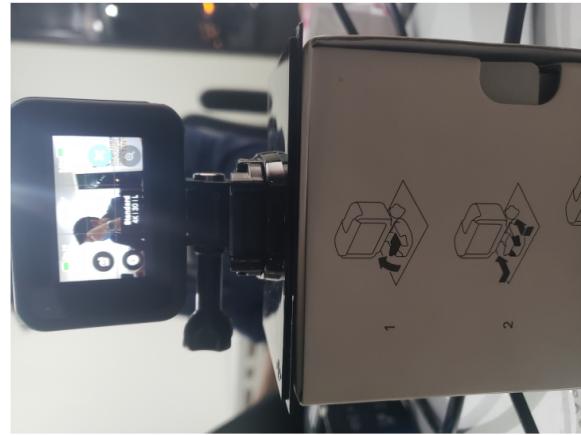


Figura 2: Fotografía por Gopro Hero 8. [7]

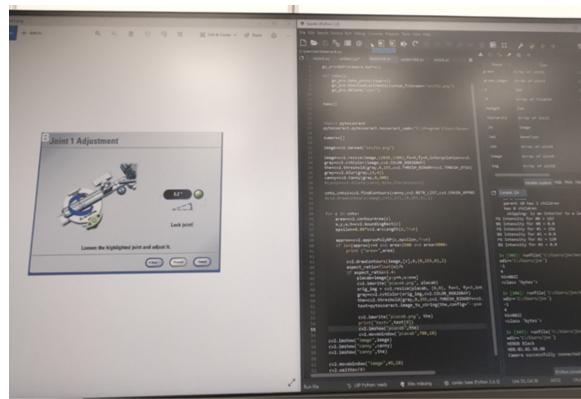


Figura 3: Resultado de aplicación de filtros. [7]



Figura 4: Recorte y binarización a imagen. [7]

Justificación

Este trabajo de graduación tiene como objetivo mejorar y optimizar los algoritmos previamente utilizados en trabajos de graduación anteriores, como los realizados por Gonzalez y Galicia. Además, busca investigar la viabilidad de migrar estas herramientas a un sistema

embebido para lograr mayor eficiencia y no depender del equipo de cómputo disponible en el área de implementación.

En trabajos anteriores, se ha enfrentado el problema de contar con equipos deficientes o inadecuados en HUMANA o en el área de implementación, lo que puede afectar el rendimiento y la ejecución correcta de los algoritmos. Se utilizó una cámara web para las capturas, lo cual generó desafíos como reflejos de luz en la pantalla, mala iluminación en el área de trabajo y problemas con la posición de la cámara. Para solucionar esto, se propone emplear un sistema de capturadora de video, con el objetivo de obtener imágenes más claras y reducir significativamente el margen de error al aplicar los algoritmos de procesamiento y obtener los datos de los ángulos deseados.

Objetivos

Objetivo General

Optimizar la herramienta de procesamiento de imágenes y reconocimiento de caracteres para el sistema Brainlab de HUMANA, para poder generar correctamente los comandos de control para un brazo robótico de asistencia en cirugías.

Objetivos Específicos

- Validar la herramienta de procesamiento de imágenes y reconocimiento de caracteres desarrollada en la fase anterior con más imágenes, obtenidas con cámaras, del sistema Brainlab de HUMANA.
- Evaluar el rendimiento de la herramienta con imágenes obtenidas como capturas de pantalla.
- Ajustar los algoritmos para mejorar el reconocimiento de caracteres logrado en la fase anterior.
- Evaluar la posibilidad de migrar la herramienta a un sistema embebido y su integración al sistema de Brainlab.
- Implementar un protocolo para el envío de comandos al sistema de control del brazo robótico que se está desarrollando para HUMANA.

Marco teórico

Visión por computadora

La visión artificial mediante computadoras permite que los dispositivos computacionales puedan percibir y distinguir información relevante presente en imágenes digitales, videos y otras entradas visuales, lo cual les permite tomar acciones en base a esa información. Un ejemplo de esto se puede observar en la Figura 5

Para lograr que la visión artificial funcione, es necesario inicialmente contar con una gran cantidad de imágenes para entrenar y capacitar el modelo para reconocimiento de elementos importantes presentes en dichas imágenes.

Dos tecnologías comúnmente utilizadas son el aprendizaje profundo (*deep learning*) y las redes neuronales convolucionales. El aprendizaje profundo utiliza modelos algorítmicos que permiten que una computadora aprenda sobre el contexto de los datos visuales. Si se dispone de suficientes datos, la computadora puede observar y aprender a diferenciar entre distintas imágenes o características y objetos.

Por otro lado, las redes neuronales convolucionales realizan convoluciones y evalúan la precisión de sus predicciones en múltiples iteraciones hasta que estas se acerquen a la realidad. Luego de este proceso, la computadora adquiere la capacidad de reconocer e interpretar imágenes de manera similar a los seres humanos. [8]

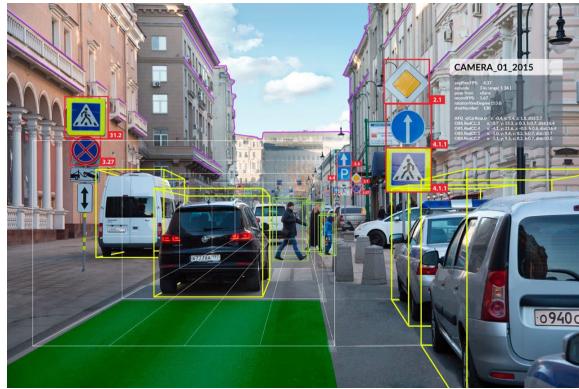


Figura 5: Ejemplo de aplicación de Visión por computadora. 5

Reconocimiento Óptico de Caracteres (OCR)

El OCR (Reconocimiento Óptico de Caracteres) es un software que identifica el texto presente en una imagen y lo convierte en cadenas de caracteres para ser guardado en un formato utilizable en programas de edición de texto. Son sistemas capaces de leer datos escritos y reconocerlos a altas velocidades, de manera individual por carácter. Sin embargo, la mayoría de los sistemas OCR aún no pueden reconocer caracteres o palabras escritas a mano ni documentos que presenten deterioro físico. En la actualidad, los sistemas OCR realizan varios procesos para procesar una imagen, como organizarla para obtener los caracteres, extraer los caracteres y clasificar los caracteres extraídos.

Cuando se tiene una imagen, ya sea una fotografía o un documento escaneado, el texto que contiene es parte integral de la imagen, al igual que cualquier otro elemento visual, como dibujos o esquemas. Si se desea extraer ese texto para editarlo, se necesita un programa de OCR que pueda reconocer y convertir el texto en una cadena de caracteres, ya sea en formato ASCII o Unicode. Posteriormente, se puede copiar esta cadena en un programa de edición de texto para trabajar con ella, lo que nos ahorra tiempo al no tener que teclear manualmente dicho texto. [9]

Motores para reconocimiento de caracteres

Algunos de los motores más populares utilizados en el mercado son Tesseract, el cuál es un motor de código abierto que fue desarrollado por google. Soporta más de 100 idiomas, y es conocido por su alta precisión y capacidad de poder trabajar con diferentes tipos de fuentes y diseños de texto. [10]

ABBY FineReader también es conocida por su alta precisión y capacidad de reconocer textos de varios idiomas, sin embargo, este también ofrece algunas funciones avanzadas como detección de tablas, preservación del formato y conversión de formatos que se pueden editar. [11]

Open CV

OpenCV es una librería de código abierto enfocada a la visión por computadora y *machine learning*. Esta librería contiene más de 2500 algoritmos optimizados, estos se pueden utilizar para detectar rostros, objetos, clasificar acciones humanas en videos, etc. Se tiene soporte para distintas plataformas y lenguajes de programación como C++, Python y MATLAB, aunque nativamente está escrita en C++. [12]

Python

Python es un lenguaje de programación versátil, multiplataforma y multiparadigma el cuál se ha destacado por su código legible y limpio. Este es un lenguaje de programación con licencia de código abierto lo cuál permite que se pueda utilizar en distintos contextos de forma gratuita.

El objetivo principal de Python es la automatización de procesos lo que hace de las tareas algo mucho más simple. Uno de los usos de Python es la inteligencia artificial, ya que al ser un lenguaje simple, escalable y robusto permite implementar las ideas complejas en unas cuantas líneas de código. [13]

Sistemas embebidos

Los sistemas embebidos son sistemas de computación diseñados para cumplir funciones específicas, y suelen tener sus componentes integrados en una placa base. En el campo del *machine learning* y la visión por computadora, es relevante destacar que estos sistemas embebidos pueden ser utilizados para llevar a cabo tareas específicas en estos campos. Por ejemplo, se pueden emplear microcontroladores con capacidades limitadas pero dedicados a tareas de reconocimiento de imágenes o análisis de datos visuales. En las figuras 6 y 7 se puede observar ejemplos de sistemas embebidos. [14]

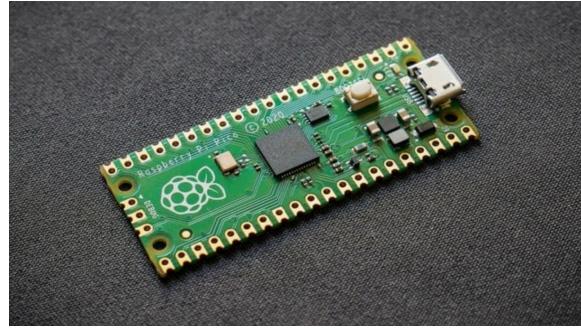


Figura 6: Raspberry Pi PICO. [15]



Figura 7: Arduino UNO. [16]

Raspberry Pi

La Raspberry Pi, un económico y compacto ordenador del tamaño de una tarjeta de crédito, se destaca por su capacidad para ser conectado a un monitor, teclado y ratón. A pesar de su tamaño reducido, esta potente plataforma es capaz de llevar a cabo una amplia gama de tareas similares a las de una computadora convencional. Sin embargo, lo que la distingue aún más es su capacidad de interactuar con su entorno físico a través de periféricos adicionales.

En el contexto de *machine learning* y visión por computadora, la Raspberry Pi se ha convertido en una opción popular debido a su versatilidad. Utilizando el lenguaje de programación Python y aprovechando el sistema operativo Linux, este dispositivo se convierte en un recurso invaluable para implementar algoritmos de aprendizaje automático y llevar a cabo análisis visual. Sus capacidades de procesamiento de datos y su capacidad para interactuar con sensores y cámaras le permiten realizar tareas complejas como el reconocimiento de imágenes, la detección de objetos y otras aplicaciones relacionadas con la visión por

computadora. [15]



Figura 8: Raspberry Pi 4. [15]

Tarjetas capturadoras de video

Las tarjetas capturadoras de video desempeñan un papel fundamental en la visión por computadora al permitir la adquisición de señales de video provenientes de fuentes externas. Estas tarjetas están diseñadas para capturar y procesar la información visual transmitida a través de conexiones como HDMI.

Una de las aplicaciones más comunes de las tarjetas capturadoras de video es la captura de juegos en consolas. Estas tarjetas reciben de manera continua los datos generados por la GPU de la computadora o consola de videojuegos. La información visual, que incluye imágenes en tiempo real y secuencias de video, es capturada en tiempo real por la tarjeta capturadora.

Una vez que los datos de video son capturados, se pueden realizar varias acciones según la configuración establecida por el usuario. La captura de video puede ser almacenada en un archivo para su posterior reproducción, grabación o análisis. Además, la tarjeta capturadora puede aplicar codificación a los datos capturados, lo que permite comprimir el contenido de video para su transmisión o almacenamiento eficiente.

Es importante destacar que las tarjetas capturadoras de video son altamente versátiles y pueden ser utilizadas en diversas aplicaciones de visión por computadora más allá de la captura de juegos. Estas tarjetas son utilizadas en la adquisición de video para el procesamiento y análisis de imágenes, detección y seguimiento de objetos, reconocimiento de patrones, entre otras aplicaciones relacionadas con la visión por computadora.[17]

Metodología

Exploración a librería OpenCV

OpenCV es una librería muy extensa que ofrece herramientas para el procesamiento de imágenes y el reconocimiento de caracteres por lo que el primer paso es indagar, investigar y poner a prueba las funciones que sean de utilidad para la identificación de los ángulos de configuración. Una vez haya sido evaluado el rendimiento de cada una de las funciones, el siguiente paso es comparar con las utilizadas en algoritmos previos y de ser necesario realizar modificaciones que se consideren adecuadas.

Optimización de algoritmos de procesamiento

Algunas de las funciones presentadas en algoritmos previos no presentan una organización en la programación. Se pretende organizar la programación en diferentes documentos para así sea mucho más eficiente y fácil la utilización de esta para personas que quieran trabajar en el código en futuros proyectos. Se trabajará en la generalización del código ya que en algoritmos anteriores se tiene el problema que se tiene que trabajar en un ambiente muy controlado respecto a la ubicación de los archivos en el sistema y la idea es que sin importar donde se ubiquen los archivos, se pueda hacer uso de los algoritmos de una forma sencilla.

Comparación con imágenes a partir de cámara y con capturas de pantalla

En las pruebas realizadas en trabajos anteriores se realizaron las capturas de las imágenes con una cámara por lo que se presentaron problemas como el reflejo de la luz en la pantalla, la posición de la cámara en la pantalla lo que provocó que se tuvieran que realizar algoritmos extras para la eliminación de estos detalles. Se pretende trabajar con capturas de pantalla por lo que primero se evaluará si es factible trabajar con las herramientas que ya se ofrecen en los equipos de cómputo y con los sistemas embebidos para así también minimizar el equipo necesario para la correcta implementación, de no ser factible dicha implementación, se procederá a utilizar una capturadora de video para la obtención de las imágenes. Finalmente, cuando se decida qué herramienta ofrece mejores resultados, se procederá realizar modificaciones al algoritmo para acoplarse a la forma final de trabajo.

Creación de GUI optimizada con Python y OpenCV

Se trabajará en la mejora y optimización del GUI ya desarrollado en trabajos anteriores, adicionalmente se implementarán nuevas funciones, herramientas y utilidades a estas como la generalización del código como el mencionado en secciones anteriores, se pretende que el usuario final no tenga que involucrarse mucho en lo que respecta al algoritmo de programa.

Migración a sistemas embebidos

Una vez implementado correctamente todo lo antes mencionado se evaluará la migración a sistemas embebidos para así poder mejorar la portabilidad y eficiencia del sistema. Se pretende hacer uso de una Raspberry PI 4 ya que ofrece una interfaz como la de una computadora, sin embargo, se implementarán algoritmos con los que una interfaz en el sistema embebido no sea necesario.

Creación de algoritmo para protocolo de comunicación con brazo robótico

Se evaluará con distintos protocolos de comunicación para la transmisión de datos hacia el sistema robótico del brazo por lo que se trabajará en la estructura en la que los datos de transmitirán junto con programación defensiva para minimizar los errores que se puedan tener al momento de la transmisión.

Pruebas finales con todo integrado

Finalmente se realizarán pruebas finales en los laboratorios de HUMANA con los sistemas y algoritmos que se hayan decidido finalmente utilizar. Se harán pruebas con diferentes escenarios para poder garantizar la efectividad en la obtención de los ángulos de configuración, junto a esto se harán pruebas con el sistema robótico del brazo para comprobar el correcto funcionamiento de todo lo que involucra este proyecto.

Cronograma de actividades

No.	Nombre de la Tarea	Fecha de inicio	Fecha de finalización	Estado
1	Instalar Biblioteca OpenCV	28/05/2023	03/06/2023	Abierto
1.1	Indagar en algoritmos desarrollados en trabajos anteriores	28/05/2023	03/06/2023	Abierto
1.2	Realizar el procesamiento a imágenes de prueba	28/05/2023	04/06/2023	Abierto
1.3	Obtener el ángulo de configuración de imágenes tomadas de HUMANA	28/05/2023	05/06/2023	Abierto
2	Mejorar algoritmos	05/06/2023	24/06/2023	Abierto
2.1	Indagar en librería OpenCV	05/06/2023	06/06/2023	Abierto
2.2	Reorganizar funciones	07/06/2023	10/06/2023	Abierto
2.3	Generalizar algoritmos y hacer programación defensiva	07/06/2023	10/06/2023	Abierto
2.4	Hacer pruebas y corregir posibles errores de implementación	12/06/2023	15/06/2023	Abierto
2.5	Explorar en la GUI propuesta en trabajos anteriores	15/06/2023	21/06/2023	Abierto
2.6	Comenzar a mejorar GUI con la programación organizada y	16/06/2023	21/06/2023	Abierto
2.7	Pruebas de procesamiento con GUI	22/06/2023	24/06/2023	Abierto
3	Migración a sistema embeddo	26/06/2023	29/07/2023	Abierto
3.1	Instalación de librerías en Raspberry PI y trabajar en redacción de tesis	26/06/2023	27/07/2023	Abierto
3.2	Pruebas de procesamiento con GUI, de ser necesario, adicionar algoritmos	28/06/2023	01/07/2023	Abierto
3.3	Investigar forma alternativa a tomar captura en Windows y transferirla a la raspberry	03/07/2023	06/07/2023	Abierto
3.4	De encontrar cómo hacerlo, comenzar a implementar algoritmos alternativos	10/07/2023	15/07/2023	Abierto
3.5	Investigar cómo acopiar una capturadora de video a algoritmo existente	17/07/2023	22/07/2023	Abierto
3.6	Trabajo en los errores del algoritmo y seguir trabajando en la implementación en la Raspberry Y protocolo de comunicación con sistema robótico	24/07/2023	29/07/2023	Abierto
4	Hacer pruebas y corregir posibles errores de implementación	01/08/2023	12/08/2023	Abierto
4.1	Visitar laboratorio de HUMANA para obtener imágenes	01/08/2023	12/08/2023	Abierto
4.2	Obtener datos de implementación deseada por HUMANA para mejorar GUI y algoritmos	01/08/2023	12/08/2023	Abierto
5	Trabajo Escrito	14/08/2023	31/08/2023	Abierto
5.1	Trabajar en detalles finales de la entrega final de la redacción de la tesis	14/08/2023	31/08/2023	Abierto

Figura 9: Cronograma de actividades

Índice preliminar

1. Prefacio	1
2. Resumen	2
3. Abstract	2
4. Introducción	3
5. Antecedentes	4
a) Investigaciones externas a la Universidad del Valle de Guatemala	4
b) Investigaciones en Universidad del Valle de Guatemala	4
6. Justificación	5
7. Objetivos	6
a) Objetivos generales	6
b) Objetivos específicos	6
8. Alcance	7
9. Marco Teórico	8
a) Vision por computadora	8
b) Reconocimiento óptico de caracteres OCR	8
c) Motores para reconocimiento de caracteres	8
d) Raspberry PI 4	8
e) Python	8
f) OpenCV	8
g) Sistemas embebidos	8
h) Dispositivos capturadores de video	8
10. Resultados	9
a) Optimización de algoritmos	9
b) Mejora en reconocimiento de caracteres	9
c) Sistema captura de pantalla	9
d) Migración a sistemas embebidos	9
e) Resultados con capturadora de video	9
f) Comunicación con sistema robótico	9
11. Conclusiones	9
12. Recomendaciones	10
13. Bibliografía	11
14. Anexos	12
15. Glosario	13

Referencias

- [1] S. Galicia, “Optimización de un sistema de procesamiento de imágenes para el reconocimiento de variables en la pantalla del sistema de Brainlab de HUMANA,” Trabajo de graduación de grado de licenciatura, Universidad del Valle de Guatemala, 2022.
- [2] HUMANA. “HUMANA.” (2023), dirección: humanagt.org (visitado 2023).
- [3] Brainlab. “Brainlab transforma el sector de la salud mediante tecnología basada en software.” (2023), dirección: brainlab.com (visitado 2023).
- [4] A. Builes y E. Palacio, “Aplicación del reconocimiento de imágenes para la manipulación robótica de objetos en movimiento,” Trabajo de grado para licenciatura en ingeniería mecánica, Universidad Pontificia Bolivariana, 2015.
- [5] L. E. Palafox, “Técnicas de procesamiento de imágenes utilizando un procesador digital de señales,” Tesis de postgrado, Instituto Politécnico Nacional, 2002.
- [6] F. J. González, “Reconocimiento de objetos utilizando OpenCV y Python en una Raspberry pi 2 en una tlapalería,” Trabajo de graduación de grado de licenciatura, Universidad Autónoma del Estado de Mexico, 2017.
- [7] J. G. González, “Diseño de un sistema de procesamiento de imágenes para la identificación de variables en una pantalla de sistema Brainlab del Centro de Epilepsia y Neurocirugía Funcional HUMANA,” Trabajo de graduación de grado de licenciatura, Universidad del Valle de Guatemala, 2022.
- [8] IBM. “¿Que es la visión artificial?” (2023), dirección: <https://www.ibm.com/mx-es/topics/computer-vision> (visitado 2023).
- [9] Microsoft. “OCR: reconocimiento óptico de caracteres.” (2023), dirección: <https://learn.microsoft.com/es-es/azure/cognitive-services/computer-vision/overview-ocr> (visitado 2023).
- [10] BytePace. “What is Tesseract and how it works?” (2019), dirección: <https://bytepace.medium.com/what-is-tesseract-and-how-it-works-dfff720f4a32> (visitado 2023).
- [11] I. Library. “Introduction to OCR and Searchable PDFs.” (2023), dirección: <https://bytepace.medium.com/what-is-tesseract-and-how-it-works-dfff720f4a32> (visitado 2023).
- [12] OpenCV. “OpenCV.” (2023), dirección: <https://opencv.org/about/> (visitado 2023).
- [13] Capacitarte. “¿Qué es y para qué sirve Python?” (2023), dirección: <https://www.capacitarte.org/blog/nota/que-es-y-para-que-sirve-python> (visitado 2023).
- [14] S. Luchetti. “Sistemas embebidos y sus características.” (2021), dirección: <https://tech.tribalyte.eu/blog-sistema-embebido-caracteristicas> (visitado 2023).
- [15] R. P. Foundation. “What is a Raspberry Pi?” (2023), dirección: <https://www.raspberrypi.org/> (visitado 2023).
- [16] Arduino. “Arduino Hardware.” (2023), dirección: <https://www.arduino.cc/en/hardware> (visitado 2023).
- [17] Filmora. “Introducción Completa a las Tarjetas Capturadoras.” (2023), dirección: <https://filmora.wondershare.es/screen-recorder/introduction-to-capture-card.html> (visitado 2023).