



UNIVERSIDAD POLITÉCNICA SALESIANA

SEDE GUAYAQUIL

CARRERA DE ELECTRÓNICA Y AUTOMATIZACIÓN

**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE SEGMENTACIÓN DE
COMPONENTES ELECTRÓNICOS EN TARJETAS ELECTRÓNICAS MEDIANTE VISIÓN
ARTIFICIAL**

Trabajo de titulación previo a la obtención del

Título de Ingeniero en Electrónica

AUTORES: MIGUEL GUSTAVO NAVARRETE SALAZAR

JONATHAN FERNANDO YANSE ESPINOZA

TUTOR: LIVINGTON ALFREDO MIRANDA DELGADO, MSc.

Guayaquil – Ecuador

2024

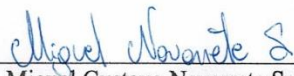
**CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE
TITULACIÓN**

Nosotros, Miguel Gustavo Navarrete Salazar con documento de identificación N° 0924216526 y Jonathan Fernando Yanse Espinoza con documento de identificación N° 0924348188; manifestamos que:

Somos los autores y responsables del presente trabajo; y, que autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

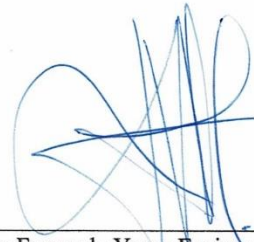
Guayaquil, 30 de enero del año 2024.

Atentamente,



Miguel Gustavo Navarrete Salazar

0924216526



Jonathan Fernando Yanse Espinoza

0924348188

**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE
TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA**

Nosotros, Miguel Gustavo Navarrete Salazar con documento de identificación N° 0924216526 y Jonathan Fernando Yanse Espinoza con documento de identificación N° 0924348188, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del Proyecto Técnico: “Diseño e implementación de un sistema de segmentación de componentes electrónicos en tarjetas electrónicas mediante visión artificial”, el cual ha sido desarrollado para optar por el título de: Ingeniero Electrónico, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

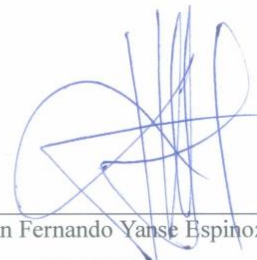
Guayaquil, 30 de enero del año 2024.

Atentamente,



Miguel Gustavo Navarrete Salazar

0924216526



Jonathan Fernando Yanse Espinoza

0924348188

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Livingston Alfredo Miranda Delgado, Msc. con documento de identificación N° 0930635172 docente de la Universidad Politécnica Salesiana Sede Guayaquil, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE SEGMENTACIÓN DE COMPONENTES ELECTRÓNICAS EN TARJETAS ELECTRÓNICAS MEDIANTE VISIÓN ARTIFICIAL. Realizado por Miguel Gustavo Navarrete Salazar con documento de identificación N° 0924216526 y por Jonathan Fernando Yanse Espinoza con documento de identificación N° 0924348188, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto Técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Guayaquil, 30 de enero del año 2024.

Atentamente,



Livingston Alfredo Miranda Delgado, MSc.

0930635172

DEDICATORIA

Dedicado a los pilares de mi vida, Karen y Gustavo quienes han sido un apoyo incondicional durante mi etapa universitaria, por confiar en mí y ayudarme a cumplir cada uno de los objetivos que me propuse. A mis hermanas Paulina y Rosanna, que mejoran mis días con sus bromas y ocurrencias.

Miguel Navarrete S.

A mis seres queridos, fuente inagotable de inspiración y apoyo. Este proyecto está dedicado a ustedes, cuyo aliento ha sido la chispa que encendió este camino. Con gratitud y amor.

Jonathan Yanse E.

AGRADECIMIENTO

A Dios por permitirme tener y disfrutar a mi familia.

A mis tías Sandra y Rosalía que me han auxiliado en mis peores momentos, por ofrecerme su sabiduría y cariño incondicional.

A mis familiares cercanos que a través de palabras de aliento y bromas me han brindado apoyo durante mi etapa universitaria.

A mis compañeros Jocelyne, Kevyn, Jonathan, Jhoell y Roy con quienes compartí risas, juegos y consejos, gracias por acompañarme en mi vida universitaria ayudándome a crecer tanto a nivel personal como profesional.

Miguel Navarrete S.

Agradezco sinceramente a mi familia que contribuyeron en este camino académico, con su apoyo y dedicación. Este proyecto es para todos los que creen en la importancia de nuestro trabajo y esfuerzo. ¡Gracias por ser parte de este sendero de aprendizaje y conocimiento!

Jonathan Yanse E.

RESUMEN

Este contemporáneo proyecto tiene como objetivo demostrar como el uso de la visión artificial puede ayudar a los estudiantes y técnicos en el proceso de identificación y segmentación de componentes electrónicos que se encuentran en las tarjetas electrónicas. La creciente complejidad de los diseños de tarjetas electrónicas y la demanda de una producción más eficiente en la industria electrónica han generado el incremento de recursos más efectivos para la inspección y clasificación de componentes. La investigación comienza con el análisis del método de procesamiento de imágenes y estrategias de aprendizaje automático para identificar el procedimiento más adecuado para elaborar la implementación.

Para el desarrollo del sistema propuesto se utiliza el Raspberry Pi como plataforma de hardware y una cámara para la adquisición de imágenes de tarjetas electrónicas. La validación del sistema se lo realiza mediante pruebas utilizando conjuntos de datos reales de tarjetas electrónicas y para finalizar se evalúa la precisión y eficiencia del algoritmo propuesto en comparación de métodos tradicionales como lo es la inspección visual.

Palabras clave: Visión artificial, segmentación, tarjetas electrónicas, Raspberry Pi.

ABSTRACT

This contemporary project aims to demonstrate how the use of artificial vision can help students and technicians in the process of identification and segmentation of electronic components found in electronic cards. The increasing complexity of electronic card designs and the demand for more efficient production in the electronics industry have generated the increase of more effective resources for the inspection and classification of components. The research begins with the analysis of the image processing method and machine learning strategies to identify the most appropriate procedure to develop the implementation.

For the development of the proposed system, the Raspberry Pi is used as a hardware platform and a camera for acquiring images of electronic cards. The validation of the system is carried out through tests using real data sets from electronic cards and finally the precision and efficiency of the proposed algorithm is evaluated in comparison with traditional methods such as visual inspection.

Keywords: Computer vision, segmentation, electronic cards, Raspberry Pi.

I ÍNDICE DE CONTENIDO

I	INTRODUCCIÓN	1
II	PROBLEMA	2
2.1	Delimitación Temporal	3
2.2	Delimitación Académica	3
2.3	Delimitación Espacial	4
III	OBJETIVOS	5
3.1	Objetivo General	5
3.2	Objetivos Específicos	5
IV	FUNDAMENTO TEÓRICO	6
4.1	Visión Artificial	6
4.2	Inteligencia Artificial	7
4.3	Machine Learning	7
4.4	Deep Learning	7
4.5	Redes neuronales	8
4.6	Redes neuronales convolucionales	8
4.6.1	Capa convolucional (Convolutional layer)	9
4.6.2	Capa de agrupación (Pooling layer)	10
4.6.3	Capa totalmente conectada (Full connected layer)	10
4.7	YOLO	11

4.8	Roboflow	12
4.8.1	Clasificación de imágenes.....	12
4.8.2	Detección de objetos.....	13
4.8.3	Segmentación.....	13
4.9	Raspberry Pi.....	13
4.10	Cámara de visión artificial	15
V	MARCO METODOLÓGICO	16
5.1	Configuración del Raspberry Pi OS.....	17
5.2	Recopilación de imágenes	26
5.3	Anotación de imágenes.....	27
5.4	Instalación de YOLO v8	29
5.4.1	Descarga de librería Ultralytics.....	29
5.4.2	Prueba de funcionamiento de librería Ultralytics.....	30
5.5	Interfaz de usuario	32
VI	RESULTADOS	34
6.1	Realizar el prototipo de segmentación de componentes electrónicos	34
6.2	Implementar el prototipo utilizando una RaspberryPi y una cámara	34
6.3	Realizar pruebas para comprobar el efectivo funcionamiento del proyecto	35
6.4	Validar el funcionamiento del prototipo a través de un juicio experto.....	37
6.4.1	Resultados de encuestas.....	38

6.4.2	Análisis de resultados	44
VII	CRONOGRAMA	45
VIII	PRESUPUESTO	46
IX	CONCLUSIONES.....	47
X	RECOMENDACIONES	48
XI	REFERENCIAS.....	49
XII	ANEXOS.....	52

Índice de Figuras

<i>Figura 1</i>	4
<i>Figura 2</i>	6
<i>Figura 3</i>	9
<i>Figura 4</i>	11
<i>Figura 5</i>	12
<i>Figura 6</i>	14
<i>Figura 7</i>	16
<i>Figura 8</i>	17
<i>Figura 9</i>	18
<i>Figura 10</i>	19
<i>Figura 11</i>	20
<i>Figura 12</i>	21
<i>Figura 13</i>	22
<i>Figura 14</i>	23
<i>Figura 15</i>	23
<i>Figura 16</i>	24
<i>Figura 17</i>	25
<i>Figura 18</i>	25
<i>Figura 19</i>	26
<i>Figura 20</i>	27
<i>Figura 21</i>	28
<i>Figura 22</i>	31

<i>Figura 23</i>	33
<i>Figura 24</i>	35
<i>Figura 25</i>	36
<i>Figura 26</i>	37
<i>Figura 27</i>	39
<i>Figura 28</i>	40
<i>Figura 29</i>	41
<i>Figura 30</i>	42
<i>Figura 31</i>	43
<i>Figura 32</i>	44

Índice de Tablas

<i>Tabla 1</i>	29
<i>Tabla 2</i>	38
<i>Tabla 3</i>	39
<i>Tabla 4</i>	40
<i>Tabla 5</i>	41
<i>Tabla 6</i>	42
<i>Tabla 7</i>	43
<i>Tabla 8</i>	45
<i>Tabla 9</i>	46

I INTRODUCCIÓN

El campo de la visión por computadora es muy interesante y está en constante evolución, múltiples estudios demuestran que la visión artificial es una disciplina científica que consta de diversas tecnologías que capturan, procesan y analizan imágenes para extraer información útil de imágenes digitales. Su objetivo es automatizar tareas en seguridad, industria, negocios, medicina y otros campos que hasta hace poco estaban reservados a los humanos. Muchas técnicas utilizadas en visión por ordenador provienen de otras disciplinas, como la inteligencia artificial o el aprendizaje profundo, que requieren amplios conocimientos matemáticos. En resumen, Esta tecnología es una rama de la inteligencia artificial que se centra en permitir que las máquinas observen, procesen y comprendan el mundo visual como lo hacen los humanos.

Para desarrollar este proyecto, la documentación se desglosa de la siguiente manera:

El Capítulo 1 examina los antecedentes del proyecto técnico, en el cual se describe la situación problemática a estudiar, la razones para llevar a cabo la implementación y objetivos planteados.

El Capítulo 2 explica el contexto de la información recopilada para la creación del sistema de segmentación de componentes electrónicos.

El Capítulo 3 describe el diseño e implementación del sistema de segmentación de componentes electrónicos mediante visión artificial.

El Capítulo 4 detalla las pruebas y muestra los resultados obtenidos en el proceso de segmentación de componente electrónicos. Para finalizar, se exponen las conclusiones y sugerencias derivadas de la interacción con la aplicación.

II PROBLEMA

En la actualidad el estudio de la forma, fondo y nombre técnico de los dispositivos electrónicos suele hacerse mediante comparaciones o en su defecto buscándolo de forma gráfica con la ayuda del internet. Por esta razón el tiempo de análisis y reparación de tarjetas electrónicas suele variar entre 45 a 120 minutos, al técnico le toma alrededor de 20 minutos el reconocimiento y clasificación de los diferentes tipos de dispositivos semiconductores que se encuentran alojados en las tarjetas electrónicas, debido a esto llegar a familiarizarse con las tarjetas electrónicas se convierte en un inconveniente sobre todo para los técnicos principiantes y con poca experiencia.

En la empresa MULTISERVICIOS ELECTRÓNICOS POLITÉCNICOS ubicada en la ciudad de Guayaquil, que se desempeña en reparaciones de tarjetas electrónicas (PCB) de diversos equipos y marcas, se establece la dificultad en el reconocimiento de los componentes y dispositivos que conforman las tarjetas electrónicas, lo cual retrasa y dificulta el progreso.

En este problema también se han visto inmersos técnicos amateurs y estudiantes de ingeniería, ya que la poca o nula interacción con distintos tipos de dispositivos electrónicos es cada vez más común, debido a que en las prácticas académicas de laboratorio se utilizan elementos básicos lo que limita conocerlos todos, también se debe incluir como otra limitante el incremento de prototipos de las industrias de semiconductores electrónicos que día a día van innovando y sacan de manera comercial elementos cada vez diferentes.

Este desconocimiento o poca familiarización con los elementos que vienen instalados en las tarjetas electrónicas ha sido reconocido como una dificultad en el ámbito laboral, para los

especialistas que se deben encargar tanto en el desarrollo mantenimiento y reparación de estas tarjetas.

En MULTISERVICIOS ELECTRÓNICOS POLITÉCNICOS, se ha observado esta deficiencia sobre todo en técnicos e ingenieros que nunca antes estuvieron familiarizados con las mismas y en vista de este inconveniente esta empresa ha puesto a disposición su problemática.

Recolectando más información sobre este problema queda al descubierto que absolutamente todos los laboratorios y talleres de reparaciones de tarjetas electrónicas padecen de esta problemática con los técnicos e ingenieros que ingresan por primera vez en las filas laborales, quedando demostrado que es una constante para todos los profesionales que comienzan en el desarrollo de reparaciones de tarjetas electrónicas, el inconveniente de reconocer los elementos que conforman una tarjeta PCB.

2.1 Delimitación Temporal

El proceso de elaboración del proyecto de titulación es de cinco meses, iniciando desde la fecha del visto bueno propuesto por las autoridades que dirigen la carrera de ingeniería en electrónica.

2.2 Delimitación Académica

El objetivo del desarrollo e implementación de este dispositivo de segmentación y reconocimiento de elementos de una tarjeta PCB es motivar a más estudiantes el uso de nuevas tecnologías e implementarlas en sus ideas y de esta manera ayuden a contribuir al desarrollo intelectual y comercial del país, ya que de esta forma utilizamos los recursos de aprendizaje que se obtuvieron durante todo el proceso académico en la Universidad Politécnica Salesiana.

2.3 Delimitación Espacial

El lugar geográfico se encuentra en la ciudad de Guayaquil direccionado en Portete de Tarqui 1619 entre José de Antepara y Av. Machala, como se muestra en la Figura 1.

Figura 1

Ubicación del sector de estudio.



Nota: Ubicación geográfica de Multiservicios Electrónicos Politécnicos.

III OBJETIVOS

3.1 Objetivo General

Diseñar un sistema de segmentación de componentes electrónicos en tarjetas electrónicas mediante visión artificial para el mejoramiento de la capacidad de reparación en los PCB's.

3.2 Objetivos Específicos

- Realizar el prototipo de segmentación de componentes electrónicos en tarjetas electrónicas.
- Implementar el prototipo utilizando una RaspberryPi y una cámara.
- Realizar pruebas para comprobar el adecuado desempeño del proyecto.
- Validar el funcionamiento del prototipo a través de un juicio experto.

IV FUNDAMENTO TEÓRICO

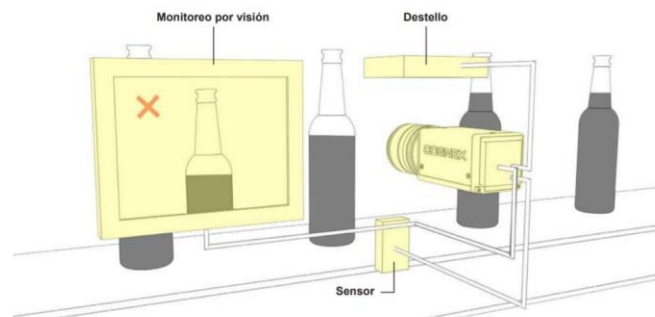
4.1 Visión Artificial

Se basa en una mezcla de software y hardware para capturar imágenes y procesar su información. Recopila datos del entorno a través de imágenes que puede analizar, empleando diversos sensores para obtener resultados altamente precisos. En el ámbito industrial, desempeña un papel integral en la cadena de producción gracias a la recepción y transmisión de datos. (UNIR, 2021)

La visión artificial intenta simular el proceso visual humano y analizarlo a través del cerebro, al igual que los humanos. Si los seres humanos pueden compartir imágenes capturadas por la visión y analizarlas mediante señales enviadas al cerebro., entonces las máquinas pueden usar cámaras de red para capturar imágenes, que se envían a un procesador que las prepara para el análisis, de manera que la máquina pueda gestionar el color y la forma de ciertos objetos, esto se puede apreciar en la Figura 1, que ilustra los componentes de un sistema de visión artificial. (Amaya, Pulgarín, & Torres, 2016).

Figura 2

Sistema de visión artificial



Nota: Elementos de un sistema de visión computarizada. (Bcnvisión, 2017)

4.2 Inteligencia Artificial

La inteligencia artificial implica el análisis de cómo lograr que las computadoras ejecuten tareas que anteriormente demandaban capacidades propias de la inteligencia humana. Busca crear algoritmos y sistemas que posibiliten a las máquinas percibir, razonar, aprender, tomar decisiones y resolver problemas de manera análoga a los seres humanos., pero no necesariamente de la misma manera. (Russell & Norving, 2008)

4.3 Machine Learning

Machine Learning, o aprendizaje automático en español, tiene como objetivo desarrollar tecnología que permita aprender a ordenadores o máquinas con capacidad de procesamiento. Pero las máquinas no aprenden por sí solas; están programados para ajustar algoritmos a medida que reciben datos. Cuantos más datos obtengan, mejores algoritmos podrán crear, siempre que los datos de entrada sean confiables y de buena calidad, porque ingresar datos incorrectos puede conducir a un aprendizaje deficiente. (de Frías, 2018)

4.4 Deep Learning

Deep Learning, o aprendizaje profundo en español, representa una evolución del Machine Learning que permite al ordenador aprender a través de la experiencia y así poder entender su entorno. Su ventaja principal es el aprendizaje sin necesidad de la intervención humana, sacando de forma autónoma las conclusiones de todos los datos. Este sistema es capaz no solamente de reconocer objetos en las imágenes o videos, sino también entender lo que están percibiendo u observando. (de Frías, 2018)

4.5 Redes neuronales

Su nombre y configuración resalta de la inspiración de las neuronas cerebrales humanas dado que imitan su forma biológica señalándose entre sí. Las redes neuronales forman parte de Machine Learning y representan el núcleo de los algoritmos de Deep Learning. Las redes neuronales se fundamentan en el entrenamiento de datos para adquirir conocimientos y perfeccionar su precisión con el transcurso del tiempo. (Amazon Web Services, s.f.)

Las capas de nodos son las que dan forma a las redes neuronales, ya que incluyen una capa de entrada, una o varias capas ocultas y una capa de salida. Cada neurona artificial (nodo) se conecta con otros y posee un peso y un umbral asociados. Si la salida de un nodo individual supera el valor del umbral establecido, dicho nodo se activa y transmite datos a la siguiente capa de la red. En caso contrario, no se transfieren datos a la siguiente capa de la red. (IBM, s.f.)

Según Amazon Web Service (s.f.) las redes neuronales se utilizan en varios sectores, entre los que podemos destacar:

- Proceso y control de calidad
- Estimación de carga eléctrica y requerimientos energéticos
- Diagnóstico clínico mediante la clasificación de imágenes médicas.
- Predicciones financieras mediante el procesamiento de datos históricos

4.6 Redes neuronales convolucionales

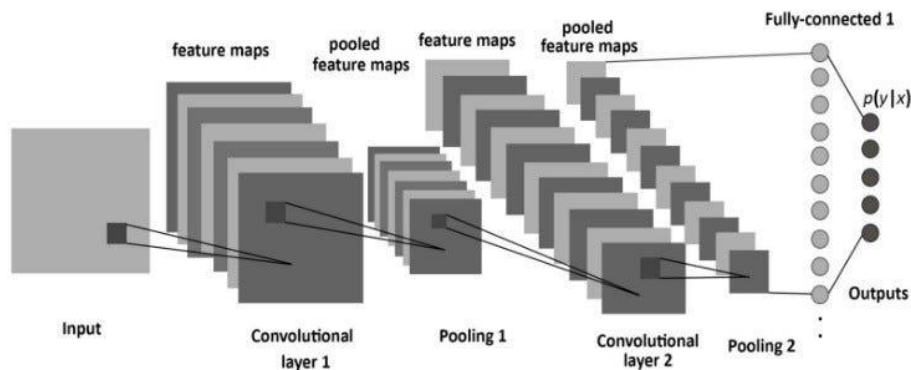
Las redes neuronales convolucionales, conocidas como Convolutional Neural Networks (CNN) en inglés, es un modelo desarrollado a partir de la década de 1950, por parte de los investigadores Hubel y Wiesel basándose en la corteza visual animal. Las redes neuronales

convolucionales presentan múltiples capas para calcular la salida de un conjunto de datos y su objetivo principal es el realizar el procesamiento de imágenes. (Bonilla, 2020)

Las redes neuronales convoluciones presentan un mayor rendimiento frente al procesamiento de imágenes, voz o señales de audio. Según se visualiza en la Figura 3, la estructura de las redes neuronales convolucionales consta de tres tipos primordiales de capas.:

Figura 3

Arquitectura de las redes neuronales convolucionales.



Nota: Redes neuronales convolucionales Nota. (Admin, 2023)

4.6.1 Capa convolucional (Convolutional layer)

Esta es la capa central donde se efectúan la mayor parte de los cálculos en las redes neuronales convolucionales, siendo indispensables el uso de una o de más capas de este tipo en dichas redes. Requiere algunos parámetros, como información de entrada, un filtro y un mapa de características. (IBM, s.f.)

El propósito de la red será aprender filtros que se activan cuando hay una función específica en una región determinada de la entrada. (Bonilla, 2020)

4.6.2 Capa de agrupación (Pooling layer)

La capa de agrupación, conocida también como submuestreo, toma los mapas de características producidos en la capa de convolución y los fusiona en una imagen. En esta etapa, se disminuye la complejidad del modelo, contribuyendo a prevenir el sobreajuste. En resumen, las capas de pooling simplifican la información en la salida de la capa convolucional. La capa de agrupación comúnmente es colocada entre las capas convolucionales sucesivas. (Bonilla, 2020)

Existen dos tipos principales de agrupación:

- **Agrupación máxima:** Durante el recorrido del filtro en la entrada, este selecciona el píxel de valor más alto para transmitirlo a la matriz de salida.
- **Agrupación media:** Mientras el filtro se mueve a lo largo de la entrada, realiza el promedio del valor en el campo receptivo y posteriormente lo envía a la matriz de salida.

Aunque la capa de agrupación pierde mucha información, esta ayuda a simplificar la complejidad, mejorar la eficiencia y minimizar el riesgo de redundancias. (IBM, s.f.)

4.6.3 Capa totalmente conectada (Full connected layer)

Cada neurona en esta capa establece enlaces con todas las neuronas de la capa precedente, más específicamente, están enlazadas con todos los mapas de características de la capa precedente. La función principal de la capa completamente conectada es agrupar la información adquirida actualmente para usarla en el cálculo de clasificación final posterior. En definitiva, la base de esta capa para realizar la tarea de clasificación es utilizar la función

de activación softmax para extraer información de la capa predecesora, y sus características se extraen de varios filtros para clasificar correctamente la entrada y generar probabilidades entre 0 y 1. (IBM, s.f.)

4.7 YOLO

You Only Look Once mejor conocido como YOLO por sus siglas en inglés, es un sistema de código abierto conocido y utilizado por llevar a cabo la segmentación de imágenes y detección de objetos, fue desarrollado en 2016 por Joseph Redmon y Ali Farhadi. A través de los años se han propuesto diferentes versiones, entre las que destaca YOLOv8 dado que admite una gama de tareas de visión artificial más amplia (detección, clasificación, segmentación, seguimiento y estimación de pose), además de presentar un mayor rendimiento, precisión y estabilidad. (Ultralytics, s.f.)

En la Figura 4 se puede observar tres tareas de visión que puede realizar YOLO v8:

Figura 4

Clasificación, detección y segmentación de componentes electrónicos.



Nota: Tareas de visión artificial que realiza YOLO V8. (AlexFocus, 2023)

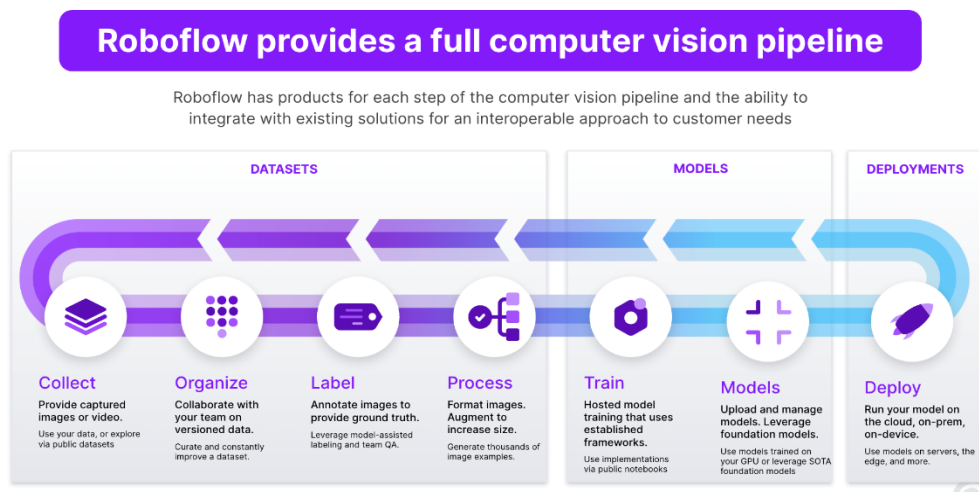
4.8 Roboflow

Se trata de una herramienta de inteligencia artificial que capacita a los usuarios para desarrollar modelos de visión por computadora funcional en pocas horas utilizando unas cuantas imágenes de ejemplo. Simplifica el proceso, puesto que proporciona las herramientas necesarias y algoritmos de aprendizaje automático, permitiendo desarrollar aplicaciones prácticas que pueden reconocer objetos, rostros y realizar diversas tareas visuales. (Aplicaciones AI, 2023)

En la Figura 5 se muestran las actividades que se pueden realizar en Roboflow.

Figura 5

Herramientas que proporciona Roboflow para crear modelos de visión artificial.



Nota: Herramientas para el desarrollo de visión artificial. (Roboflow, s.f.)

4.8.1 Clasificación de imágenes

Un modelo de clasificación de imágenes permite tomar una o varias imágenes y extraer una clase correspondiente a cada una de ellas. (Tech AMP, 2021)

4.8.2 Detección de objetos

Un modelo de detección de objetos muestra cada una de las clases y coordenadas de los objetos para los que fue entrenado, arroja las coordenadas de cada una de las detecciones para poder localizarlas dentro de la imagen. (Tech AMP, 2021)

4.8.3 Segmentación

En los modelos de segmentación se van a encontrar distintos elementos dentro de la imagen, pero en vez de darnos las coordenadas del rectángulo que enmarca el elemento, nos indicará el grupo de píxeles representa cada objeto. La ventaja de la segmentación es que esta no solamente indica el objeto y el rectángulo limitante, sino que también nos dice la forma de los objetos que detecto. (Tech AMP, 2021)

4.9 Raspberry Pi

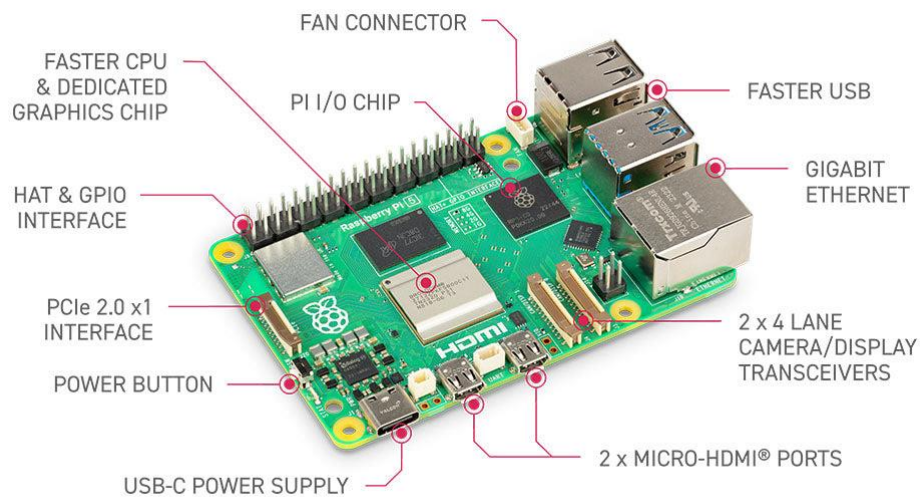
Es una pequeña computadora, creada en el año 2012 por la Raspberry Pi Foundation, que ejecuta Linux con el objetivo de permitir que individuos de todas las edades exploren computadoras y adquieran conocimientos en lenguajes de programación como Scratch y Python, esta herramienta de inteligencia artificial es capaz de gestionar diversas tareas, desde las más básicas, como la navegación por la web, reproducción de vídeos en alta definición y trabajo con documentos de Office, hasta actividades de entretenimiento como los juegos. (Raspberry Pi, 2022)

Cuenta con un procesador de 64 bits Arm Cortex A76 de cuatro núcleos con una velocidad de 2.4GHz, un procesador gráfico VideoCore VII y una memoria RAM de hasta 8GB, también dispone de distintos puertos de conexión, los cuales serán enlistados a continuación: (Raspberry Pi, s.f.)

- 2 puertos USB 3.0
- 2 puertos USB 2.0
- 2 puertos HDMI 4Kp60
- 1 puerto Gigabit Ethernet
- 1 puerto de alimentación CC mediante USB-C
- 1 ranura micro SD
- Conexiones WiFi y Bluetooth

Figura 6

Raspberry Pi 5



Nota: Partes del Raspberry Pi 5. (Raspberry Pi, s.f.)

4.10 Cámara de visión artificial

Una cámara digital se conecta a una computadora para capturar y enviar imágenes a través de Internet, representando una evolución tecnológica respecto a las cámaras fotográficas antiguas que usaban película química. (Porto & Merino, 2022)

En el campo de la visión artificial, las cámaras de visión desempeñan la tarea de registrar la imagen proyectada en el sensor mediante lentes, para luego transferirla a un sistema electrónico. Estas cámaras proporcionan un control completo sobre aspectos como los tiempos, señales y la velocidad de obturación, siendo esenciales en sistemas de visión artificial para aplicaciones tanto científicas como industriales. (INFAIMON, 2023)

V MARCO METODOLÓGICO

En este capítulo se examina el procedimiento llevado a cabo para poner en marcha el proyecto, detallando de manera gradual la preparación del software y el desarrollo del proceso de segmentación de componentes electrónicos en tarjetas electrónicas con visión artificial.

En la Figura 7 se detallan los pasos que se siguieron para la ejecución del modelo de segmentación de componentes electrónicos con visión artificial.

Figura 7

Etapas del sistema de segmentación de componentes



Nota: Pasos a seguir para el desarrollo del sistema.

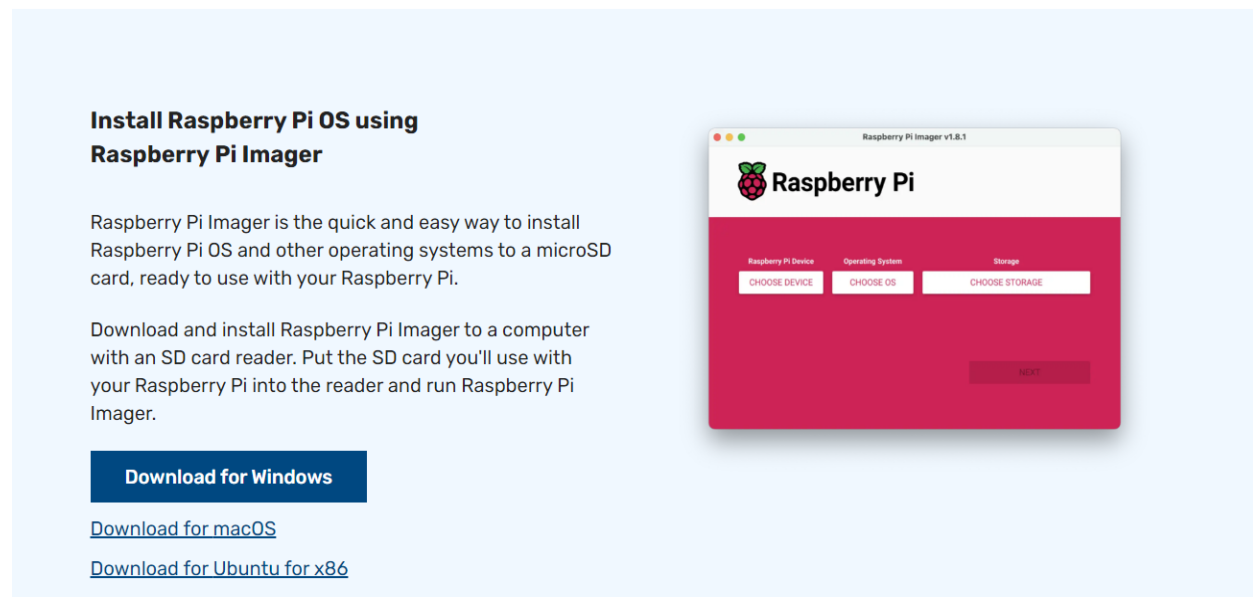
Para realizar el prototipo de segmentación de componentes electrónicos en tarjetas electrónicas e implementarlo utilizando una Raspberry Pi y una cámara se ejecutan las siguientes acciones.

5.1 Configuración del Raspberry Pi OS

Para implementar el sistema operativo, es necesario obtener el archivo ejecutable “Raspberry Pi Imager” el cual se encuentra en la página oficial de la empresa, tal como se muestra en la Figura 8, luego se realiza el almacenamiento de la imagen en una tarjeta microSD.

Figura 8

Sitio web de Raspberry Pi

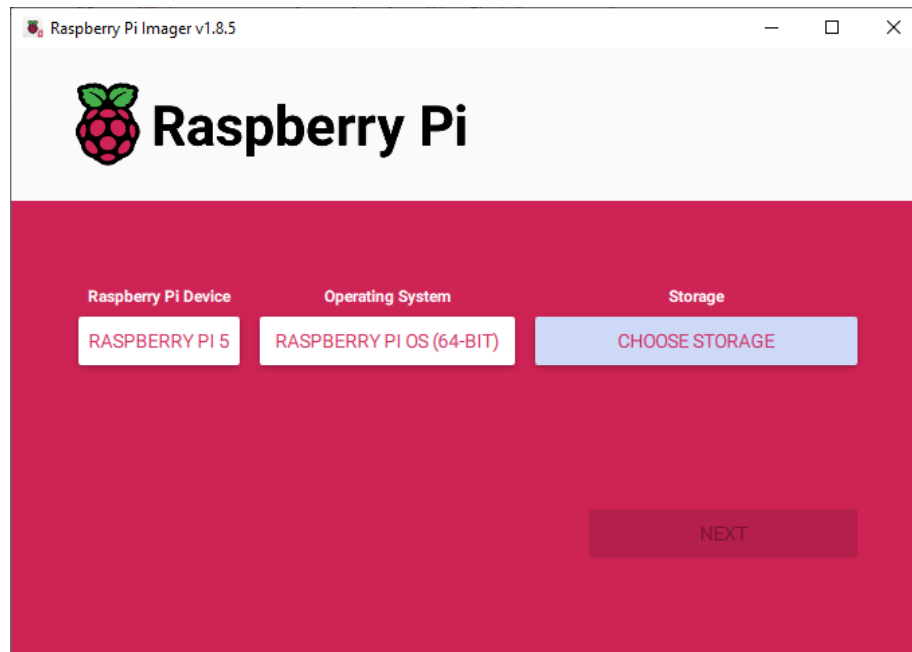


Nota: Enlace de descarga del Raspberry Pi Imager. (Raspberry Pi, s.f.)

Realizada la instalación del Raspberry Pi Imager se lo ejecuta para grabar la información en una tarjeta micro SD. En la Figura 9 se presenta la interfaz del Raspberry Pi Imager, en la cual se debe elegir el tipo de placa a utilizar, el sistema operativo y el dispositivo en donde se va a almacenar la información.

Figura 9

Interfaz del Raspberry Pi Imager



Nota: Elección de dispositivo, sistema operativo y almacenamiento.

Finalizado el proceso de almacenamiento se procede a extraer la micro SD del computador y se inserta de nuevo, esto debido a que se tiene que crear un archivo de nombre “ssh” sin extensión el cual es el que permite realizar el acceso remoto al Raspberry Pi desde el computador.

Como siguiente paso, se retira la tarjeta micro SD y se coloca en la ranura del módulo Raspberry Pi. La configuración del sistema operativo se lo realiza de forma remota mediante Real VNC Viewer.

Para iniciar la instalación se debe enlazar el Raspberry Pi a un router mediante un cable de red y conectar la fuente de alimentación para energizar el módulo. En la Figura 10 se observa el módulo con las conexiones antes especificadas.

Figura 10

Raspberry Pi con fuente de alimentación y conexión a la red.

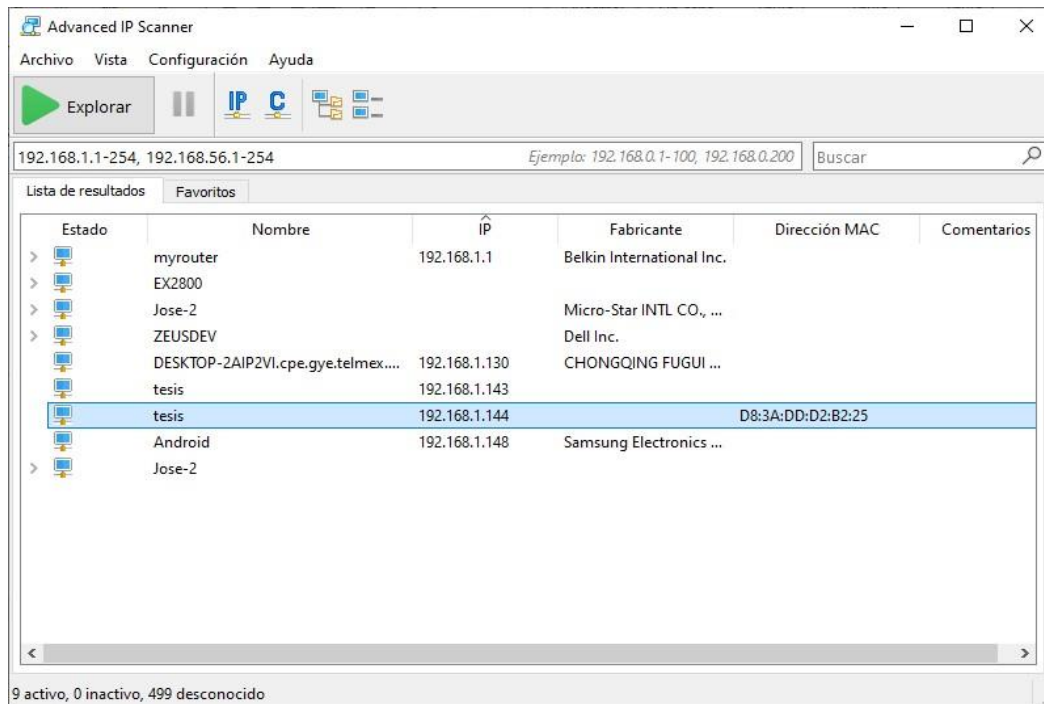


Nota: Conexión del Raspberry para establecer control remoto.

Como resultado del enlace entre el Raspberry Pi y el router se asigna una IP al módulo, para conocer que IP fue asignada al módulo se utiliza el programa Advanced IP Scanner, el cual es un programa que escanea todos los dispositivos de red. Abierto el programa se hace clic en la opción Explorar y en unos pocos segundos muestra todos los dispositivos conectados a la red, tal y como se muestra en la Figura 11.

Figura 11

Interfaz de Advanced IP Scanner



Nota: Dispositivos conectados a la red.

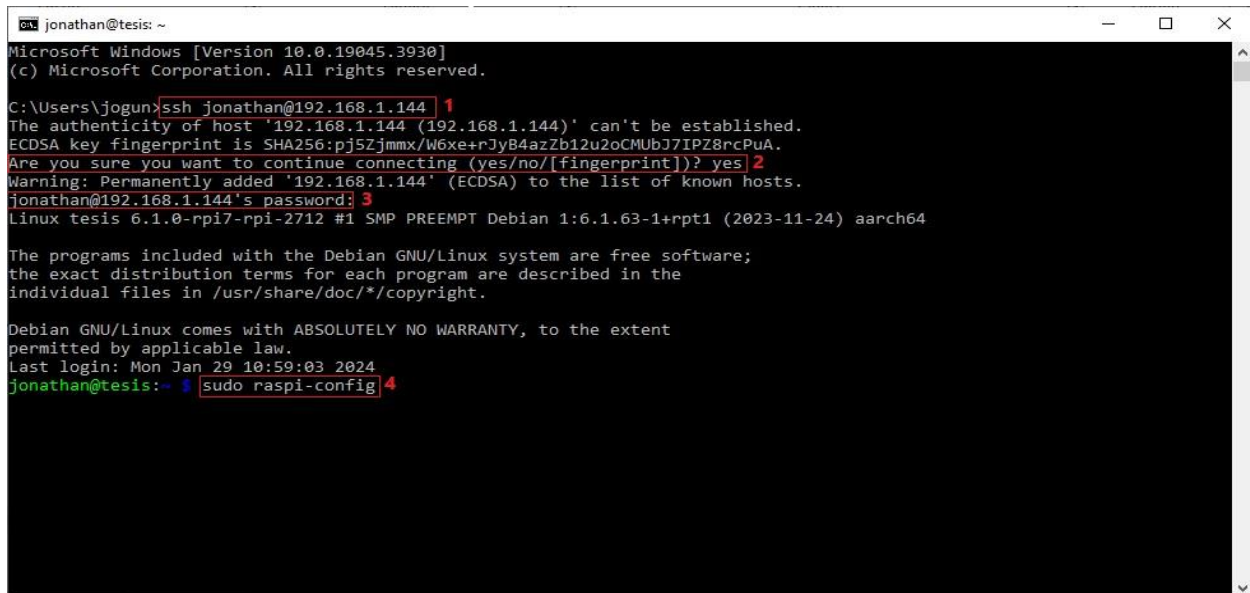
En la Figura 11 se muestra que la dirección IP asignada al Raspberry Pi es 192.168.1.144, teniendo la IP se abre el terminal del sistema y se realizan los siguientes pasos:

- Escribir en el terminal: `ssh "nombre de usuario"@"dirección IP"`
- Responder "Yes" a la pregunta que se formula
- Escribir la contraseña de acceso al Raspberry
- Habilitar VNC Server de la Raspberry Pi

En la Figura 12 se exhibe la ejecución de los pasos detallaron anteriormente.

Figura 12

Ejecución de comandos en símbolo de sistema



```
jonathan@tesis: ~
Microsoft Windows [Version 10.0.19045.3930]
(c) Microsoft Corporation. All rights reserved.

C:\Users\jogun>ssh jonathan@192.168.1.144
The authenticity of host '192.168.1.144 (192.168.1.144)' can't be established.
ECDSA key fingerprint is SHA256:pj5Zjmmx/W6xe+rJyB4azZb12u2oCMUbj7IPZ8rcPuA.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.1.144' (ECDSA) to the list of known hosts.
jonathan@192.168.1.144's password:
Linux tesis 6.1.0-rpi7-rpi-2712 #1 SMP PREEMPT Debian 1:6.1.63-1+rpt1 (2023-11-24) aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

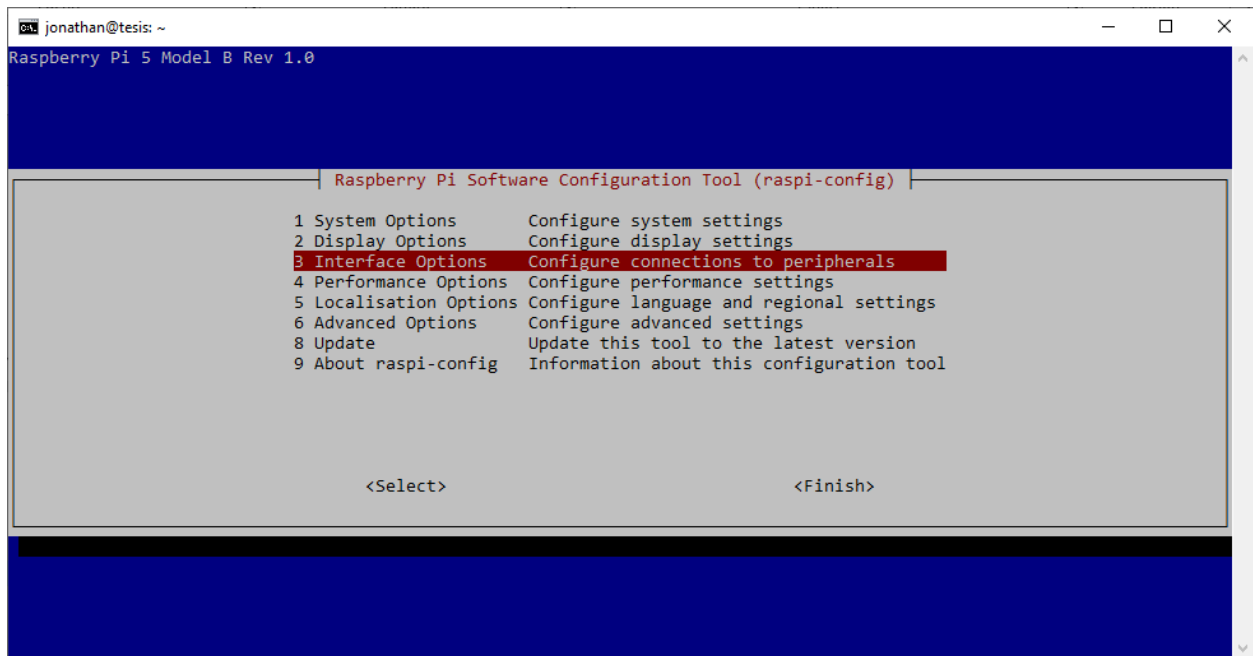
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Jan 29 10:59:03 2024
jonathan@tesis: ~$ sudo raspi-config
```

Nota: Pasos para habilitar el acceso remoto al Raspberry Pi.

En la Figura 13 se evidencia la Herramienta de Configuración de Software del Raspberry Pi, para acceder a la herramienta se utiliza el comando `sudo raspi-config`. En la ventana de la herramienta de configuración se enlistan varias opciones, pero se escogerá la opción 3, Interface Options, el cual permite configurar las conexiones de los periféricos.

Figura 13

Herramienta de Configuración de Software

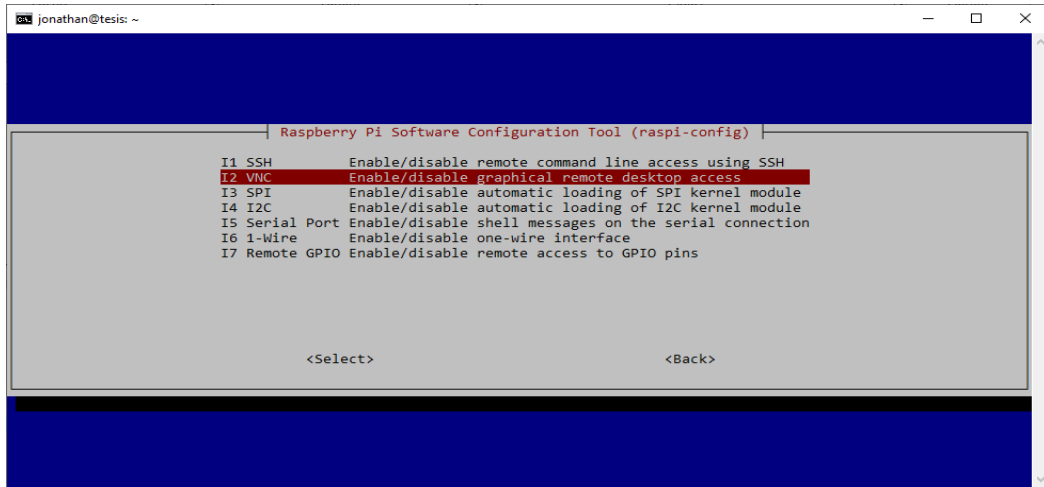


Nota: Selección de opción "Interface Options".

Seleccionada la opción 3 se despliega otro menú el cual se puede visualizar en la Figura 14, el cual muestra diferentes opciones. Se escoge la opción 2 la cual permite habilitar VNC Server.

Figura 14

Sub menú de Interface Options

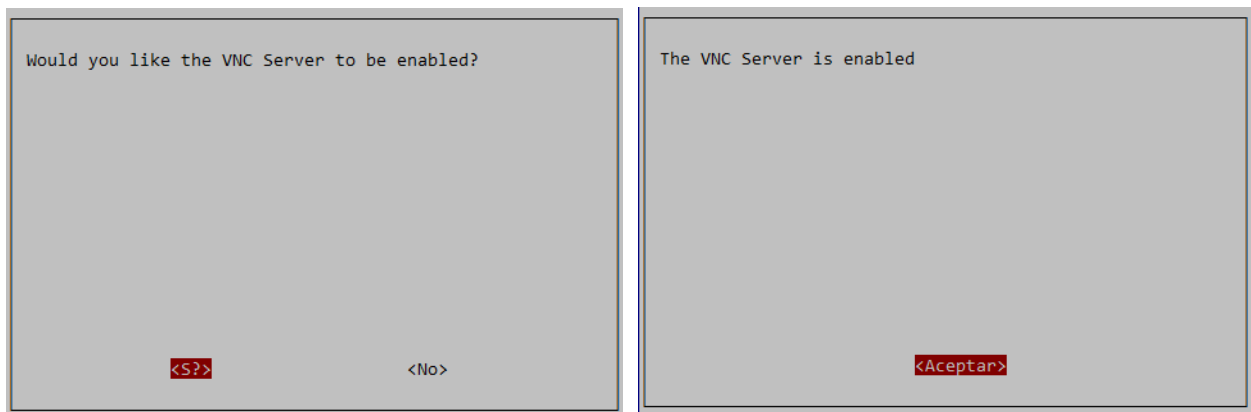


Nota: Selección de opción “VNC”.

Seleccionada la opción 2, se abre una nueva ventana preguntando si se quiere activar VNC Server a lo cual se responde que SI, posteriormente aparece una ventana confirmando la habilitación del VNC Server en el Raspberry Pi, esto se puede apreciar en la Figura 15.

Figura 15

Confirmar activación de VNC

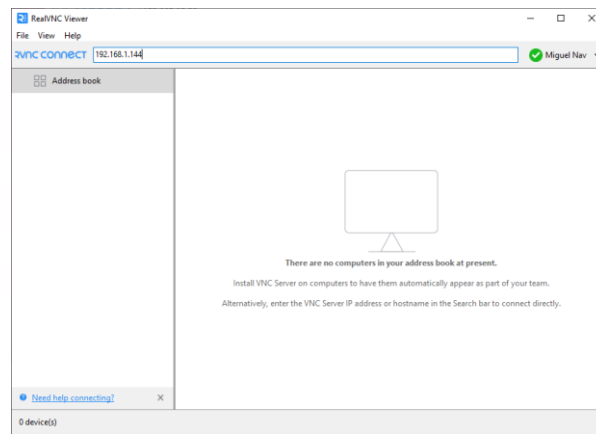


Nota: Aceptar habilitación de VNC y mensaje de confirmación.

Para administrar de forma gráfica el Raspberry desde el computador es necesario tener instalado Real VNC Viewer. Con el programa ya instalado se procede a ejecutarlo y mostrará la ventana de inicio de VNC, aquí se debe digitar la dirección IP correspondiente al Raspberry Pi, así como se visualiza en la Figura 16.

Figura 16

Interfaz de Real VNC Viewer

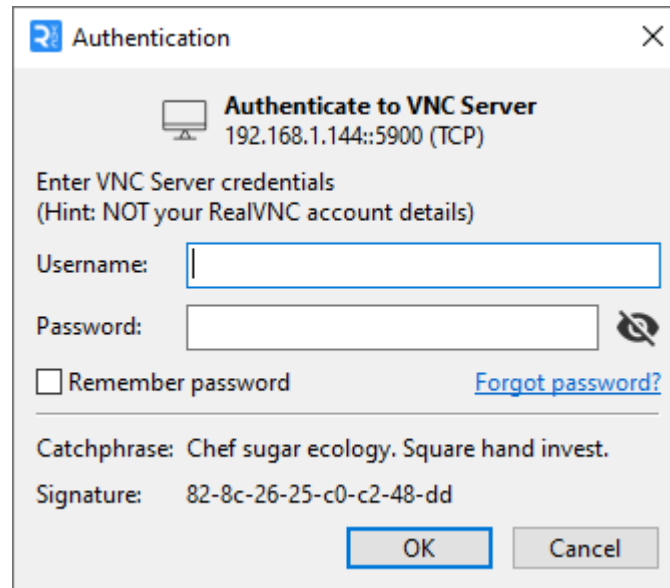


Nota: Ingreso de dirección IP en barra de búsqueda.

Ingresa la dirección IP solicita digitar las credenciales de autenticación, las cuales por defecto son “pi” como nombre de usuario y “raspberrypi” como contraseña. En la figura 17 se presenta la ventana de autenticación y en la Figura 18 se evidencia el escritorio de la Raspberry Pi.

Figura 17

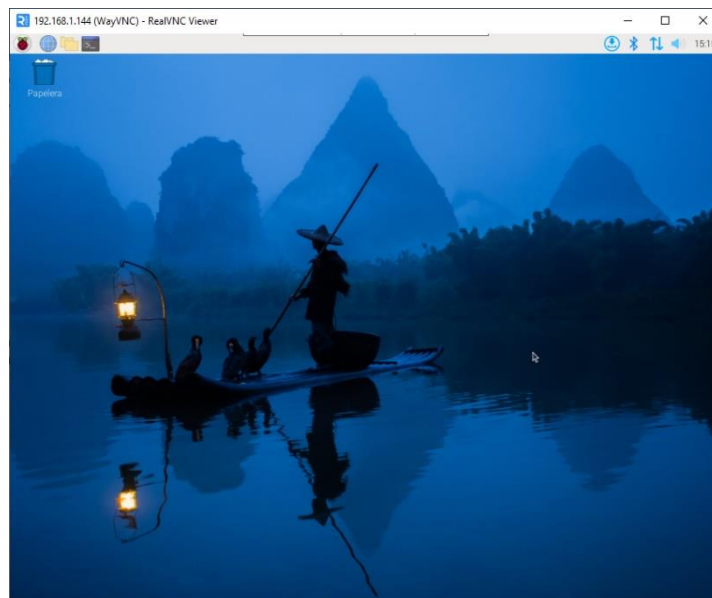
Ventana de autenticación de VNC Server



Nota: Usuario: pi (default), Contraseña: raspberry (default).

Figura 18

Escritorio de Raspberry Pi



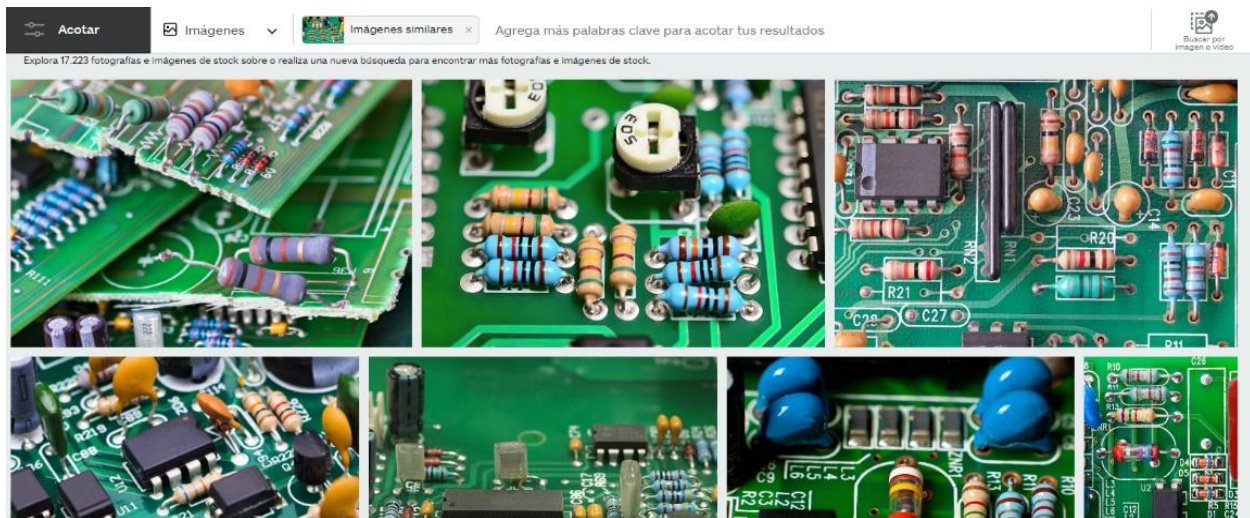
Nota: Se puede configurar de acuerdo a las necesidades del autor.

5.2 Recopilación de imágenes

En el proceso de desarrollo del modelo de segmentación, se realizó la recopilación de un conjunto variado de imágenes que representan los distintos componentes electrónicos presentes en las tarjetas electrónicas. Dada la limitación en la disponibilidad de tarjetas electrónicas específicas para nuestra investigación, se tomó la decisión de obtener imágenes de la plataforma IStock. Esta plataforma, conocida por albergar una extensa colección de recursos visuales de libre licencia, nos brindó acceso a una variedad de fotografías que capturan la esencia de los elementos electrónicos que buscábamos analizar en nuestro proyecto. La elección de IStock permitió enriquecer nuestra base de datos visual y garantizar la representatividad de los componentes electrónicos en nuestro conjunto de datos, contribuyendo así a la robustez y generalización del modelo de segmentación que estábamos desarrollando. En la Figura 19 se muestra algunas de las imágenes para que se logró recolectar del sitio web.

Figura 19

Sitio web IStock



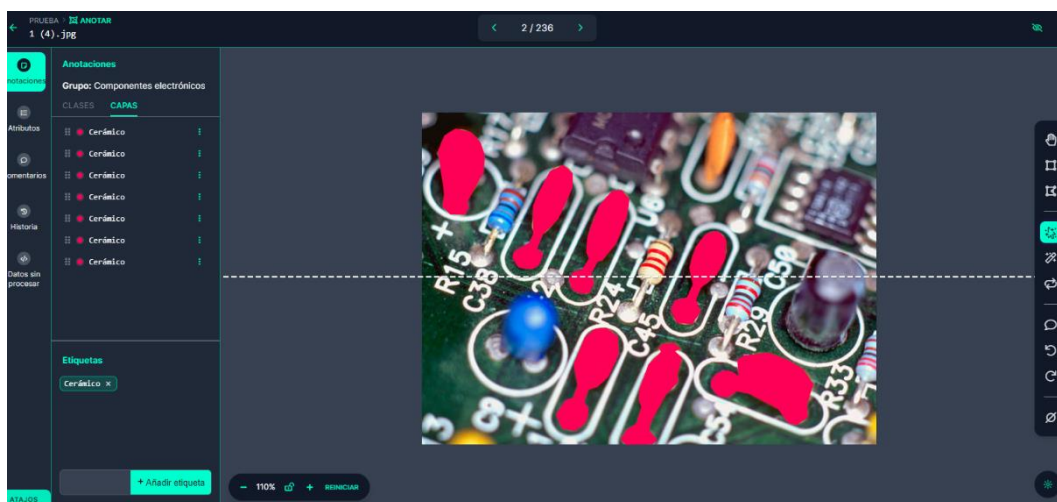
Notas: Búsqueda de componentes electrónicos en tarjetas electrónicas. (IStock, 2023)

5.3 Anotación de imágenes

La tarea de etiquetado y segmentación de cada imagen se llevó a cabo de manera manual utilizando la plataforma Roboflow. Este sitio web se erige como una herramienta integral que posibilita no solo el etiquetado y segmentación de imágenes, sino también el preprocesamiento de las mismas. Roboflow ofrece una funcionalidad versátil al permitir la creación de modelos para detección, clasificación o segmentación de manera eficiente. Su interfaz intuitiva y capacidades avanzadas facilitaron el proceso de anotación de nuestras imágenes, asegurando una preparación óptima de los datos para la posterior implementación y entrenamiento del modelo de segmentación. La elección de Roboflow se fundamenta en su capacidad para agilizar y simplificar el flujo de trabajo, optimizando así la calidad y precisión de las anotaciones realizadas dentro del contexto de nuestra investigación. En la Figura 20 se muestra el proceso de segmentación manual realizado en cada una de las imágenes recolectadas.

Figura 20

Segmentación de componentes electrónicos

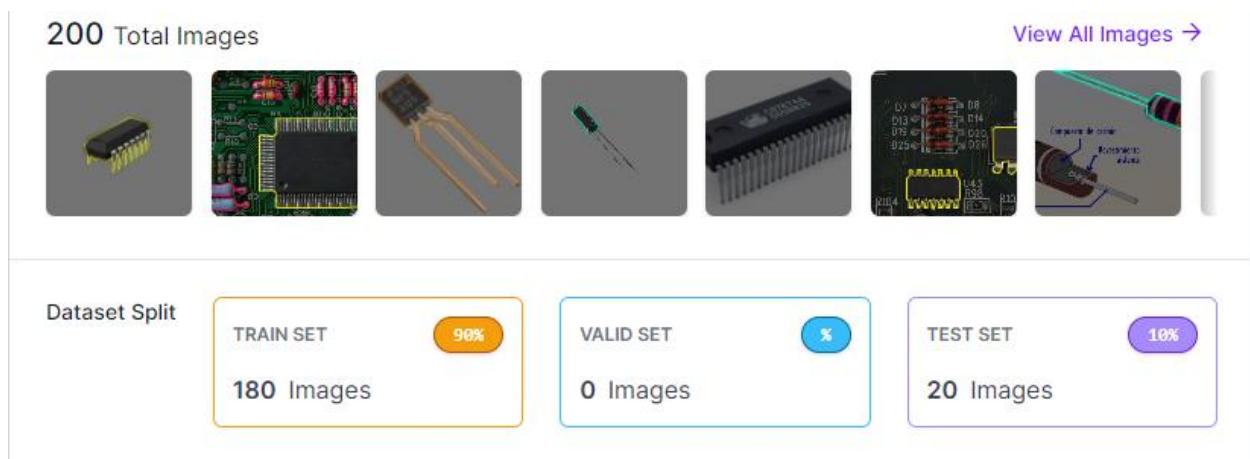


Nota: Anotación y etiquetado manual de componentes.

Una vez completada la anotación, etiquetado y segmentación de todas las imágenes, el siguiente paso implica la creación del conjunto de datos utilizando la misma plataforma de Roboflow. Antes de consolidar el conjunto de datos, esta herramienta ofrece la flexibilidad de seleccionar el porcentaje de imágenes destinadas a las etapas de capacitación, validación y pruebas del modelo que se va a entrenar. Esta funcionalidad permite una personalización eficiente de la distribución de datos, posibilitando una cuidadosa configuración de las proporciones para cada etapa del proceso. La capacidad de ajustar estas proporciones según las necesidades específicas del proyecto añade un nivel adicional de control y adaptabilidad al proceso de preparación de datos, contribuyendo así a la optimización del rendimiento del modelo a desarrollar, esto se puede visualizar en la Figura 21. Una vez que disponemos del conjunto de datos, podemos proceder con el entrenamiento del modelo.

Figura 21

Partición de imágenes



Nota: El grupo de entrenamiento cuenta con un 90% de las muestras totales, mientras que el grupo de prueba cuenta con el 10% de todas las muestras.

5.4 Instalación de YOLO v8

Para realizar la construcción del modelo de segmentación de componentes electrónicos en tarjetas electrónicas mediante visión artificial, se deben seguir una serie de pasos bien definidos. Al comienzo del proceso, se procede con la adquisición y configuración de la biblioteca Ultralytics, la cual es un componente esencial para la ejecución exitosa del modelo.

5.4.1 Descarga de librería Ultralytics

La primera tarea en este proceso implica la instalación de las dependencias necesarias para garantizar que Ultralytics se descargue e instale correctamente. Para llevar a cabo este paso, se abre el símbolo del sistema, comúnmente conocido como CMD, y se introduce el comando "*pip3 install*", seguido del nombre de cada librería y su versión correspondiente.

La Tabla 1 proporciona un desglose detallado de las librerías requeridas, cada una con su respectiva versión y breve descripción.

Tabla 1

Dependencias necesarias para instalar biblioteca Ultralytics

Librería	Versión	Descripción
matplotlib	3.3.0	Generador de gráficos
numpy	1.22.2	Cálculos matemáticos
opencv-python	4.6.0	Visión artificial
pillow	7.1.2	Soporte de imágenes
pyyaml	5.3.1	Serialización de imágenes
requests	2.23.0	Biblioteca HTTP
scipy	1.4.1	Cálculo científico
torch	1.8.0	Redes neuronales dinámicas
torchvision	0.9.0	Transf. para visión artificial
tqdm	4.64.0	Barras de progreso
psutil	-	Utilización del sistema
py-cpuinfo	-	Mostrar información CPU
thop	0.1.1	Cálculo de FLOP

pandas	1.1.4	Trabajar archivos cvs, xlsl
seaborn	0.11.0	Visualización de datos estadísticos

Nota: Desglose de librerías junto con la versión mínima requerida y una breve descripción.

Una vez completada la instalación de estas dependencias, se procede a la instalación de la librería Ultralytics. Este paso se realiza nuevamente a través del comando "`pip3 install ultralytics`". Durante este proceso, el sistema descarga e instala Ultralytics, asegurando así que la biblioteca esté disponible y lista para su implementación en el desarrollo del modelo de segmentación.

Este enfoque detallado y paso a paso para la descarga e instalación de Ultralytics garantiza un entorno adecuado y optimizado para el desarrollo del modelo YOLO v8. Además, la inclusión de las dependencias específicas proporciona un marco de trabajo completo que abarca desde la generación de gráficos hasta el cálculo científico y la visualización de datos estadísticos, asegurando así la eficacia y robustez del modelo en el proceso de segmentación de componentes electrónicos en tarjetas electrónicas.

5.4.2 Prueba de funcionamiento de librería Ultralytics

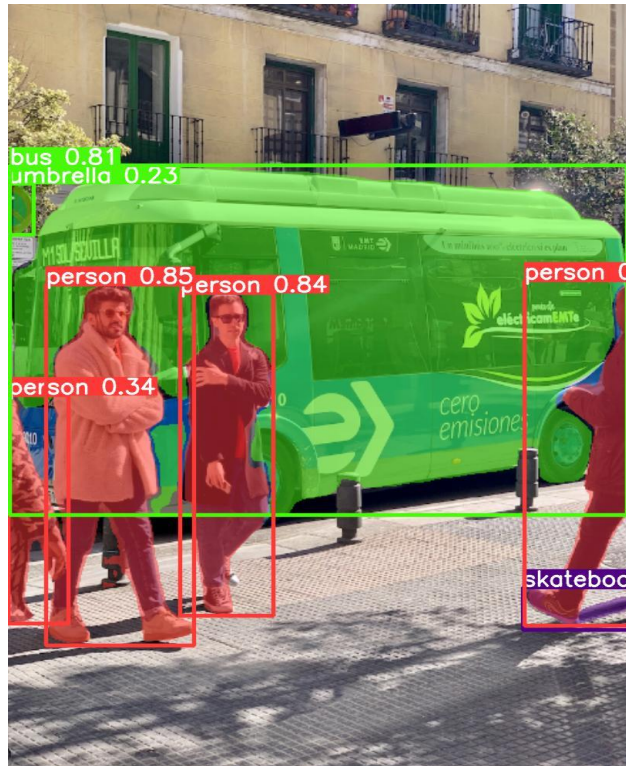
Concluida la instalación de la biblioteca, se procede a realizar un breve ejemplo con el objetivo de verificar el correcto funcionamiento de la segmentación de objetos utilizando YOLO v8. Este paso práctico proporciona una validación inicial del entorno y de la implementación de la técnica de segmentación en el contexto del modelo desarrollado.

Para llevar a cabo esta comprobación, se ejecuta un pequeño ejemplo que ilustra la capacidad de YOLO v8 para identificar y segmentar objetos en una imagen específica. La Figura 22 presenta visualmente los resultados obtenidos de esta prueba, ofreciendo

una representación gráfica de cómo el modelo ha logrado identificar y delimitar los objetos de interés en la imagen de prueba.

Figura 22

Segmentación de objetos con YOLO v8



Nota: Cada clase se identifica con un color único. (Roboflow, 2023)

Esta fase de verificación no solo sirve como un control de calidad inicial, sino que también proporciona una comprensión precisa de la eficacia del modelo en la tarea de segmentación de objetos. La visualización de los resultados en la Figura 22 permite evaluar de manera tangible la capacidad del modelo YOLO v8 para reconocer y delimitar correctamente los objetos en la imagen de prueba.

5.5 Interfaz de usuario

Con el propósito de asegurar la practicidad y eficacia del sistema, se diseñó una interfaz de usuario que posibilita la visualización en tiempo real de la imagen capturada por la cámara web.

La interfaz incluye botones los cuales añaden funcionalidades prácticas al sistema. Por un lado, el botón de captura instantánea permite tomar fotografías en el momento preciso, brindando la capacidad de almacenar imágenes específicas para futuros análisis o referencias. Por otro lado, el botón de segmentación facilita la ejecución de la tarea de segmentación en la imagen recién capturada, proporcionando resultados detallados y específicos sobre los elementos electrónicos presentes.

En la Figura 23, se presenta la interfaz de usuario desarrollada para el proyecto.

Figura 23

Interfaz de usuario del proyecto



Nota: La interfaz cuenta con botones que permiten capturar fotos, realizar el procesamiento de imágenes y cerrar la pantalla.

VI RESULTADOS

6.1 Realizar el prototipo de segmentación de componentes electrónicos

Durante el desarrollo del prototipo se lograron resultados favorables. El sistema de visión artificial implementado exhibió una notable capacidad para reconocer, clasificar y delimitar con precisión los diferentes componentes presentes en las tarjetas electrónicas.

La arquitectura del algoritmo de segmentación permitió la identificación de componentes como resistencias, capacitores, circuitos integrados, entre otros. Además, la flexibilidad del prototipo se evidenció al adaptarse eficazmente a diferentes diseños de tarjetas electrónicas.

6.2 Implementar el prototipo utilizando una RaspberryPi y una cámara

La elección de la Raspberry Pi como plataforma de implementación y la integración de una cámara resultaron ser decisiones acertadas. La Raspberry Pi, con su bajo costo y tamaño compacto, proporciona la potencia computacional necesaria para ejecutar el algoritmo de visión artificial de manera eficiente. Por su lado, la cámara captura imágenes de calidad apreciable lo cual facilita el proceso de segmentación y permite una respuesta en tiempo real.

La implementación se realizó teniendo en cuenta aspectos como la optimización de recursos y la fácil integración en entornos existentes. Además, se configuraron interfaces de usuario intuitivas para facilitar la interacción con el sistema, haciendo que el prototipo sea accesible incluso para usuarios no especializados.

En la Figura 24 se aprecia la combinación de la Raspberry Pi con la cámara, así como la operación conjunta con el sistema operativo específico del Raspberry Pi.

Figura 24

Prototipo utilizando Raspberry Pi con una cámara



Nota: Se muestra parte de la codificación realizada para el sistema de segmentación de componentes.

6.3 Realizar pruebas para comprobar el efectivo funcionamiento del proyecto

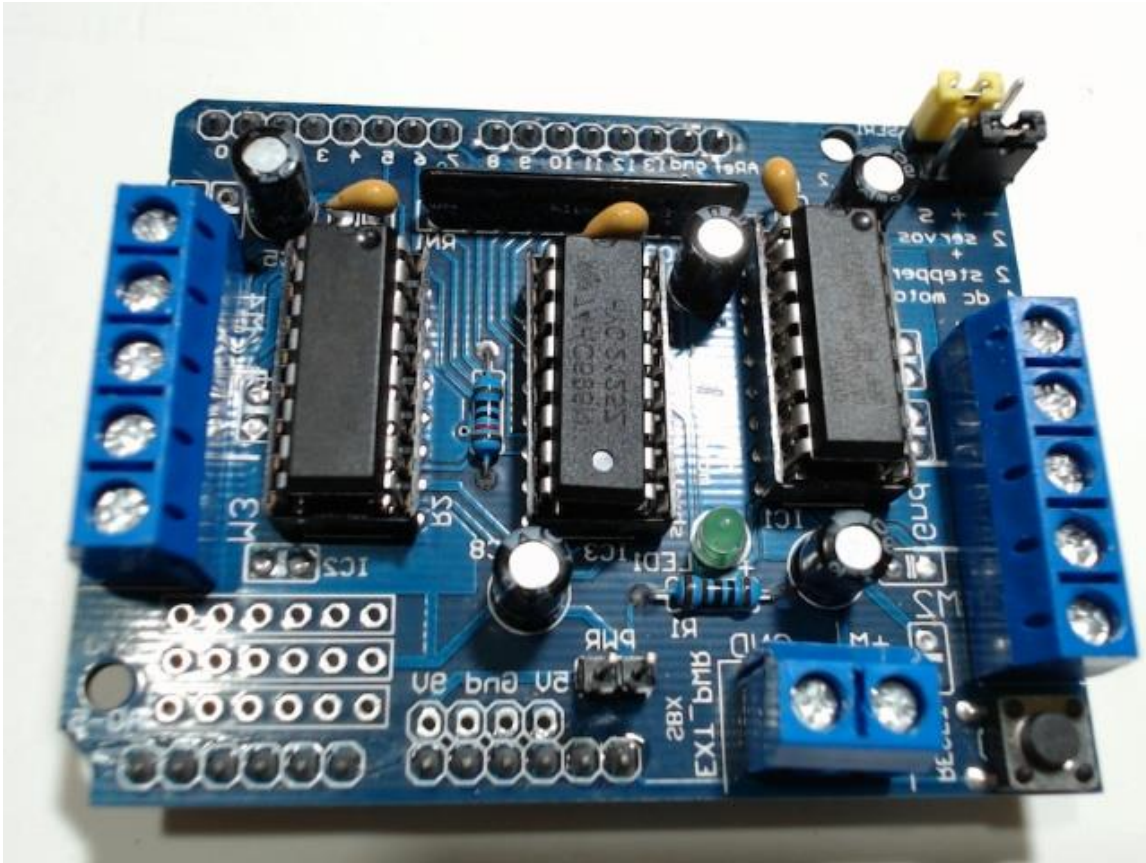
Las pruebas de funcionamiento establecieron una etapa crucial para validar la robustez y confiabilidad del prototipo. Se realizaron diversas pruebas con tarjetas electrónicas de distintos fabricantes, considerando variaciones en el diseño y en la disposición de los componentes. Los resultados obtenidos fueron consistentes, resaltando la habilidad del sistema para ajustarse a situaciones variables y mantener niveles de precisión notables.

Se realiza una muestra del alcance del prototipo desarrollado en la cual la Figura 25 es la imagen original la cual fue sometida al modelo de segmentación y la Figura 26 es la imagen

resultante. Como resultado se obtiene una gran respuesta por parte del modelo, destacando la capacidad que posee para identificar y delimitar con precisión los diferentes componentes electrónicos.

Figura 25

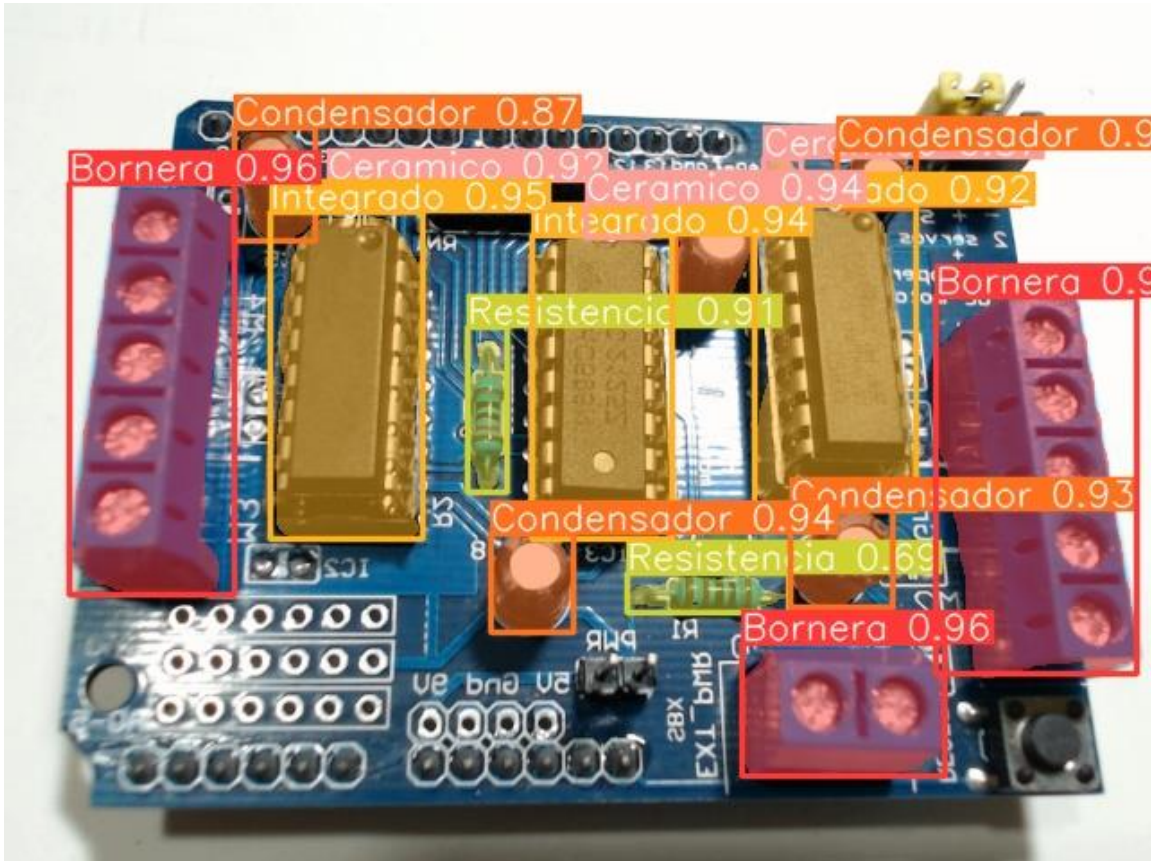
Imagen original



Nota: Imagen de prueba para comprobar eficiencia del modelo.

Figura 26

Imagen resultante



Nota: Segmentación de componentes electrónicos en imagen original.

6.4 Validar el funcionamiento del prototipo a través de un juicio experto

Realizada la programación e implementación del prototipo de segmentación de componentes electrónicos en tarjetas electrónicas, se realizaron pruebas de operatividad con el fin de corroborar el adecuado funcionamiento del sistema. Estas pruebas se las realiza en conjunto con técnicos en reparación de tarjetas electrónicas, dado que su participación resulta fundamental aportando su conocimiento y experiencia, obteniendo una retroalimentación desde varios puntos de vista.

Además, se realiza una pequeña encuesta con el objetivo de adquirir apreciaciones que ayuden a realizar ajustes, garantizando de esta manera que el sistema cumpla con los requisitos esenciales para ser considerado fiable. Los resultados obtenidos de la encuesta indican lo siguiente:

6.4.1 Resultados de encuestas

Pregunta 1. ¿Conoce algún sistema que permita el reconocimiento y segmentación de componentes electrónicos en tarjetas electrónicas?

Tabla 2

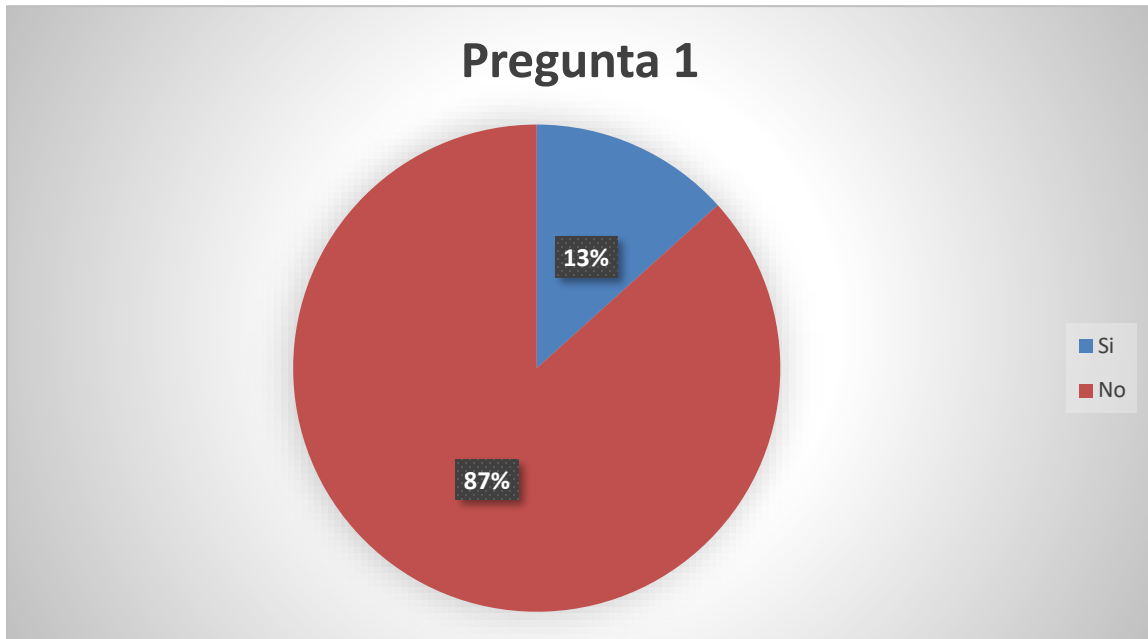
Respuestas correspondientes a Pregunta 1

	Frecuencia	Porcentaje	Porcentaje Total
Sí	2	13.3	13.3
No	13	86.7	100.0
Total	15	100.0	

Nota: Detalle de respuestas de Pregunta 1.

Figura 27

Representación gráfica del resultado de la Pregunta 1



Nota: Exposición gráfica de los resultados provenientes de la Pregunta 1.

Pregunta 2. ¿Considera beneficioso contar con un sistema que pueda reconocer y segmentar componentes electrónicos en una placa de circuito impreso (PCB)?

Tabla 3

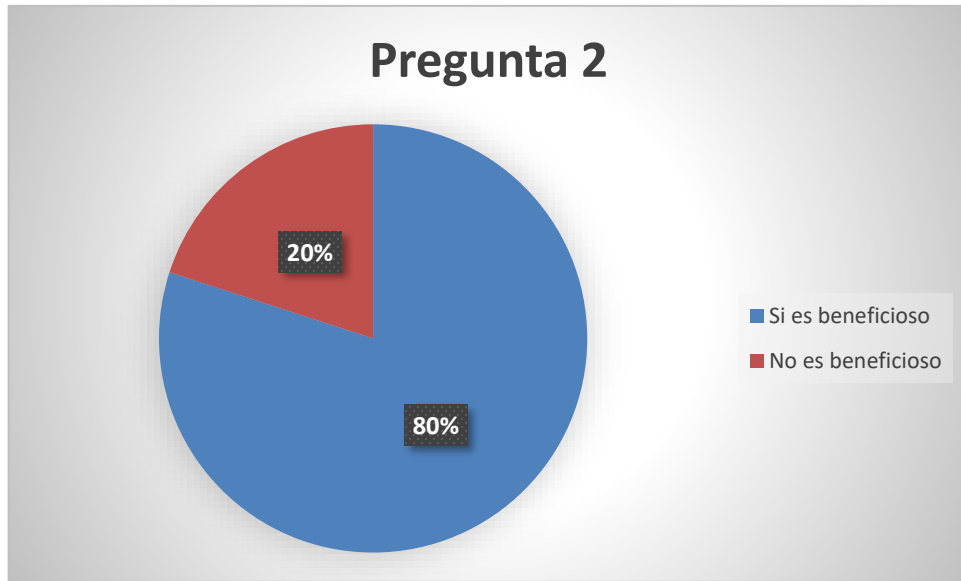
Respuestas correspondientes a Pregunta 2

	Frecuencia	Porcentaje	Porcentaje Total
Sí es beneficioso	12	80.0	80.0
No es beneficioso	3	20.0	100.0
Total	15	100.0	

Nota: Detalle de respuestas de Pregunta 2.

Figura 28

Representación gráfica del resultado de la Pregunta 2



Nota: Exposición gráfica de los resultados provenientes de la Pregunta 2.

Pregunta 3. ¿Qué opción describe mejor la precisión del sistema de segmentación de componentes electrónicos en tarjetas electrónicas demostrado?

Tabla 4

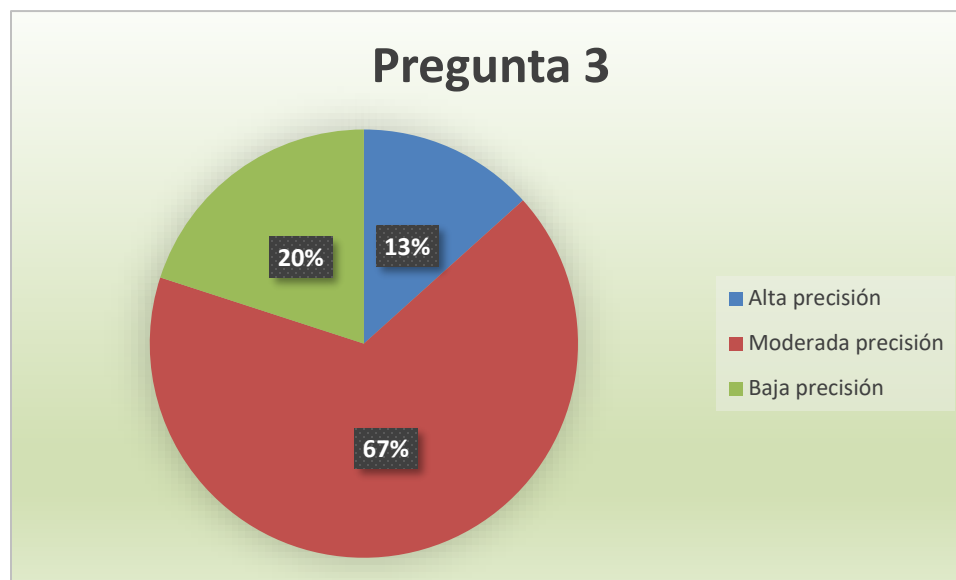
Respuestas correspondientes a Pregunta 3

	Frecuencia	Porcentaje	Porcentaje Total
Alta precisión	2	13.3	13.3
Precisión moderada	10	66.7	80.0
Baja precisión	3	20.0	100.0
Total	15	100.0	

Nota: Detalle de respuestas de Pregunta 3.

Figura 29

Representación gráfica del resultado de la Pregunta 3



Nota: Exposición gráfica de los resultados provenientes de la Pregunta 3.

Pregunta 4. ¿Cómo se compara la velocidad de procesamiento del sistema con respecto a otras soluciones disponibles en el mercado?

Tabla 5

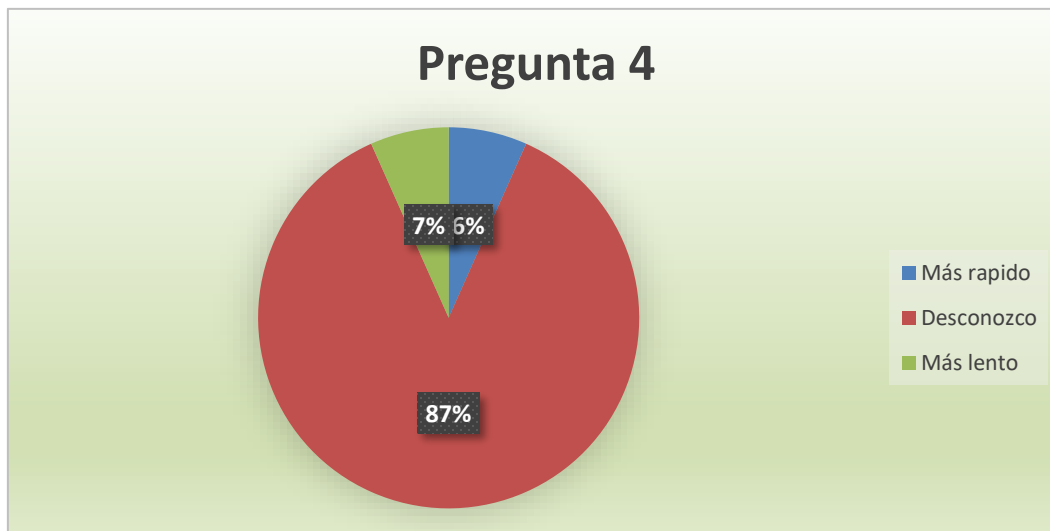
Respuestas correspondientes a Pregunta 4

	Frecuencia	Porcentaje	Porcentaje Total
Más rápido	1	6.7	6.7
Desconozco	13	86.6	93.3
Más lento	1	6.7	100.0
Total	15	100.0	

Nota: Detalle de respuestas de Pregunta 4.

Figura 30

Representación gráfica del resultado de la Pregunta 4



Nota: Exposición gráfica de los resultados provenientes de la Pregunta 4.

Pregunta 5. ¿Cómo afecta la complejidad del diseño de la tarjeta electrónica a la eficacia del sistema de segmentación?

Tabla 6

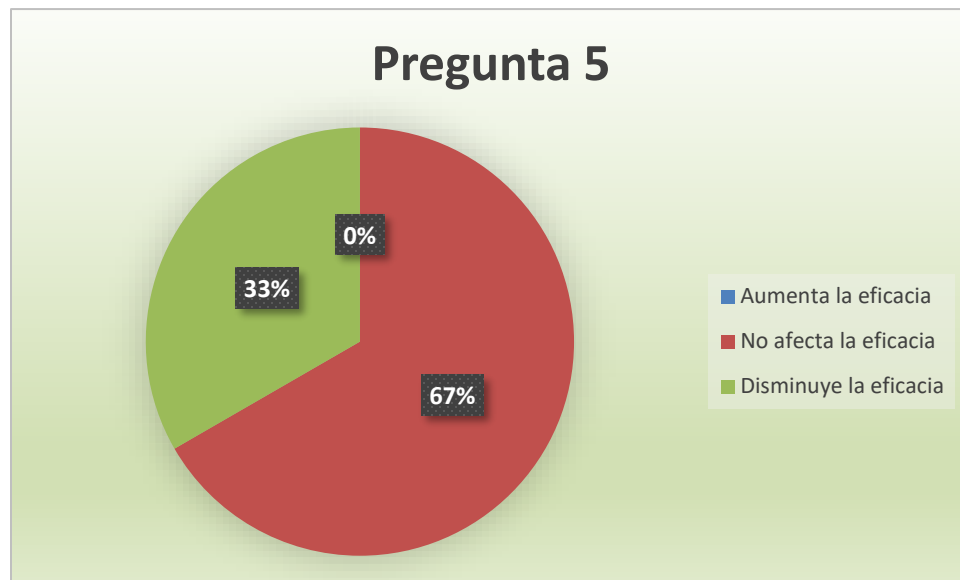
Respuestas correspondientes a Pregunta 5

	Frecuencia	Porcentaje	Porcentaje Total
Aumenta la eficacia	0	0.0	0.0
No afecta la eficacia	10	66.7	66.7
Disminuye la eficacia	5	33.3	100.0
Total	15	100.0	

Nota: Detalle de respuestas de Pregunta 5.

Figura 31

Representación gráfica del resultado de la Pregunta 5



Nota: Exposición gráfica de los resultados provenientes de la Pregunta 5.

Pregunta 6. Observado el funcionamiento del sistema desarrollado, ¿Qué opción representa mejor la recomendación que daría a otros expertos sobre el sistema?

Tabla 7

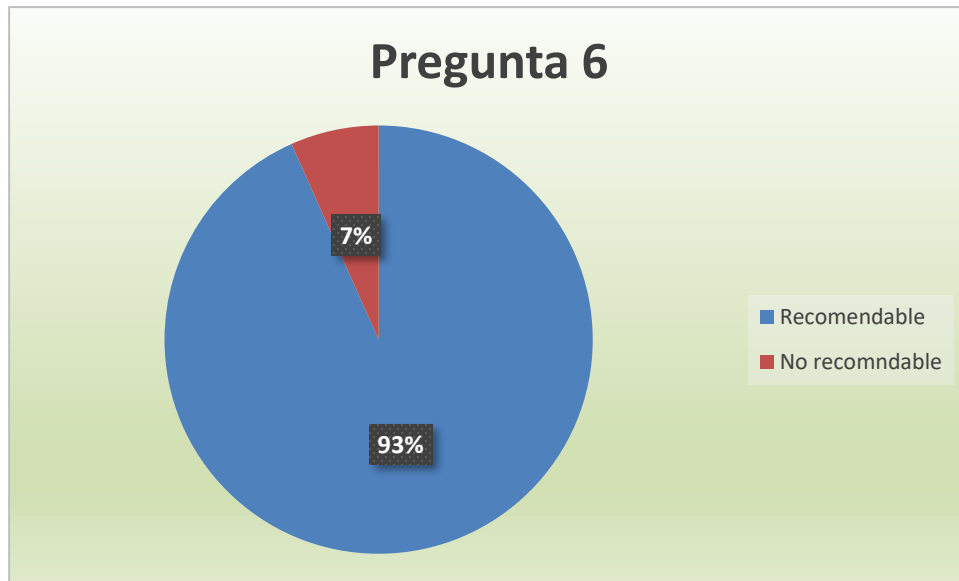
Respuestas correspondientes a Pregunta 6

	Frecuencia	Porcentaje	Porcentaje Total
Recomendable	14	93.3	93.3
No recomendable	1	6.7	100.0
Total	15	100.0	

Nota: Detalle de respuestas de Pregunta 6.

Figura 32

Representación gráfica del resultado de la Pregunta 6



Nota: Exposición gráfica de los resultados provenientes de la Pregunta 6.

6.4.2 Análisis de resultados

La validación del prototipo mediante un juicio experto fortaleció la confianza en la calidad y eficacia del sistema. Los técnicos dedicados a la reparación de tarjetas electrónicas evaluaron el rendimiento del prototipo en detalle, confirmando su capacidad para abordar desafíos específicos de la industria. La retroalimentación positiva destacó la relevancia del prototipo en términos de velocidad y aplicabilidad práctica.

La validación no solo respaldó los resultados de las pruebas, sino que también proporcionó comentarios valiosos para posibles mejoras y ajustes futuros. La aceptación positiva de técnicos en el campo respalda la viabilidad del prototipo como un recurso significativo para el sector industrial de la electrónica y la inspección de tarjetas electrónicas.

VII CRONOGRAMA

En la Tabla 8 se presenta el plan de actividades que se llevara a cabo a lo largo del Trabajo de Titulación.

Tabla 8.

Cronograma de actividades correspondiente al Proyecto de Titulación.

ACTIVIDADES	octubre				noviembre				diciembre				enero	
	1	2	3	4	1	2	3	4	1	2	3	4	1	
Búsqueda de información	■	■	■											
Modelamiento de sistema para captura de imágenes			■	■	■									
Modelamiento de sistema para segmentación de objetos				■	■									
Diseño de interfaz de visualización					■	■								
Pruebas de funcionamiento								■	■	■				
Desarrollo del documento de tesis										■	■	■	■	

VIII PRESUPUESTO

La totalidad del gasto del Trabajo de Titulación se encuentran desglosada en la Tabla 9 donde se proporciona un desglose de cada material a utilizar junto con su precio respectivo en el mercado.

Tabla 9.

Presupuesto estimado para el Proyecto de Titulación.

Tipo	Descripción	Cantidad	P. Unitario	P. Final
Materiales	Raspberry Pi	1	\$128.00	\$128.00
	Cámara web	1	\$96.00	\$96.00
	Case Raspberry Pi	1	\$23.00	\$23.00
Gastos varios	Tarjetas de control Split	2	\$80.00	\$160.00
	Horas de ingeniería	300	\$1.79	\$537.00
			TOTAL	\$944.00

IX CONCLUSIONES

Con base en los resultados alcanzados al concluir el proyecto, es evidente que se ha cumplido con el primer objetivo, el cual consistía en el desarrollo de un prototipo para la segmentación de componentes electrónicos en tarjetas electrónicas mediante el uso de visión artificial.

El segundo objetivo también se ha alcanzado de manera exitosa, ya que se implementó el prototipo utilizando una cámara para la captura de muestras y el módulo Raspberry Pi, encargado de procesar la imagen en conjunto con la programación desarrollada para obtener la segmentación de componentes correspondiente.

En relación al tercer objetivo, los resultados obtenidos, tal como se muestra en la Figura 26, son convincentes. Tras someter la imagen de muestra al sistema, se demostró la confiabilidad del mismo al obtener resultados muy buenos.

Con respecto al cuarto objetivo, la retroalimentación de los técnicos fue mayormente positiva en cuanto al funcionamiento del sistema, el cual demostró su eficacia al ser empleado en la segmentación de componentes electrónicos.

En resumen, los resultados obtenidos destacan la capacidad de la plataforma propuesta para mejorar la eficiencia en la inspección de componentes electrónicos. Esto, a su vez, contribuye de manera significativa a la reducción de tiempos en los procesos de revisión y reparación de tarjetas electrónicas, consolidando así el impacto positivo de este proyecto en el campo de la tecnología y la automatización.

X RECOMENDACIONES

El dispositivo de captura de imágenes debe fijarse en una superficie nivelada, además de ubicarlo en un ambiente con buena iluminación lo cual representará una mayor calidad de imagen a tomar.

El Raspberry Pi debe conectarse a una fuente de poder de 5.1V/5A, preferiblemente utilizar el adaptador original, para el correcto funcionamiento del módulo, utilizar fuentes de alimentación con otras características podría ocasionar reinicios repentinos o mal funcionamiento del sistema.

Instalar las librerías requeridas con su versión mínima para garantizar el adecuado rendimiento del sistema, puesto que si no se instala la librería o se descarga una versión desactualizada podría causar conflictos o errores al momento de ejecutar el programa.

XI REFERENCIAS

- Admin. (4 de Agosto de 2023). *Redes neuronales convolucionales: aplicaciones en procesamiento de imágenes*. Obtenido de Data Universe: <https://data-universe.org/redes-neuronales-convolucionales-aplicaciones-en-procesamiento-de-imagenes/>
- AlexFocus. (31 de Enero de 2023). *Detección de objetos en imágenes con YOLO V8 y Python*. Obtenido de YouTube: https://www.youtube.com/watch?v=fKv4YlyH_VQ
- Amaya, S., Pulgarín, D., & Torres, I. (30 de Marzo de 2016). *Desarrollo e implementación de un sistema de visión artificial basado en lenguajes de uso libre para un sistema seleccionador de productos en un Centro Integrado de Manufactura*. Obtenido de Universidad Católica Luis Amigó: <https://revistas.ucatolicaluissamigo.edu.co//index.php/lampsakos/article/view/1702>
- Amazon Web Services. (s.f.). *¿Qué es una red neuronal? - Explicación de las redes neuronales artificiales*. Obtenido de Amazon Web Services: <https://aws.amazon.com/es/what-is/neural-network/#:~:text=Una%20red%20neuronal%20es%20un,lo%20hace%20el%20cerebro%20humano.>
- Aplicaciones AI. (31 de Diciembre de 2023). *Roboflow - Reseña*. Obtenido de Aplicaciones AI: <https://aplicaciones.ai/roboflow/>
- Bcnvisión. (12 de Septiembre de 2017). *Componentes de la visión artificial*. Obtenido de <https://bcnvision.es/blog-vision-artificial/componentes-vision-artificial/>

Bonilla, C. (Junio de 2020). *Redes Convolucionales*. Obtenido de Depósito de Investigación Universidad de Sevilla (idUS):
<https://idus.us.es/bitstream/handle/11441/115221/TFG%20DGMyE%20Bonilla%20Carri%20Carmelo.pdf?sequence=1&isAllowed=y>

de Frías, C. J. (2018). *Visión artificial aplicada en la identificación de objetos y su parametrización geométrica*. Obtenido de CORE:
<https://core.ac.uk/download/pdf/288502112.pdf>

IBM. (s.f.). *¿Qué son las redes neuronales convolucionales?* Obtenido de IBM:
<https://www.ibm.com/es-es/topics/convolutional-neural-networks>

IBM. (s.f.). *¿Que son las redes neuronales?* Obtenido de IBM: <https://www.ibm.com/es-es/topics/neural-networks>

INFAIMON. (17 de Enero de 2023). *Cámaras*. Obtenido de INFAIMON:
<https://infaimon.com/camaras-vision-industrial/>

Porto, J., & Merino, M. (13 de Octubre de 2022). *Cámara web - Qué es, usos, definición y concepto*. Obtenido de Definición.de: <https://definicion.de/camara-web/>

Raspberry Pi. (26 de Abril de 2022). *¿Qué es Raspberry Pi?* Obtenido de Raspberry Pi:
<https://raspberrypi.cl/que-es-raspberry/>

Raspberry Pi. (s.f.). *Raspberry Pi 5*. Obtenido de Raspberry Pi:
<https://www.raspberrypi.com/products/raspberry-pi-5/>

Roboflow. (s.f.). *Roboflow Docs*. Obtenido de Roboflow : <https://docs.roboflow.com/>

Russell, S., & Norving, P. (2008). *Inteligencia Artificial: Un enfoque moderno*. Obtenido de WordPress: <https://luismejias21.files.wordpress.com/2017/09/inteligencia-artificial-un-enfoque-moderno-stuart-j-russell.pdf>

Tech AMP. (7 de Diciembre de 2021). *El mejor modelo de visión por computadora para tu proyecto*. Obtenido de Youtube: <https://www.youtube.com/watch?v=5MYpEtt0vsk>

Ultralytics. (s.f.). *Ultralytics YOLO v8*. Obtenido de Ultralytics: <https://docs.ultralytics.com>

UNIR. (16 de Noviembre de 2021). *La visión artificial: una revolución en la industria 4.0*. Obtenido de UNIR: <https://www.unir.net/ingenieria/revista/vision-artificial/>

XII ANEXOS

Código para el entramiento del modelo con dataset personalizado

Programa 1

Miguel Navarrete - Jonathan Yanse

Entrenar modelo con el dataset personalizado

from ultralytics **import** YOLO

Ruta del archivo .yaml

DATA = 'C:/Users/jogun/OneDrive/Escritorio/Project_PCB/Tesis_Segmetation/data.yaml'

Carga de pesos de modelo pre-entrenado

model = YOLO('yolov8n-seg.pt')

Entrenamiento de modelo con imagenes personalizadas (componentes electrónicos)

model.train(data=DATA, epochs=25, imgsz=640)

Código para realizar prueba de funcionamiento con una imagen

```
# Programa 2

# Miguel Navarrete - Jonathan Yanse

# Funcionamiento de modelo con imagen de prueba

from PIL import Image

from ultralytics import YOLO

# Ruta donde se aloja pesos de modelo personalizado pre-entrenado

data='runs_25_epochs/segment/train/weights/best.pt'

# Carga de pesos de modelo personalizado pre-entrenado

model = YOLO(data)

# Variable que almacena los resultados de la predicción en la imagen de entrada

results = model.predict('Fotos/IMG4.jpg', save = False, imgsz = 640, conf = 0.5)

for r in results:

    # Graficar predicción de matriz numpy BGR

    im_array = r.plot()

    # Convertir imagen a RGB

    im = Image.fromarray(im_array[... ::-1])

    # Mostrar imagen

    im.show()

    # Guardar imagen - Especificar nombre y formato

    im.save('Imagen.jpg')
```

Código de interfaz gráfica de usuario

```
# Programa 3

# Miguel Navarrete - Jonathan Yanse

# Desarrollo de Interfaz de Usuario

# Librerías necesarias

from PIL import Image, ImageTk

from ultralytics import YOLO

from tkinter import ttk

import tkinter as tk

import cv2

import os

class TesisApp:

    def __init__(self, ventana):

        self.image = None

        self.ventana = ventana

        self.ventana.title("Proyecto de Titulación")

        # Configurar tamaño de ventana

        self.ventana.geometry("800x600")
```

```
# Logotipo de Universidad Politécnica Salesiana
```

```
self.image = tk.PhotoImage(file="UPS.png")
```

```
logo_label = tk.Label(ventana,  
                       image=self.image,  
                       compound = "center")
```

```
logo_label.grid(row=4,  
                column=1,  
                columnspan=2,  
                padx=10,  
                pady=10)
```

```
# Tema de Trabajo de Titulación (ETIQUETA)
```

```
author_label = tk.Label(ventana,  
                        compound="center",  
                        text="DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE "  
                            "\nSEGMENTACIÓN DE COMPONENTES ELECTRÓNICOS "  
                            "\nEN TARJETAS ELECTRÓNICAS MEDIANTE VISIÓN  
ARTIFICIAL",  
                        font=("Times", 12))
```

```
author_label.grid(row=0, column=1, columnspan=2, padx=10, pady=10)
```

```
# Nombre de Autores (ETIQUETA)
```

```
author_label = tk.Label(ventana,  
  
    compound="center",  
  
    text="Autores: "  
  
        "\nMiguel Gustavo Navarrete Salazar "  
  
        "\nJonathan Fernando Yanse Espinoza",  
  
    font=("Times", 12))
```

```
author_label.grid(row=1,  
  
    column=1,  
  
    columnspan=2,  
  
    padx=10,  
  
    pady=10)
```

```
# Muestra de video en tiempo real
```

```
self.canvas = tk.Canvas(ventana,  
  
    width=480,  
  
    height=320)
```

```
self.canvas.grid(row=2,  
  
    column=1,  
  
    padx=10,  
  
    pady=10)
```

```

### Botones de Procesos ###

# Botón Tomar Foto - Enlazado a función "capture_image"

self.btn_capture = ttk.Button(ventana,
                               text="Tomar Foto",
                               command=self.capture_image)

self.btn_capture.grid(row=3,
                      column=0,
                      padx=10,
                      pady=10,
                      sticky="W")

# Botón Procesar Imagen - Enlazado a función "process_image"

self.btn_process = ttk.Button(ventana,
                               text="Procesar Imagen",
                               command=self.process_image)

self.btn_process.grid(row=3,
                      column=1,
                      colspan=2,
                      padx=10,
                      pady=10)

# Botón Cerrar - Enlazado a función "ventana.destroy"

self.btn_close = ttk.Button(ventana,

```

```

        text="Cerrar",

        command=ventana.destroy)

self.btn_close.grid(row=3,

                    column=3,

                    padx=10,

                    pady=10,

                    sticky="E")

# Inicializar la captura de video

# 0 - Corresponde a cámara externa

self.camara = cv2.VideoCapture(0)

self.update_video()

# Actualizacion de video

def update_video(self):

    ret, frame = self.camara.read()

    if ret:

        # Conversion de BGR a RGB

        frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)

        # Convierte arreglo numpy a objeto de imagen

        image = Image.fromarray(frame)

        # Convierte el objeto de imagen a objeto "PhotoImage"

        photo = ImageTk.PhotoImage(image=image)

```



```

self.canvas.create_image(0, 0, anchor=tk.NW, image=photo)

self.canvas.photo = photo

self.ventana.after(10, self.update_video)

# Función para capturar y almacenar imagen
def capture_image(self):
    ret, frame = self.camara.read()

    if ret:
        cv2.imwrite("captura.png", cv2.cvtColor(frame, cv2.COLOR_BGR2RGB))

# Función para segmentar imagen capturada
def process_image(self):
    # Ruta donde se aloja pesos de modelo personalizado pre-entrenado
    data = 'runs_25_epochs/segment/train/weights/best.pt'

    # Carga de pesos de modelo personalizado pre-entrenado
    model = YOLO(data)

    # Variable que almacena los resultados de la predicción
    results = model.predict('captura.png', save=False, imgsz=640, conf=0.5)

    for r in results:
        # Graficar matriz numpy BGR de predicciones

```

```
im_array = r.plot()

# RGB PIL image

im = Image.fromarray(im_array[... ::-1])

# Mostrar imagen

im.show()

# Guardar imagen

im.save('Imagen_Seg.jpg')

pass

if __name__ == "__main__":

    ventana = tk.Tk()

    app = TesisApp(ventana)

    ventana.mainloop()
```