



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

Desarrollo de un prototipo que brinde accesibilidad a imágenes de circuitos eléctricos para personas con discapacidad visual mediante reconocimiento de la posición dactilar

Mena Huera, Juan Enrique

Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Ingeniería en Electrónica y Telecomunicaciones

Trabajo de titulación previo a la obtención del título de Ingeniero en Electrónica y Telecomunicaciones

Ing. Larco Bravo, Julio César.

01 de Febrero del 2022



Tesis Final Juan Mena2022.pdf

Scanned on: 12:55 February 2, 2022 UTC



Overall Similarity Score



Results Found



Total Words in Text



Financiado y patrocinado por:
**JULIO CESAR
LARCO BRAVO**

Identical Words	462
Words with Minor Changes	117
Paraphrased Words	245
Ommited Words	0



Website | Education | Businesses



DEPARTAMENTO DE ELÉCTRICA, ELECTRÓNICA Y TELECOMUNICACIONES

CARRERA DE INGENIERÍA EN ELECTRÓNICA Y TELECOMUNICACIONES

CERTIFICACIÓN

Certifico que el trabajo de titulación, **“Desarrollo de un prototipo que brinde accesibilidad a imágenes de circuitos eléctricos para personas con discapacidad visual mediante reconocimiento de la posición dactilar”** fue realizado por el señor **Mena Huera, Juan Enrique**, el cual ha sido revisado y analizado en su totalidad por la herramienta de verificación de similitud de contenido; por lo tanto cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, razón por la cual me permito acreditar y autorizar para que lo sustente públicamente.

Sangolquí, 1 de febrero del 2022

Firma:



Firmado electrónicamente por:
**JULIO CESAR
LARCO BRAVO**

Ing. Larco Bravo, Julio César.

C. C. 1710638808



DEPARTAMENTO DE ELÉCTRICA, ELECTRÓNICA Y TELECOMUNICACIONES

CARRERA DE INGENIERÍA EN ELECTRÓNICA Y TELECOMUNICACIONES

RESPONSABILIDAD DE AUTORÍA

Yo, Mena Huera, Juan Enrique, con cédula de ciudadanía n°1724152457, declaro que el contenido, ideas y criterios del trabajo de titulación: **"Desarrollo de un prototipo que brinde accesibilidad a imágenes de circuitos eléctricos para personas con discapacidad visual mediante reconocimiento de la posición dactilar"** es de mi autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Sangoquí, 1 de febrero del 2022

Mena Huera, Juan Enrique

C.C.:1724152457



DEPARTAMENTO DE ELÉCTRICA, ELECTRÓNICA Y TELECOMUNICACIONES

CARRERA DE INGENIERÍA EN ELECTRÓNICA Y TELECOMUNICACIONES

AUTORIZACIÓN DE PUBLICACIÓN

Yo, Mena Huera, Juan Enrique, con cédula de ciudadanía n°1724152457, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación:

"Desarrollo de un prototipo que brinde accesibilidad a imágenes de circuitos eléctricos para personas con discapacidad visual mediante reconocimiento de la posición dactilar" en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

Sangolquí, 1 de febrero del 2022

Mena Huera, Juan Enrique

C.C.:1724152457

Dedicatoria

Dedico este logro a Dios el cual me guía a seguir siendo una mejor persona y a sobrellevar algunos obstáculos en la vida que me ha permitido madurar y afrontar de una mejor manera todo lo que me ha pasado hasta la actualidad, siempre toca tener fe en él porque siempre nos va a proteger de todo mal.

Dedico a mis padres, Juan Eduardo y Martha Cecilia, que han sido mi impulso para que día a día pueda superarme y lograr a cumplir los objetivos que eh logrado hasta ahora y que lograre en un futuro, gracias por ayudarme tanto a mejorar día a día como persona, ah entender que la vida no es fácil y que toca lucharla todos los días para que nunca que deje derrotado, gracias por sus conejos, por su apoyo condicional y por todo lo que eh vivido con ustedes no lo cambio por nada en el mundo, los amo con todo mi corazón.

Dedico a mi hermana que a pesar de que no nos comprendemos en ciertas situaciones, sabemos que vamos a apoyarnos cuando en verdad nos necesitemos, porque ese es el amor de hermanos no va a ser perfecta, pero si va a ser única y espero que sigamos cosechando muchos frutos juntos como familia y que cada quien pueda llegar a tener todo lo que de niño soñamos, gracias por estar a mi lado apoyando mis locuras, te quiero hermanita linda.

A mis mascotas, aunque algunos piensen que no importan, pero fueron los que me acompañaron en todas las desveladas hasta el último momento que me vaya a dormir, fueron mi fiel compañía.

A mis amigos que me dio la vida universitaria, porque sin ellos no hubiera sido lo mismo todos los desvelos, malas noches buenas notas, malas notas y todo lo que festejamos cuando alguien le iba bien mal o bien, gracias por las anécdotas y las que tendremos cuando seamos profesional. Quisiera agradecer a un sin número de personas, pero me quedaría cortos de palabras, solo diré gracias a cada una de las personas que me permitieron convertirte en un profesional del Ecuador.

Juan Enrique Mena Huera

Agradecimiento

Me va ser difícil agradecer a pocas personas, porque en realidad todas las personas que me rodean día a día siempre me apoyaron de alguna forma a alcanzar el tan anhelado logro de ser ingeniero.

Sin embargo, no podía dejar afuera a mis padres que fueron pilar fundamental que pudiera culminar mis estudios universitarios, ellos siempre me apoyaron en todo lo que tuvieron a su alcance y fue de inmensa ayuda ya que me dejaron disfrutar estudiar y vivir la vida universitaria, **MUCHAS GRACIAS POR TODO.**

Agradecer a mi hermana que me apoyó en mis momentos más difíciles en la U como los fracasos que se aprende de ellos para poder mejorar y mejorar siempre, siempre serás mi mejor amiga la cual puede confiar ciegamente y nunca me va a decepcionar.

Agradezco al docente Julio Larco, el cual fue un excelente director de tesis, apoyándome en todo momento en situaciones donde no encontraba solución a problemas que se me presentaron en el camino y gracias a ello pude terminar el trabajo con éxito, a todos los docentes que aportaron su granito de arena con su enorme conocimiento para que pueda estar preparado para todas las ramas de mi carrera, muchas gracias por su información brindada va a ser de gran ayuda en el futuro. Gracias a todas las personas no videntes que ayudaron en la elaboración de la etapa final del proyecto, gracias por darme un momento de su apreciado tiempo.

Juan Enrique Mena Huera

Índice de Contenidos

Resumen	15
Abstract	16
Capítulo I	17
Generalidades	17
Introducción.....	17
Justificación e importancia	18
Estado del arte	20
Alcance	22
Objetivos	24
Objetivo general.....	24
Objetivos específicos	24
Capítulo II	26
Marco teórico	26
Discapacidad visual.....	26
Discapacidad visual en el Ecuador	26
Tipos de discapacidad visual	29
Circuitos eléctricos	31
Generalidades	31
Tipos de circuitos eléctricos	32
Procesamiento digital de imágenes.....	34

Generalidades	34
Pasos para el procesamiento digital de imágenes	36
Tipos de extracción de características de imágenes digitales	37
Redes neuronales	40
Generalidades	40
Redes neuronales artificiales	41
Red neuronal profunda	46
Red neuronal convolucional.....	48
Síntesis de habla.....	50
Definición.....	50
TTS.....	51
Detección de la mano.....	54
Características.....	54
Detección de los dedos de la mano	56
Tarjeta NVIDIA Jetson Nano	58
Definición.....	58
Características.....	58
Software	59
Python	59
OpenCV.....	60
Mediapipe	60

	10
gTTS.....	60
Visual Studio Code	60
Capítulo III	62
Metodología de desarrollo del proyecto	62
Problema y necesidades	62
Diagrama general del sistema	64
Captura.....	65
Preprocesamiento.....	66
Detección de la mano (Landmarking)	71
Detección del elemento (circuito)	74
Visualización del objeto en pantalla	76
Reproducción del nombre del objeto.....	77
Capítulo IV	80
Implementación y resultados	80
Introducción.....	80
Prueba 1: Evaluación del funcionamiento del sistema.....	83
Resultados de evaluación de funcionamiento	86
Prueba 2: Valoración del rendimiento.....	92
Resultados de valoración del rendimiento.....	92
Capítulo V	96
Conclusiones y recomendaciones.....	96

Conclusiones.....	96
Recomendaciones.....	98
Trabajos futuros	99
Bibliografía.....	100
Anexos.....	108

Índice de tablas

Tabla 1	Clasificación según distante y cercana de presentación.....	30
Tabla 2	Elementos básicos de un circuito eléctrico	31
Tabla 3	Métodos de extracción de puntos.....	37
Tabla 4	Métodos de extracción de líneas.....	38
Tabla 5	Método de extracción de círculos	39
Tabla 6	Características técnicas de la Cámara Web Logitech HD C270	66
Tabla 7	Características tarjeta JETSON NANO	67
Tabla 8	Elementos del circuito reconocidos por el sistema	75
Tabla 9	Identificación de usuarios.....	81
Tabla 10	Encuesta de evaluación de resultados del sistema	87
Tabla 11	Evaluación de resultados del sistema	94

Índice de figuras

Figura 1 Población con discapacidad visual según educación y laboralmente activas	27
Figura 2 Población según grado de discapacidad	28
Figura 3 Población según grupo etario	29
Figura 4 Circuito eléctrico en serie	33
Figura 5 Circuito eléctrico en paralelo	33
Figura 6 Circuito eléctrico mixto	34
Figura 7 Comparativa neurona cerebral y RNA	42
Figura 8 Principios de funcionamiento de RNA	43
Figura 9 Tipos de aprendizaje RNA	45
Figura 10 Arquitectura RNN	47
Figura 11 Ejemplo de resultado de procesamiento kernel en CNN	49
Figura 12 <i>Arquitectura básica CNN</i>	49
Figura 13 Arquitectura de la tecnología de síntesis de voz (TTS).....	51
Figura 14 Proceso de síntesis de voz (TTS).....	53
Figura 15 <i>Etapas del reconocimiento gestual</i>	55
Figura 16 Ubicación de puntos de referencia en las manos (landmarks).....	57
Figura 17 Vista frontal y trasera de la NVIDIA Jetson Nano	59
Figura 18 Diagrama de bloques del sistema	64
Figura 19 Soporte del prototipo	65
Figura 20 <i>Cámara</i>	66
Figura 21 Tarjeta Jetson Nano	67
Figura 22 <i>Jumper para fuente de alimentación</i>	68
Figura 23 Escritorio JETSON NANO	68

Figura 24 <i>Detección de los puntos de la mano</i>	71
Figura 25 <i>Visualización en pantalla de la mano e identificación del dedo cursor</i>	73
Figura 26 <i>Circuito ejemplo para el reconocimiento del sistema</i>	74
Figura 27 <i>Ejemplo para el reconocimiento de coordenadas en el sistema</i>	75
Figura 28 <i>Visualización en pantalla de un circuito</i>	77
Figura 29 <i>Posicionamiento del usuario en el prototipo</i>	82
Figura 30 <i>Indicaciones de uso del dispositivo</i>	82
Figura 31 <i>Visualización y reconocimiento en circuito 1</i>	83
Figura 32 <i>Visualización y reconocimiento en circuito 2</i>	83
Figura 33 <i>Visualización y reconocimiento en circuito 3</i>	84
Figura 34 <i>Visualización y reconocimiento en circuito 4</i>	84
Figura 35 <i>Visualización y reconocimiento en circuito 5</i>	84
Figura 36 <i>Visualización de objetos en pantalla</i>	85
Figura 37 <i>Testeo de usuarios</i>	85
Figura 38 <i>Código QR videos de implementación</i>	86
Figura 39 <i>Registro de sesión de pruebas</i>	94
Figura 40 <i>Registro de sesión de pruebas</i>	95

Resumen

La discapacidad visual es una condición que tiene un importante número de estudiantes del Ecuador; lamentablemente, el sistema educativo no provee de suficientes herramientas y métodos para promover y garantizar el aprendizaje de esta población. Esto es particularmente relevante en materias de electricidad y/o electrónica, puesto que los circuitos eléctricos son elementos básicos, y tienen como base del aprendizaje y estudio la abstracción y representación gráfica de estos en diagramas y simbologías.

El presente trabajo tuvo como objetivo desarrollar un dispositivo de reconocimiento de la posición dactilar, que facilitara el aprendizaje de los distintos elementos que componen los circuitos eléctricos el cual cuenta con tres módulos (procesamiento de objeto eléctrico OpenCV, detección dactilar MediaPipe y síntesis de audio gTTS). Mediante el ingreso de imágenes en tiempo real vía cámara web, el módulo MediaPipe realiza el reconocimiento de los gestos de la mano y OpenCV permite el reconocimiento del componente eléctrico identificado. Una vez se realiza este reconocimiento, el módulo gTTS sintetiza en voz el nombre del componente. Esto se realizó mediante la adaptación del framework MediaPipe para crear un algoritmo adaptado a estas necesidades particulares.

El prototipo fue probado por seis usuarios con distintos grados de discapacidad, quienes obtuvieron buenos resultados en las pruebas de reconocimiento de componentes eléctricos, y valoraron positivamente el dispositivo y su potencial.

Palabras clave:

- **RECONOCIMIENTO DACTILAR**
- **DISCAPACIDAD VISUAL**
- **APRENDIZAJE AUTOMATICO**
- **PROCESAMIENTO DE IMÁGENES**

Abstract

Visual impairment is a condition that a significant number of students in Ecuador have; Unfortunately, the educational system does not provide enough tools and methods to promote and guarantee the learning of this population. This is particularly relevant in matters of electricity and/or electronics, since electrical circuits are basic elements, and their learning and study are based on the abstraction and graphic representation of these in diagrams and symbols.

The objective of this work was to develop a fingerprint position recognition device, which would facilitate the learning of the different elements that make up the electrical circuits, which has three modules (OpenCV electrical object processing, MediaPipe fingerprint detection and gTTS audio synthesis). By entering images in real time via a webcam, the MediaPipe module performs the recognition of hand gestures and OpenCV allows the recognition of the electrical component identified. Once this recognition is done, the gTTS module synthesizes the name of the component in speech. This was done by adapting the MediaPipe framework to create an algorithm tailored to these particular needs.

The prototype was tested by six users with different degrees of disability, who obtained good results in the tests of recognition of electrical components, and positively valued the device and its potential.

Keywords:

- **HAND TRACKING**
- **VISUAL DISABILITY**
- **MACHINE-LEARNING**
- **IMAGE PROCESSING**

Capítulo I

Generalidades

Introducción

Según información obtenida de la Organización Mundial de la Salud (OMS) a nivel mundial se conoce que existen cerca de 1300 millones de personas que poseen algún tipo de deficiencia visual, siendo las primordiales causas de estos los errores de refracción no corregidos y las cataratas (OMS, 2019). En Ecuador, del total de personas registradas que presentan algún tipo de discapacidad, el 11,54 % es de tipo visual, lo que corresponde a más de 54.000 personas con esta condición. Dentro de este grupo, alrededor del 34 % presenta un grado de discapacidad de 30 a 49 %, mientras que el 29,5 % presenta de 75 a 84 %, y más del 9 % presenta más del 85 % de discapacidad.

Desde los 18 años de edad se acrecienta la prevalencia de esta discapacidad: más de 80 % de los discapacitados visuales se encuentran entre los 25 y más años. La mayoría de los estudiantes con discapacidad visual asisten a varios centros de educación especial motivo por el cual quedan apartados del sistema educativo nacional convencional siendo identificados como personas “diferentes”, con esto disminuyendo las posibilidades de integración para tener acceso una educación de calidad (OMS, 2019).

En el mundo, existen una serie de técnicas que se enfocan en la asistencia y desarrollo de personas con discapacidad visual leve o severa, una de estas técnicas es la tiflotecnología que tiene como objetivo utilizar la tecnología y herramientas para mejorar la calidad de vida de las personas invidentes.

Con el empleo de la tecnología *tiflo* (ciego en griego) que su definición es “permitir la adaptación de técnicas, conocimientos, recursos y procedimientos para su

utilización de gente no vidente” (Aldaz, 2016, p. 23). Las personas con discapacidad visual buscan oportunidades para incluirse en la sociedad. Por ello, buscan una educación de calidad en centros educativos especializados que establezcan métodos adaptados a características específicas como el sistema Braille.

El Braille es un sistema de lectoescritura más utilizado en este campo. Con los años, se ha vuelto cada vez menos usado y solo algunos usuarios pueden interpretarlo, pues, consiste en una disposición de seis puntos dispuestos en tres filas y dos columnas, presentando limitaciones en la comprensión y el aprendizaje del texto, así como velocidad lenta en la lectura y no cuenta con signografía especial para matemática, física, química, etc.

Para dar solución a esta problemática surge la idea de utilizar la electrónica enfocada a las personas no videntes debido a que ayudan a diseñar herramientas autónomas con el fin de establecer conexiones a larga distancia, automatización de tareas y facilitar el desarrollo de actividades. El Aprendizaje Automático (*Machine Learning*) utiliza algoritmos para crear operaciones automatizadas según lo programado como el reconocimiento de imágenes, palabras, entre otros, los cuales benefician a las personas invidentes. De esta forma, en el presente trabajo se utiliza el módulo MediaPipe (MediaPipe, 2021), basado en Aprendizaje Automático, que permite el reconocimiento de manos humanas a partir del ingreso de las imágenes en tiempo real con una cámara web, lo que se ha llamado visión por computadora (Ichi, 2021).

Justificación e importancia

De acuerdo con el Consejo Nacional para la Igualdad de Discapacidades (CONADIS) determina que en Ecuador existen 470.820 personas con algún tipo de discapacidad, de este total un 19,75% presenta un nivel leve y severo, incluso el

11,54% presente el tipo de discapacidad visual (CONADIS, 2021). Según la Constitución de Ecuador de 2008, la integración de las personas con discapacidad vive un gran auge en la inclusión social, representando un grupo de atención prioritaria con derecho a la educación y una vivienda digna (Constitución de la República del Ecuador, 2008). El Plan Nacional del Buen Vivir establece que este grupo vulnerable tiene derecho a una educación gratuita de calidad y la oportunidad de conseguir un empleo digno (SENPLADES, 2017).

La accesibilidad académica es un aspecto fundamental para cumplir con los deberes del Estado, y, por tanto, para garantizar el efectivo ejercicio de los derechos fundamentales de las personas. Esta noción implica la supresión de las barreras u obstáculos que no permiten el desarrollo adecuado de aprendizajes y competencias en la población discapacitada, y es imprescindible para generar una real integración de estas personas al ámbito académico (Palma, 2021). En particular, el área de las ingenierías y tecnicaturas en electrónica son áreas que requieren de aprendizajes basados en la identificación de elementos abstractos (símbolos) o teóricos y otros concretos (elementos eléctricos concretos), de modo que es fundamental apoyar a la población con discapacidad visual para adquirir estos aprendizajes considerando las dificultades que el aprendizaje tradicional puede representar para ellos.

Este proyecto es relevante porque ayuda a las personas con discapacidad visual en el reconocimiento de imágenes, siendo útil para el desenvolvimiento e inclusión a nivel educativo, social y laboral. Los circuitos eléctricos constituyen un aprendizaje esencial que deben adquirir los estudiantes y trabajadores del área, por tanto, este trabajo representa una herramienta de soporte para determinar los componentes de un circuito mediante la utilización de un prototipo programado en Python, con diversas aplicaciones potenciales.

Es así que en este proyecto se emplea *Machine Learning* en Python 3.0 para para el reconocimiento de la palma de la persona no vidente, ayudando a identificar mediante la posición dactilar un elemento de una imagen de un circuito electrónico, utilizando una cámara en tiempo real. Una vez que lo detecte, se emite un audio con el nombre del componente eléctrico encontrado, siendo muy importante para mejorar las condiciones de vida de este grupo vulnerable.

Estado del arte

Se conoce que existen estudios relacionados con interfaces para ayudar a personas con discapacidad visual a integrarse al mundo tecnológico, esto con el fin de que aprendan de una manera didáctica y divertida diferentes temas educativos que aporten conocimiento.

El trabajo de Muñoz (2021) desarrolló una aplicación con uso de imágenes de circuitos eléctricos básicos (simbología) que permitiera reconocer los distintos componentes presentes en ellos. Como resultado, el software ofrece documentos de texto con una descripción pormenorizada de los tipos de conexiones y componentes del circuito en particular. Para realizar esto, Muñoz tomó como base el Aprendizaje Automático de redes neuronales, en particular YOLO, y una base de datos de 465 imágenes para obtener aciertos por sobre el 95 %. Este trabajo fue relevante en términos metodológicos para la presente investigación, no obstante, se diferencian en su orientación: como herramienta de accesibilidad para personas con discapacidad visual, el reconocimiento de circuitos y sus componentes se realizó a través de visión por computadora en tiempo real, y para obtener un audio con la información de estos.

El trabajo de Akanksha & Dhole (2018) utilizó el reconocimiento de imágenes para circuitos eléctricos dibujados a mano basado en KNN, con el que puede detectar

imágenes. KNN se utiliza para percibir y diferenciar los caracteres eléctricos de la imagen del circuito eléctrico; proporcionando la imagen del circuito verificado para que la máquina creada pueda usarla. Así, el algoritmo realiza el procesamiento a partir del preprocesamiento del input (circuito dibujado a mano) y la esqueletización de este, lo que se da mediante una conversión del color a escala de grises, la binarización y la eliminación del ruido de la imagen. Luego, la imagen es segmentada en los distintos componentes y la extracción de sus características, para posteriormente pasar al entrenamiento KNN. Como resultado, se obtuvo el reconocimiento de elementos con mucha precisión. El estudio descrito fue de especial interés tanto metodológico como teórico; en él se indicó, además, la importancia de contar con métodos eficientes de reconocimiento de los diagramas de circuitos eléctricos, aunque se diferenció de proyecto desarrollado en este trabajo en el tipo de input y la funcionalidad general del sistema, que busca la accesibilidad para personas con discapacidad visual.

Más enfocado en la accesibilidad para discapacitados visuales, la investigación presentada por Cachimuel (2021) diseñó una aplicación enfocada a personas con pérdida parcial y total de la vista para otorgarles un método de acceso a expresiones matemáticas, utilizando audio tridimensional, la aplicación fue creada en Python y es bastante útil para que las personas puedan aprender funciones y modelos matemáticos a través de un audio en 3D. El trabajo representó un aporte metodológico importante, además de ofrecer funcionalidad para la accesibilidad de personas no videntes. Esto permitió evidenciar la importancia de integrar módulos de síntesis de audio para el aprendizaje de personas no videntes.

Por otro lado, se conoce que el aprendizaje automático (ML) consiste en automatizar la detección de patrones a través del análisis de datos, donde se pueden emplear técnicas a nivel estadístico, algoritmos de optimización y procesamiento de

grandes cantidades de datos. Cuando se incorporan estas técnicas a los sistemas ayudan a reconocer los patrones y predecir comportamientos automáticamente. El aprendizaje automático se deriva de la Inteligencia Artificial (IA), identificando patrones complejos de millones de datos para predecir comportamientos específicos y son capaces de mejorar de manera independiente a través de la retroalimentación.

Estos estudios han constituido un aporte en distintas dimensiones para el desarrollo de este trabajo. Por una parte, evidencian la utilidad de las redes neuronales y Machine Learning para el desarrollo de aplicaciones que presten funcionalidades como visión por computadora, conversión de texto a audio, etc. Por otra parte, aportan una metodología de trabajo que se tuvo en cuenta para desarrollar este trabajo, y permitió optar por Python y los módulos MediaPipe y OpenCV como frameworks y librerías adecuadas para los objetivos de este proyecto.

Alcance

El alcance del presente proyecto es el diseño de un prototipo de detección dactilar para el reconocimiento de los componentes de un circuito eléctrico para personas no videntes. Este dispone de una cámara web en un pedestal a una altura adecuada para enfocar las imágenes y la mano de los usuarios. El dispositivo consta de un software desarrollado en Python3, con lenguaje de código legible y con módulos principales OpenCV, el cual permite reconocer componentes de un circuito electrónico, como resistencias, capacitores, inductores y fuentes de alimentación a partir de los gestos de la mano. También dispone de un módulo de transformación texto-audio (gTTS) para emitir la información correspondiente al componente electrónico a estudiar, para lo que se utilizará como *hardware* una tarjeta Nvidia JETSON NANO.

Para esto, se implementó un algoritmo que permite el reconocimiento de cinco tipos de circuitos con sus respectivos componentes mediante la interacción entre el usuario y la imagen del componente. Para reconocer la palma se utiliza el módulo OpenCV (OpenCV, 2021), el cual reconoce imágenes en colores RGB (Red Green Blue / Rojo Verde Azul) que consta de 3 componentes con valores de 0 a 255; esto permite obtener el color específico de la mano para fácil reconocimiento. El módulo MediaPipe ayuda al reconocimiento de la mano humana a través de la visión por cámara; reconoce la mano y la divide en 21 puntos específicos (entre los que cuenta la identificación del dedo índice como dedo cursor). Luego, se reconoce en el gesto de la mano la simbología que representa, con lo que el prototipo emite un audio con el nombre del componente electrónico identificado basado en el procesamiento del módulo gTTS de síntesis de voz a partir de un texto (Parzibyte, 2019). Las herramientas de desarrollo se relacionan con Python 3.0 debido a su accesibilidad, portabilidad y legibilidad de la utilización del código, siendo un lenguaje de programación multiparadigma (Phyton, 2021).

En el proyecto actual se considera el Aprendizaje Automático en tiempo real, el cual se basa en realizar predicciones en tiempo real, reconocimiento comportamientos como el modelo de detección de la mano que trabaja con la imagen completa de la cámara web en vivo y devuelve un cuadro delimitado de la mano, luego se trabaja con otro modelo que una vez identificada la mano se reconozca los puntos clave de la mano en 3D de alta fidelidad, después, con los puntos de referencia se reduce drásticamente la necesidad de tener gran cantidad de datos, ahorrando la memoria interna del dispositivo electrónico que se esté utilizando.

El prototipo diseñado se testeó a través de su aplicación a 6 usuarios pertenecientes a la población de estudiantes no videntes (con pérdida total o parcial de

la vista) de diversas instituciones de educación del Distrito Metropolitano de Quito, quienes utilizaron el dispositivo y valoraron su efectividad a través del criterio de usuarios mediante la aplicación de encuestas y entrevistas.

Si bien el prototipo diseñado puede prestar utilidad en distintos ámbitos y podrá ser utilizado por personas con discapacidad visual parcial o total, no considera otro tipo de discapacidades, de modo que solo se enfoca en este tipo de discapacidad. Su desarrollo tampoco considera una evaluación económica-financiera, por lo que solo se evaluará su eficacia técnica para resolver la problemática en cuestión.

Objetivos

Objetivo general

Desarrollar un prototipo que brinde accesibilidad a imágenes de circuitos eléctricos para personas con discapacidad visual mediante reconocimiento de la posición dactilar usando procesamiento digital de imágenes mediante librerías de Python 3 ejecutados en una tarjeta Jetson NANO.

Objetivos específicos

- Investigar sobre trabajos previamente realizados enfocados en aplicaciones informáticas que utilicen reconocimiento dactilar, *machine learning*, reconocimiento de imágenes para las personas que tengan discapacidad visual parcial y total.
- Familiarizarse con el funcionamiento de la tarjeta Nvidia JETSON NANO.
- Investigar y estudiar las librerías y módulos del lenguaje Python que permitan el procesamiento de imágenes.
- Implementar un algoritmo que permita identificar la posición dactilar con una

tarjeta Jetson Nano.

- Determinar la posición del dedo índice de la mano sobre una imagen en tiempo real capturada por una cámara.
- Transformar la posición del reconocimiento dactilar a un audio descriptivo.
- Implementar un prototipo que permita identificar la posición dactilar sobre la imagen de un circuito eléctrico.
- Realizar pruebas de funcionamiento del prototipo.
- Evaluar el desempeño del prototipo mediante análisis de los datos adquiridos con la participación de personas con distintos grados de discapacidad visual.

Capítulo II

Marco teórico

Discapacidad visual

Discapacidad visual en el Ecuador

Según Arias, et al., (2018) la discapacidad se refiere a las limitaciones que tiene un individuo para desarrollar una actividad, involucrando factores personales y sociales, pues, presenta restricciones en su participación. Los tipos de discapacidad pueden ser física, auditiva, intelectual, psicosocial y visual.

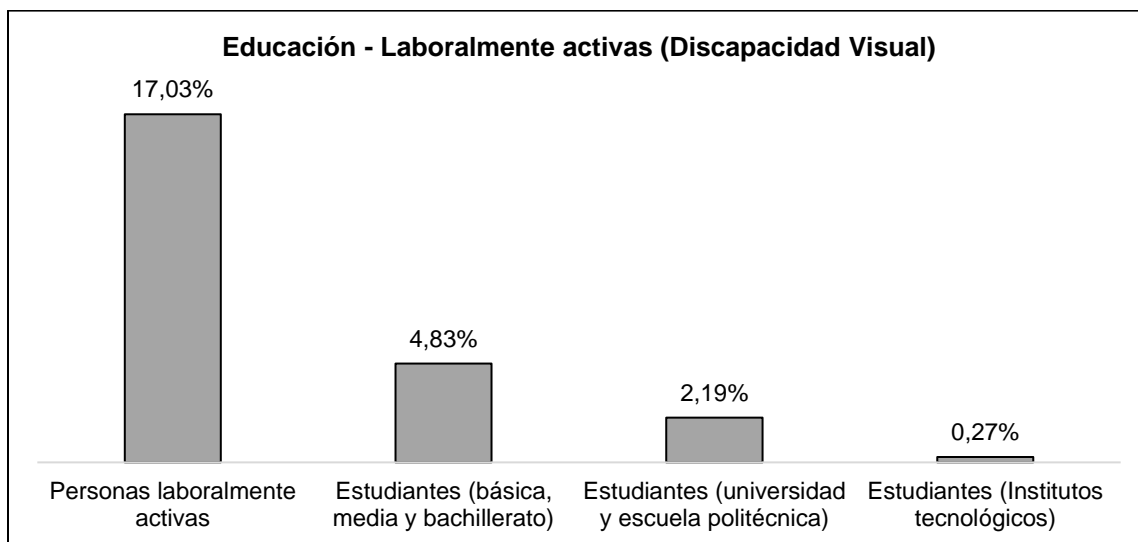
Por su parte, Llamazares, et al., (2017) hace referencia que la discapacidad visual representa una condición que afecta a la forma de percibir las imágenes u objetos ya sea a corta o larga distancia.

En este contexto, en el país existen alrededor de 470.820 personas con discapacidad, de esta cifra un 11,54% corresponde a individuos con discapacidad visual, 14,63% a sujetos laboralmente activas, 10,11% estudiantes de educación Básica, Media y Bachillerato, 1,26% estudiantes de universidad – Escuelas Politécnicas y 0,30% alumnos de Institutos Tecnológicos (CONADIS, 2021).

Existen 54.343 personas con discapacidad visual, de este total hay 9.254 personas que se encuentran laboralmente activas, 2.625 estudiantes de educación Básica, Media y Bachillerato, 1.188 alumnos de universidades y 147 de Institutos Tecnológicos (CONADIS, 2021). En la Figura 1 se presenta las personas con discapacidad correspondientes a estudiantes y están laboralmente activas en Ecuador:

Figura 1

Población con discapacidad visual según educación y laboralmente activas

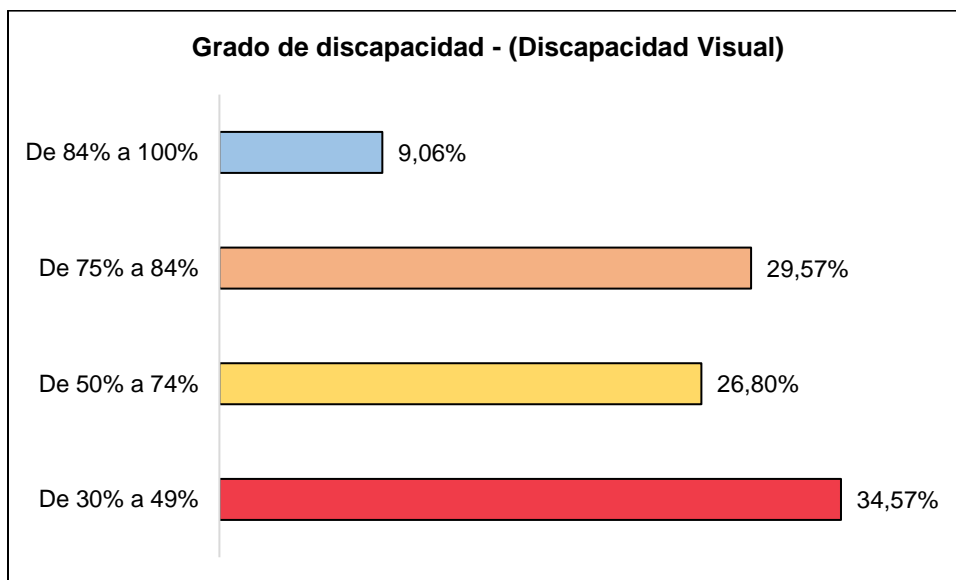


Nota. El gráfico representa a personas con discapacidad visual. Adaptado de *Estadísticas de Discapacidad* (párr.2), por CONADIS, 2021, <https://www.consejodiscapacidades.gob.ec/estadisticas-de-discapacidad/>

Además, se identificó que, según el género, un 41,40% (22.500) son mujeres, el 58,59% (31.841) hombres y un 0,004% (2) otros. De acuerdo al grado de discapacidad de personas no videntes, donde la mayoría tiene un nivel de 30 – 49% (18.785), seguido de sujetos con 75 – 84% (16.068), entre 50 – 74% corresponde a 14.566 individuos y en menor proporción se encuentran las personas que están en un grado de discapacidad entre 84 – 100% (4.924), esto indica que la mayoría está en un rango moderado. La Figura 2 muestra a las personas invidentes de acuerdo al grado o nivel de discapacidad.

Figura 2

Población según grado de discapacidad

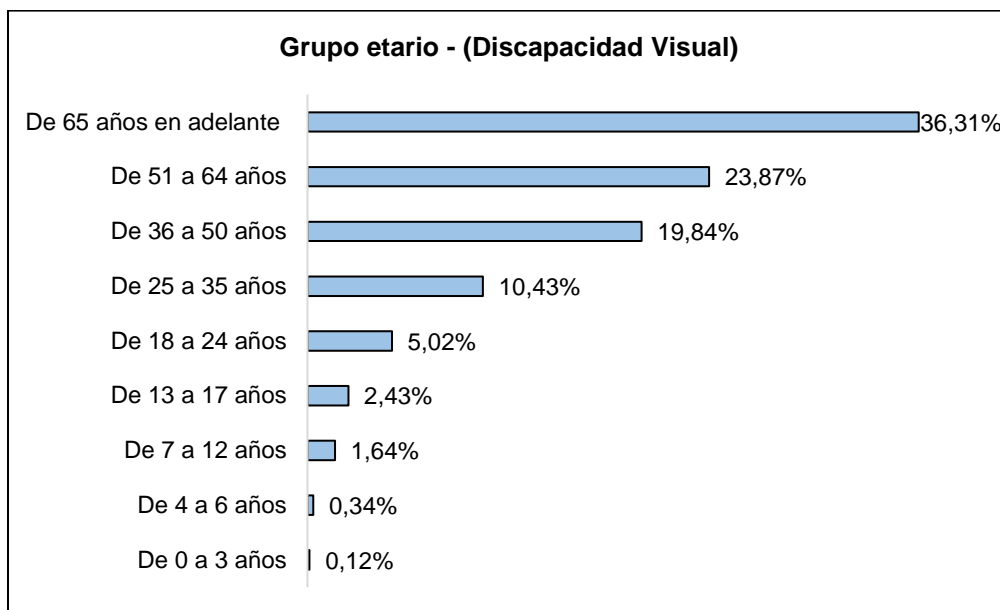


Nota. El gráfico representa a personas con grado de discapacidad. Adaptado de *Estadísticas de Discapacidad* (párr.3), por CONADIS, 2021, <https://www.consejodiscapacidades.gob.ec/estadisticas-de-discapacidad/>

De igual modo, en la Figura 3 se tiene la distribución de las personas con discapacidad visual según edad, donde 19.734 corresponde a la población de más de 65 años, seguido de 12.970 individuos entre 51 a 64 años, 10.784 personas con 36 – 50 años, 5.668 sujetos de 25 a 35 años, 2.726 individuos de 18 a 24 años y 2.461 personas menores a 18 años. Con esto se evidencia que a medida que aumenta la edad de la población existe la probabilidad de que incremente el nivel de discapacidad visual (Miqueli, et al., 2016).

Figura 3

Población según grupo etario



Nota. El gráfico representa a personas con discapacidad visual por edad. Adaptado de *Estadísticas de Discapacidad* (párr.4), por CONADIS, 2021, <https://www.consejodiscapacidades.gob.ec/estadisticas-de-discapacidad/>






Tipos de discapacidad visual

De acuerdo con la OMS (2019) la discapacidad visual se divide en dos grupos como distante y cercana de presentación. El grupo de distante de presentación está conformado por discapacidad visual leve, moderada, grave y ceguera. El segundo grupo se enfoca en agudeza visual cercana. Este trabajo en particular se enfoca en el primer grupo mencionado.

Los tipos de discapacidad visual se detalla en la Tabla 1.

Tabla 1

Clasificación según distante y cercana de presentación

Tipos	Características
<p>Leve</p> 	<ul style="list-style-type: none"> • Grado: 20/30 a 20/60. • Visión casi normal, perciben objetos pequeños. • Uso de ayudas ópticas.
<p>Moderada</p> 	<ul style="list-style-type: none"> • Grado: 20/70 a 20/160. • Baja percepción de colores y detalles. • Uso de lentes.
<p>Grave</p> 	<ul style="list-style-type: none"> • Grado: 20/200 a 20/400. • Perciben luz y sombras pero necesita de Braille u otra tecnología para leer y escribir.
<p>Ceguera</p> 	<ul style="list-style-type: none"> • Grado ceguera casi total: 20/500 a 20/1000. • No perciben ni objeto ni luz con rango menor a 3/60. • Uso de Braille para lectoescritura y bastón.
<p>Agudeza visual cercana</p> 	<ul style="list-style-type: none"> • Inferior a N6 o a 40cm. • No perciben y detectan detalladamente los objetos o leer a distancia cercana.

Nota. La tabla representa los tipos de discapacidad visual. Adaptado de *Análisis sobre el uso de TICS para personas con discapacidad visual en centro Municipal Cuatro de Enero* (p.15), por C. García, 2018, Universidad de Guayaquil

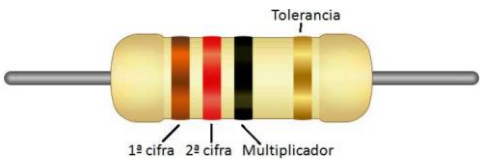

Circuitos eléctricos

Generalidades

De acuerdo con Cabello (2019) el circuito eléctrico se refiere a un conjunto de operadores vinculados entre sí, esto ayuda la circulación de los electrones con el fin de lograr un efecto de luz, movimiento, calor, entre otros (Cabello, 2019). Los elementos de un circuito eléctrico se describen a continuación en la Tabla 2:

Tabla 2

Elementos básicos de un circuito eléctrico

Elementos	Descripción
<p>Resistencia</p> 	<ul style="list-style-type: none"> • Componente de un circuito eléctrico que permite resistir el paso de corriente eléctrica a través de un circuito. • Pueden ser utilizados para reducir la intensidad de la corriente o bien para transformarla en valor. • Suelen estar compuestas por carbono.
<p>Inductor o bobina</p> 	<ul style="list-style-type: none"> • Componente de un circuito de carácter pasivo. • A partir de la autoinducción genera un campo magnético que reacciona ante cambios bruscos de corriente, generando voltajes opuestos proporcionales al recibido.

Capacitor



- Componente pasivo que almacena energía, mediante la formación de un campo magnético.
- Se compone de dos placas conductoras con cargas positiva y negativa respectivamente.
- Puede estar fabricado con discos de cerámico, mylar, tantalio, entre otros.

Fuente



- Son los elementos que permiten de proveer la corriente eléctrica al circuito.
- Estas pueden ser ideales o bien reales. Las ideales se utilizan para los cálculos teóricos, mientras que las reales presentan determinada resistencia que no permite que se presenten tal como las ideales.
- Las fuentes reales pueden ser de tensión o bien de intensidad.

Nota. La tabla representa los elementos básicos de circuitos eléctricos. Adaptado de *Principios de circuitos eléctricos* (p.19), por T. Floyd, 2007, Pearson Educación; *Resistencia Eléctrica*, por Fernández, (2021), en https://www.fiscalab.com/apartado/resistencia-electrica-conductor#resistencia_circuitos.

Tipos de circuitos eléctricos

Existen diversos tipos de circuitos eléctricos, y, por tanto, distintos diagramas que los representan con sus componentes. Estos dependen de cómo esté configurada

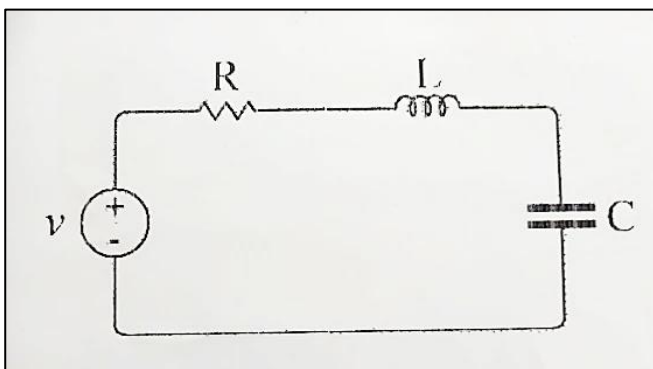
su conexión, y corresponden a circuitos eléctricos en serie, en paralelo y mixtos. En este trabajo se propone el reconocimiento de estos tres tipos de circuitos, los cuales se describen a continuación:

Los tipos de circuitos eléctricos son los siguientes:

- En serie: este tipo de circuito eléctrico permite la conexión de los dispositivos de manera secuencial (Figura 4) (Millán, 2018).

Figura 4

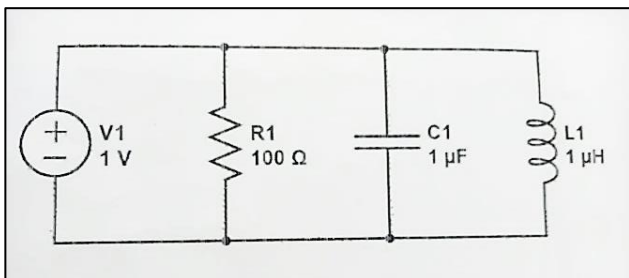
Circuito eléctrico en serie



- Paralelo: en este circuito eléctrico la alimentación y la salida de terminales de los dispositivos es similar para todos (Figura 5) (Millán, 2018).

Figura 5

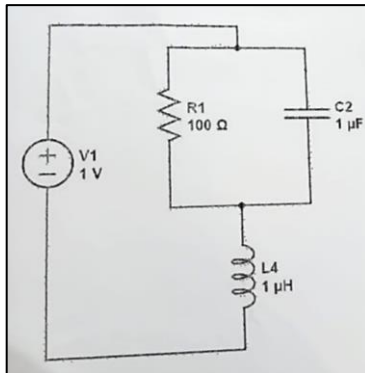
Circuito eléctrico en paralelo



- Mixto: este circuito eléctrico se deriva el tipo en serie y paralelo, pues, se emplean ambas para la conexión de dispositivos (Figura 6) (Cinjordiz, 2015).

Figura 6

Circuito eléctrico mixto



Procesamiento digital de imágenes

Generalidades

De acuerdo con Catalán (2019) el Procesamiento Digital de Imágenes (PDI) representa al conjunto de técnicas que se emplean para mejorar la calidad o buscar información de la imagen digital.

Según los antecedentes históricos se conoce que en los años 20s se realizó una de las aplicaciones iniciales de las imágenes digitales por la industria del periódico, donde se enviaron fotografías mediante cable submarino denominado sistema Bartlane. Posteriormente, se cambió por el sistema uso de cintas perforadas. En los años 60s – 70s se implementaron técnicas para observación de imágenes relacionadas con la tierra y astronomía. En los 70s se utilizó el PDI para la tomografía axial computarizada, obteniendo imagen 3D, a partir de esta década se produjo avances impresionantes, por lo que se comenzó al emplear los algoritmos en diversas aplicaciones, incluso las

nuevas tecnologías y el Internet ha permitido crear nuevas oportunidades en el procesamiento digital de imágenes (Mejía, 2017).

En la actualidad el PDI se aplica en la mayoría de campos como en la medicina nuclear, medicina, biología, astronomía, litografía, microscopía, organizaciones para el control de calidad, biometría, agricultura, ganadería, entre otros. En la medicina nuclear se utiliza rayos gamma para realizar la tomografía de emisión de positrones. En la astronomía, los rayos gamas se emplea para detectar la radiación natural. En la medicina se relaciona con la angiografía para obtener imágenes de venas y arterias; incluso en la industria se usa rayos x para el control de calidad. En la litografía se utiliza ultravioleta basado en la microscopía de fluorescencia. Las ondas de radios se emplean en el sector médico para realizar resonancia magnética (Duarte & González, 2017).

Por otro lado, Valverde (2016) detalla que los componentes principales de un sistema de procesamiento digital de imágenes son los siguientes:

- Sensor: componente sensible a la energía.
- Hardware de procesamiento especializado: representa el digitalizador que facilita la transformación de la salida del sensor en imágenes.
- Computadora: se puede emplear una PC o supercomputadora para el procesamiento.
- Software: se refiere a módulo que se utiliza para efectuar tareas.
- Almacenamiento en masa: se considera almacenamiento a corto plazo (memoria de computadora), en línea (discos ópticos) y de archivo.
- Dispositivos de despliegue de imágenes: se refieren al uso de monitores planos de color y dispositivos estéreo (lentes).

- Dispositivos de copia dura: se utiliza impresoras láser – tinta, cámara de película, entre otros.
- Red: se trata del ancho de banda de la red adecuada para el procesamiento.

Pasos para el procesamiento digital de imágenes

Para Mejía (2017) los pasos para realizar el procesamiento de imágenes se detallan a continuación:

- Adquisición de imágenes: inicia con el preprocesamiento enfocado en escalar la imagen.
- Mejorar imagen: se procede a recabar los detalles con precisión o las características de interés para tengan un mejor aspecto.
- Restaurar la imagen: se realiza una mejora de la imagen de forma objetiva, es decir, la imagen se transforma como era previo a que se dañe.
- Procesamiento de color: se aplica mecanismos especiales para añadir color.
- Ondeletas: se realiza la representación de las imágenes en diferentes niveles de resolución.
- Compresión: se enfoca en la reducción de almacenamiento de una imagen, así como el ancho de banda para la transmisión.
- Procesamiento morfológico: permite la extracción de componentes de las imágenes con el fin de realizar de representación y formas.
- Segmentación: en este paso se realiza la división de la imagen en

diferentes partes.

- Representación y descripción: se determina si la forma debe tratarse a nivel de frontera o región, lo que permite extraer atributos.

Tipos de extracción de características de imágenes digitales

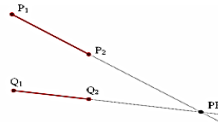
Para González (2016) la extracción de las características de las imágenes mediante el procesamiento digital debe cumplir con cuatro condiciones. La primera se refiere a al tiempo extracción debe ser lo más pequeño y el sistema no debe ser de un alto costo. La segunda se trata de determinar la localización exacta, evitando errores en la estimación de las características. En la tercera, la caracterización debe ser estable y robusta. En la última condición se debe extraer la información geométrica básica de la imagen como los puntos, líneas y círculos.

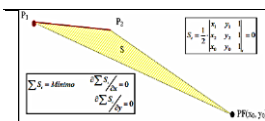
- **Extracción de puntos**

Para la extracción de puntos se considera dos enfoques para la detección de las características. Con el primero se obtiene los puntos mediante inserción de aristas (puntos de fuga) y el segundo trabaja de forma directa sobre imágenes gris (puntos esquina). Esto se aprecia en la Tabla 3.

Tabla 3

Métodos de extracción de puntos

Métodos	Descripción
A partir de aristas: puntos de fuga	Método de inserción de rectas: <ul style="list-style-type: none"> • Permite establecer la intersección de dos rectas. • Se tiene una recta con distancia al origen y ángulo.
	Método de minimización del área del triángulo:



- Cálculo de la superficie mínima del triángulo.
- Se considera los segmentos extraídos y del punto de fuga.

A partir de propia

esquina: puntos

esquina



- Considera todos los puntos de la imagen.
- El punto es una esquina.
- Detectan los niveles de grises del área horizontal, vertical y diagonal.

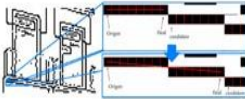
Nota. La tabla representa los métodos de extracción de puntos. Adaptado de *Procesamiento de imágenes* (pp.11-15), por D. González, 2016, Universidad de Salamanca.

• Extracción de líneas

Para la extracción automática de líneas se considera los métodos de Canny + Burns, RANSAC + MMCC y transformada de Hough. En la Tabla 4 se describen los métodos para la extracción de líneas:

Tabla 4

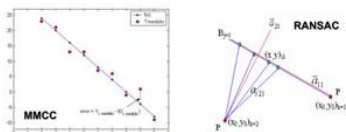
Métodos de extracción de líneas

Métodos	Descripción
Canny + Burns 	<ul style="list-style-type: none"> • Extrae bordes utilizando algoritmo Canny para imágenes con geometría regular. • El algoritmo Canny es preciso para localización del contorno y fiabilidad en detección de puntos. • Luego de la segmentación utiliza el algoritmo

Burns como base la interpretar la escena en 3D y el procesamiento depende de los pixeles.

RANSAC+MMCC

(RANmdom SAmple Consensus)



- RANSAC ayuda en la detección de errores, seleccionando dos puntos aleatorios, valida la recta y elimina los errores.
- MMCC facilita la extracción de líneas.

Transformada de Hough

- Es un algoritmo que permite encontrar líneas.
- Transforma los puntos de borde en espacios.
- Presenta robustez al ruido.

Nota. La tabla representa los métodos de extracción de líneas. Adaptado de *Procesamiento de imágenes* (pp.16-21), por D. González, 2016, Universidad de Salamanca

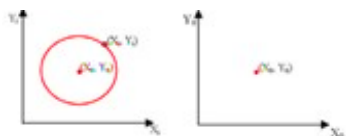
- **Extracción de círculos**

En la Tabla 5 se muestra el método para extracción de círculos:

Tabla 5

Método de extracción de círculos

Métodos	Descripción
Transformada de Hough para círculos	<ul style="list-style-type: none"> • Aplica el mismo proceso que la transformada de Hough de líneas. • Utiliza espacio de parámetros, pues, cuando se conoce el radio, los círculos se transforman en bidimensionales, donde el espacio es un punto. • El dominio espacial representa a un círculo



formado por todos los puntos.

- Detecta píxeles mediante filtros.

Nota. La tabla representa los métodos de extracción de círculos. Adaptado de *Procesamiento de imágenes* (p.22), por D. González, 2016, Universidad de Salamanca

Redes neuronales

Generalidades

Las redes neuronales representan un elemento esencial en la inteligencia artificial, pues, se trata de un modelo diseñado según las funciones del cerebro humano. Se encuentran formadas por varios nodos cuyo funcionamiento se basa en neuronas debido a que envían información entre las mismas. Una vez que las redes reciben información, procesan y generan la salida enfocada con las predicciones acorde a lo programado, aplicadas específicamente para dar soluciones (Russell, 2018).

Respecto al origen de las redes neuronales se conoce que está vinculado con los comienzos de la inteligencia artificial, el cual fue evolucionando según las investigaciones desarrolladas. A partir de 1943, los investigadores McCulloch y Pitts crearon la denominada lógica umbral basado en un modelo de red neuronal, con este avance se dio espacio para estudios enfocados en el desarrollo biológico del cerebro y la implementación en redes de inteligencia artificial. Luego se desarrolló el aprendizaje de Hebb, este se basó en un modelo de aprendizaje no supervisado (Three Points The School for Digital Business, 2021).

Después, se crearon las máquinas tipo B por Turing, estas funcionaban según estudios de psicología y neurología debido a que se enfocaba en el proceso de la plasticidad neuronal. Por lo tanto, las investigaciones de las redes neuronales humanas permitieron diseñar las funciones de las artificiales, lo cual ayudó a profundizar el

aprendizaje de máquina. Para 1958 Rosenblatt desarrolló el algoritmo perceptrón relacionado con el aprendizaje de redes neuronales por computadora, utilizando dos capas (adición y sustracción) (Three Points The School for Digital Business, 2021).

Por otra parte, los modelos de las redes neuronales se definen con funciones matemáticas (X-Y) o algoritmos de aprendizaje. Además, tiene dos funciones relacionadas con la red y aprendizaje (Three Points The School for Digital Business, 2021).

- **Función de red:** Se trata de las interconexiones de las neuronas, donde la primera capa representa un mecanismo de entrada para recepción de información que se envían a la segunda y tercera capa respectivamente.
- **Función de aprendizaje:** En esta función se determina una tarea a cumplir basado en aprendizaje supervisado, el que corresponde a la resolución de tareas basadas en conocimientos previos; el no supervisado, que refiere al establecimiento de distintos parámetros para una solución; y el aprendizaje por refuerzo, el cual corresponde a aquel que no dispone de datos, pero se fundamenta en la interacción de determinados agentes con el entorno.

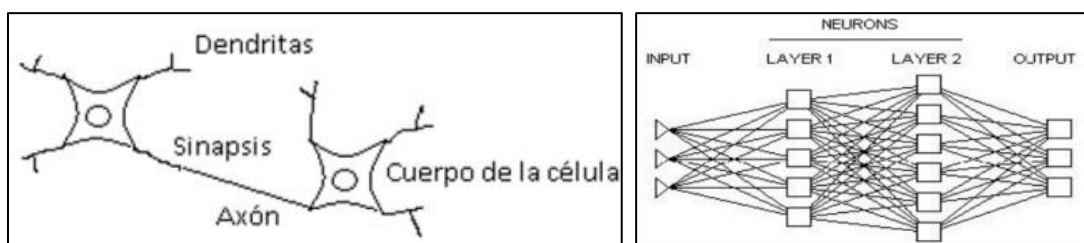
Redes neuronales artificiales

Las redes neuronales artificiales (RNA) corresponden a un sistema compuesto por un conjunto sucesivo de capas, que contienen a su vez numerosas unidades interrelacionadas, las cuales han sido denominadas como neuronas por la similitud que presentan a las neuronas cerebrales. En este modelo, los nodos representan a las neuronas, mientras que cada una de las conexiones que se establecen entre ellos corresponden a un símil de la sinapsis, a modo de réplica de las características biológicas del cerebro humano (Valderrama et al, 2021).

Dadas sus características, las RNA tienen la capacidad de adquirir aprendizajes a medida que ejecutan determinadas soluciones, y, por tanto, no requieren de programación explícita. Entre los tipos de capacidades que desarrollan a través del aprendizaje se encuentran la capacidad para predecir valores, identificar semejanzas o reconocer patrones, entre otras. Así, a través de complejos procesos estadísticos, son capaces de tomar información disponible y generar nuevos aprendizajes, incluso en caso de que no existan datos previos (Erzin & Turkoz, 2016).

Figura 7

Comparativa neurona cerebral y RNA



Nota. En la imagen de la izquierda se observan dos neuronas y sus componentes. A la derecha se presenta el modelo de capas en RNA. Ambas imágenes son tomadas de *Principios y características de las redes neuronales artificiales*, por Acevedo et al, 2017, en *Desarrollo e innovación en Ingeniería*, Editorial Instituto Antioqueño de Investigación.

En términos generales, las RNA presentan las siguientes capas:

- Entrada: refiere a la capa que es capaz de captar información del entorno para ingresarla al sistema.
- Oculta: corresponde al conjunto de capas donde se realizan los procesamientos de información.
- Salida: corresponde al output del sistema, el cual corresponde a una salida de información que sirve de entrada para otra neurona.

Principios y elementos de RNA

Una vez que las RNA son entrenadas, son capaces de resolver una serie de problemas basadas en el aprendizaje. Esto se encuentra fundamentado en estos principios (Figura 8).

Figura 8

Principios de funcionamiento de RNA

Aprendizaje adaptativo	Funcionan a partir de entrenamiento.
	El entrenamiento pueden ser ejemplos, sin necesidad de ser modelos establecidos a priori.
Autoorganización	La RNA, dependiendo del objetivo determinado, puede modificarse.
	Pueden reconocer patrones a partir de la generalización de experiencias previas sin necesidad de disponer de un patrón previo.
Tolerancia a fallos	La información en la red es redundante y distribuido, de modo que ante fallos no se pierde.
	Aun cuando parte de las neuronas o conexiones fallen, las RNA tienen tolerancia a estos fallos y pueden continuar con su funcionamiento.
Funcionamiento en tiempo real	Dado que la información se actualiza simultáneamente, tienen la capacidad para procesar datos en tiempo real.
Adaptación a la tecnología existente	Tanto en hardware como en tecnologías diversas, las RNA se adaptan de manera sencilla y rápida a estas, siendo muy versátiles.

Nota. Adaptado de *Principios y características de las redes neuronales artificiales*, por Acevedo et al, 2017, en *Desarrollo e innovación en Ingeniería*, Editorial Instituto Antioqueño de Investigación.

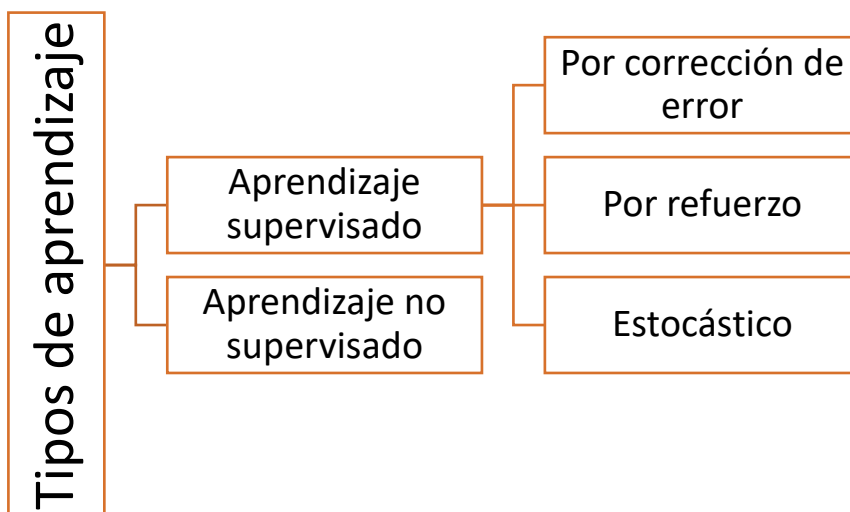
Como elementos que conforman a las RNA, pueden destacarse los siguientes (Acevedo et al, 2017):

- **Peso sináptico:** corresponde a un valor asignado a cada neurona de la red que le otorga determinada funcionalidad y permite almacenar su información.
- **Propagación:** corresponde a una regla en que se determina la propagación de las salidas de cada neurona para combinarlas con otras según determinado patrón de conexión. Con esto es posible establecer el valor de entrada de cada uno de los nodos o neuronas.
- **Funciones de activación:** esta función refiere a aquella que, en conjunto con la regla de propagación, permite darle salida a cada neurona. Se elige según el tipo de entrenamiento de la RNA.

Respecto del tipo de aprendizaje que pueden presentar las RNA, destacan (Figura 9):

Figura 9

Tipos de aprendizaje RNA



Nota. Adaptado de *Principios y características de las redes neuronales artificiales*, por Acevedo et al, 2017, en *Desarrollo e innovación en Ingeniería*, Editorial Instituto Antioqueño de Investigación.

Como se observa, el aprendizaje de las RNA puede ser supervisado o no supervisado. En lo que refiere al primero, cabe mencionar que el entrenamiento que se aplica a la RNA para aprender estará bajo la supervisión de los diseñadores que programen esta instancia, indicando el tipo de respuesta que debe dar la red ante determinada entrada. Cuando las salidas no correspondan al resultado esperado, el supervisor modifica cada uno de los pesos sinápticos asignados para modificar y entrenar la respuesta esperada para conseguir una mayor aproximación (Acevedo et al, 2017).

En particular, el aprendizaje por corrección de error consiste en el proceso de enseñanza de los resultados esperados, y se va ajustando a medida que se van obteniendo resultados, de modo que cada vez se esté más cerca de la salida ideal.

Por otra parte, el aprendizaje por refuerzo implica el entrenamiento sin ejemplos, sino con un refuerzo al obtener un resultado, el cual puede ser el éxito o el fracaso. Así se va ajustando la respuesta por parte del supervisor, hasta obtener resultados adecuados (Cubero & Poler, 2020).

El aprendizaje estocástico refiere a la ejecución de cambios aleatorios en cada uno de los pesos sinápticos de la red para evaluar su aprendizaje. Cuando las salidas no corresponden a los resultados esperados, se modifican nuevamente bajo nuevos parámetros basados en otra distribución de probabilidades.

Por último, en aquellos aprendizajes que no son supervisados, los entrenamientos no se realizan bajo la supervisión de programadores o de diseñadores, de modo que la RNA se autoorganiza a sí misma (hasta cierto punto). Es de manera autónoma que realiza el descubrimiento de patrones, categorías, etc. (Acevedo et al, 2017).

Red neuronal profunda

Las redes neuronales profundas son un tipo de RNA que presentan determinadas capas ocultas entre la entrada y la salida, de modo que con ello, al igual que las otras RNA no profundas, son capaces de establecer relaciones complejas no lineales (Código Fuente, 2021). Estas se distinguen de las RNA no profundas por la cantidad de capas escondidas que presentan entre una entrada y salida, siendo mucho mayor.

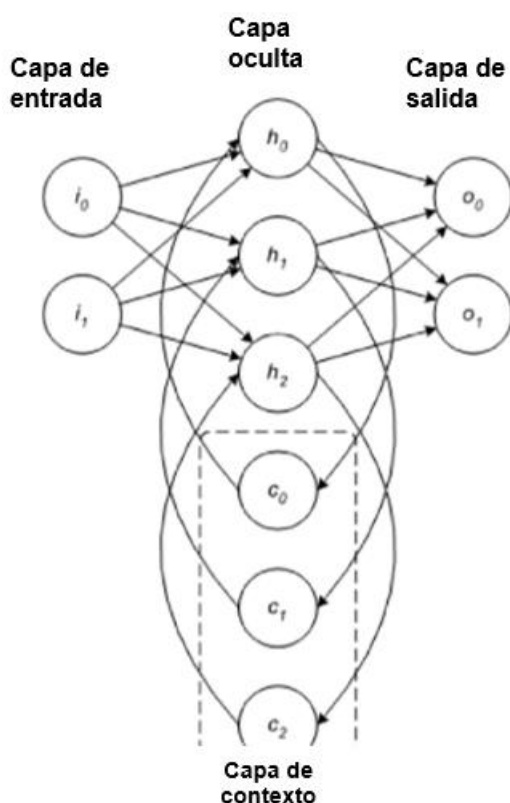
El aprendizaje profundo (DL por sus siglas en inglés) que llevan a cabo estas redes ha propiciado un enorme desarrollo de la inteligencia artificial (IA). Permite la operar a los sistemas computacionales a partir de la representación de datos a diferentes niveles de abstracción, y sus algoritmos pueden identificar patrones en grandes cantidades de datos, aun cuando se encuentren muy ocultos. El algoritmo

responsable de esto es el de retroprogramación, con el cual la red, con más de millones de distintos pesos sinápticos ajustados y millones de ejemplos, realiza un aprendizaje de alta complejidad (Cubero & Poler, 2020).

En las RNA que se basan en aprendizajes profundos el desarrollo de distintas arquitecturas se ha sostenido con dependencia de sus aplicaciones (Jones, 2017). Una arquitectura básica de una RNA profunda se caracteriza por tener muchas más capas ocultas que una “tradicional”, tal como se observa a continuación (Figura 10) un ejemplo de red neural recurrente (RNN).

Figura 10

Arquitectura RNN



Nota. Tomado de *Arquitecturas de aprendizaje profundo*, por Jones, 2017, en <https://developer.ibm.com/es/articles/cc-machine-learning-deep-learning-architectures/>

Otro tipo de arquitectura de RNA basada en el aprendizaje profundo son las redes neuronales convolucionales, las que se exponen a continuación.

Red neuronal convolucional

Las redes neuronales convolucionales (CNN por sus siglas en inglés) corresponden a un tipo de RNA, DL. Es el algoritmo que más suele utilizarse actualmente para su aplicación y desarrollo en distintos ámbitos (Esparza, 2020), con especial énfasis en el reconocimiento de imágenes.

En particular, los nodos o neuronas de esta RNA funcionan homologando el campo receptivo de una neurona de tipo “visual” en un cerebro humano. A partir de una variación multicapa de un perceptrón, pueden ser ampliamente utilizadas en visión por computadora. Las múltiples capas que presentan filtros (conjunto de kernels) de tipo convolucionales realizan mapeos no lineales.

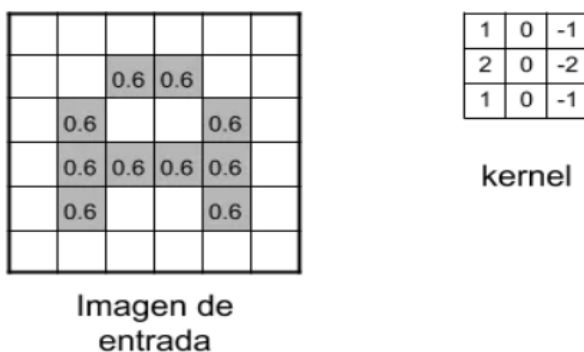
Para su aprendizaje, las CNN deben ser entrenadas en base a ejemplos, los cuales pueden ser enormes bases de datos, a partir de las cuales pueden realizar inferencias y generalizaciones de los elementos, objetos o gestos que esté visualizando. El proceso convolucional consiste en tomar el grupo de píxeles que se encuentran cerca de la entrada, sobre los cuales se realiza una operación escalar sobre una matriz kernel (Big Data, 2021).

Como se observa en la Figura 11, la imagen de entrada se encuentra primeramente binarizada, para luego desarrollarse el proceso de convolución. En el ejemplo, la matriz kernel es de 3x3, suponiendo que es adecuado para procesar la información de las neuronas tanto de datos de entrada como de salida. Como se plantea en Big Data (2021), los datos son aleatorios en principio, y luego se adecuan

mediante backpropagation. Estos procesamientos se reproducen a un gran número dentro de una CNN, puesto que se cuenta con varios filtros.

Figura 11

Ejemplo de resultado de procesamiento kernel en CNN

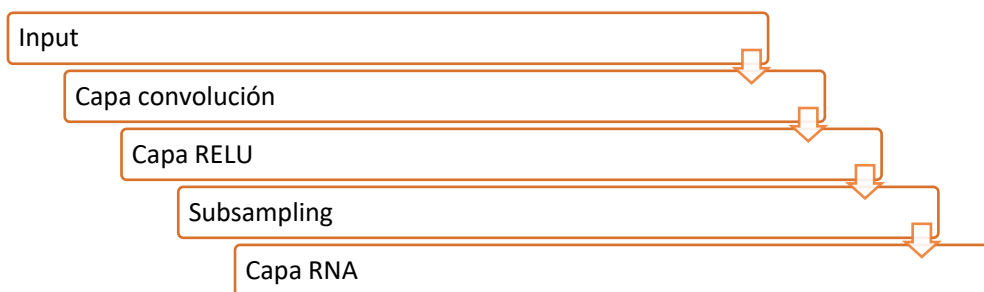


Nota. Procesamiento de un kernel. En caso de ser imagen a color, el kernel sería de 3x3x3. Tomado de *Redes Neuronales Convolucionales*, por Big Data, 2021, en <https://www.juanbarrios.com/redes-neurales-convolucionales/>

De esta manera, en particular las CNN presentan una arquitectura como la siguiente (Figura 12):

Figura 12

Arquitectura básica CNN



Nota. Adaptado de *Redes Neuronales Convolucionales*, por Big Data, 2021, en <https://www.juanbarrios.com/redes-neurales-convolucionales/>

Luego del primer procesamiento convolucional, se desarrollan la segunda y posteriores convoluciones, en donde se aplicarán las funciones de aplicación y muestreo, lo que deriva en una red neuronal multicapa, derivando posteriormente en una salida.

Síntesis de habla

Definición

La síntesis de habla corresponde al proceso mediante el cual se produce la síntesis de voz artificialmente. Estos procesos se presentan en dos grupos, los cuales han sido desarrollados desde su origen hasta la actualidad (Duxans & Costa, 2017):

- Respuesta vocal: este tipo de síntesis de habla tiene como objetivos el desarrollo de frases acústicas en una estructura fija, la cual solo presenta variaciones de léxico. Esto se produce a través de la concatenación de fragmentos grabados con anterioridad.
- Conversor texto-voz: este grupo tiene como objetivo reproducir un texto de manera acústica con independencia de su estructura u origen. Esta tecnología se sostiene en dos grandes subprocesos, que son el procesamiento del texto para reconocer sus elementos y posteriormente la sintetización de la voz como salida.

De estos dos grupos, es claro que la tecnología de conversión texto a voz representa una mayor potencialidad en sus aplicaciones, puesto que permite generar un conjunto de servicios que pueden tener aplicaciones como adaptaciones para personas con capacidades diferentes, reconocimiento y aprendizaje TIC, etcétera.

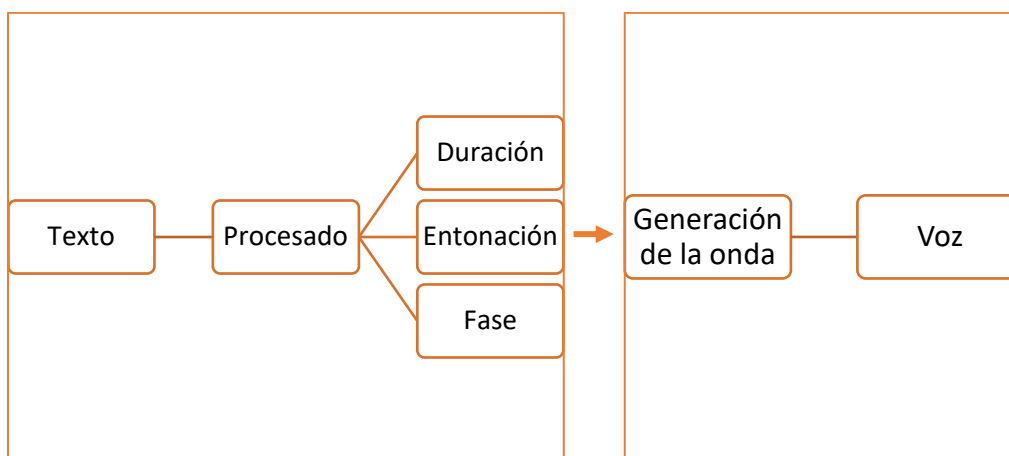
TTS

La síntesis de habla basada en la producción de un texto reproducido en audio. Esta es realizada en la actualidad a través de la *tecnología text to speech* (TTS). Este proceso tiene dos grandes etapas: representación lingüística y la sintetización (Llisterri, 2017). El primer módulo realiza un procesamiento en que se reproduce un lenguaje natural, mientras que el sintetizador digitaliza la información (Duxans & Costa, 2017).

La arquitectura básica de esta tecnología es la siguiente (Figura 13):

Figura 13

Arquitectura de la tecnología de síntesis de voz (TTS)



Nota. Se representan las dos etapas del proceso TTS. A la izquierda Front-End, a la derecha sintetizador. Adaptado de *Adaptación del sistema de conversión de texto a voz extremo a extremo Tacotron a otras lenguas*, por García, 2019, Universidad del País Vasco.

Como se observa, cada uno de los módulos se encarga de una función particular; se comienza por la introducción de un texto, el cual es procesado en la fase Front – End, la cual corresponde a la representación. En sí misma, está compuesta por (Duxans & Costa, 2017):

- Normalización de un texto: aquí se realiza el análisis morfosintáctico del texto que ingresa, y es convertido a una forma convencional, es preparado para su transcripción. Se identifican elementos como abreviaturas, se realiza una silabificación, se identifica la acentuación correspondiente, entre otros elementos.
- Conversión fonética: en este subproceso se realiza la transcripción a fonemas de cada partícula o palabra en particular, secuenciando los fonemas según el idioma.
- División prosódica: el texto es separado en diferentes unidades que permiten ajustar condiciones melódicas y otro tipo de patrones, como los rítmicos.

La segunda fase del proceso, el sintetizador, corresponde a la transformación de la salida que produce la primera etapa y la transforma en una voz sintética. Esto puede realizarse a través de distintas tecnologías, entre ellas síntesis articulatorias, que son las tecnologías que aparecieron primeramente, o la concatenación, que es la más utilizada en la actualidad. Esta última consiste en la concatenación de partículas de audio que se encuentran en una base de datos que suelen ser grabadas por un único locutor (Duxans & Costa, 2017).

Las redes neuronales son una de las tecnologías fundamentales mediante las cuales se desarrolla la síntesis de voz. Estas, a diferencia de otras tecnologías, son capaces de realizar el proceso completo, tanto el procesamiento del texto como su sintetización y conversión al audio de salida, el cual se conoce como sistema *end to end*. Esta tecnología se ha denominado Deep Voice con sus distintas versiones (Moya, 2021).

Tacotron, por su parte, es una tecnología que realiza la síntesis de voz mediante las RNA y su entrenamiento, las que transforman el texto de entrada en espectrogramas, dando como resultado salidas de alta calidad, es decir, resultados muy legibles. DeepSpeech, herramienta de código abierto impulsada por Mozilla, está desarrollada en Python y tiene una librería extensa de código abierto en donde sus colaboradores aumentan constantemente los repositorios (Moya, 2021).

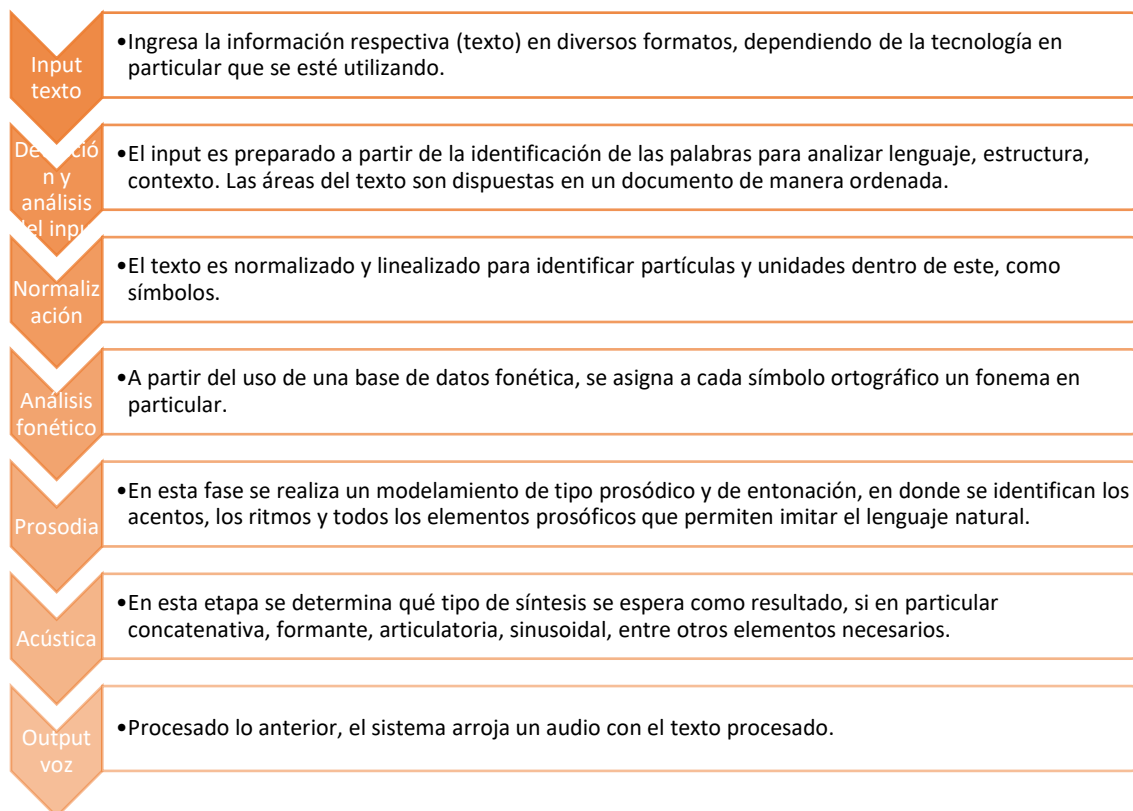
Proceso TTS

Como plantean Duxans y Costa (2017), el proceso de la tecnología de síntesis de voz se desarrolla a través del siguiente proceso (

Figura 14):

Figura 14

Proceso de síntesis de voz (TTS)



Nota. Adaptado de *Síntesis del habla*, por Duxans & Costa, 2017, Universitat Oberta de Catalunya.

Detección de la mano

A partir del desarrollo de la IA se han gestado un sinnúmero de aplicaciones de esta a distintos servicios que permiten ofrecer experiencias de usuario mucho más ajustadas a las necesidades que sin ella. Actualmente, existen distintos dispositivos que pueden realizar distintas tareas, como es la detección de gestos y la palma de la mano (Rocha, Segundo, & Hernández, 2015).

Dado que el objetivo de generar interacción humano-máquina, se han establecido distintos dispositivos hardware para realizar la identificación de gestos de las palmas de la mano, como distintos tipos de sensores ópticos o guantes, entre otros, pero han tenido elevados costos y no son lo más adecuado para los usuarios (Langford et al, 2016). Debido a lo anterior, lo más desarrollado en la actualidad se sostiene en la visión por computadora, lo que ha permitido que la máquina detecte movimiento y gestos para controlar cursor, navegar en 3D, etc. Esto ha podido desarrollarse gracias a las tecnologías de cámara como RGB-D (sensores de profundidad), que captan bastante bien los gestos.

Características

En términos generales, el reconocimiento gestual que se realiza a través de la tecnología de visión artificial o visión por computadora ha avanzado conforme avanza la investigación en IA. Así, además de conformarse por dispositivos de entrada, como cámaras, debe basarse en algoritmos que permitan identificar hasta determinado grado el movimiento corporal, incluyendo aquí el movimiento que realizan las manos. De esta

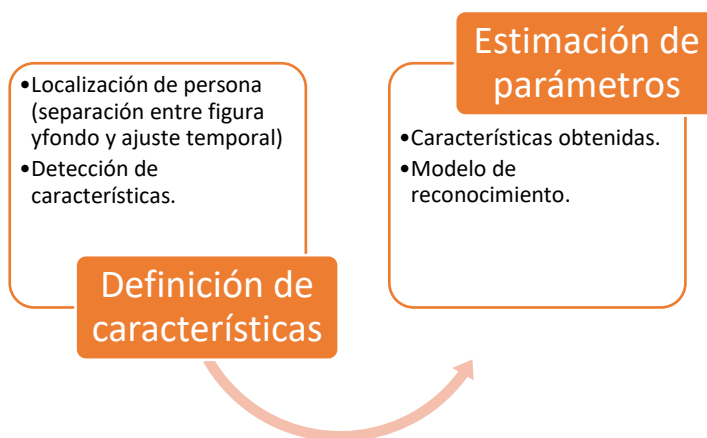
manera, es preciso identificar como uno de los elementos fundamentales el tipo de gesto o de movimiento que debe ser reconocido (González & Yimes, 2016):

- Gesto estático: en este caso, se reconoce una postura fija.
- Gesto dinámico: se reconoce el movimiento mediante la valoración de la postura a través del tiempo.

Se encuentran, por tanto, dos etapas básicamente en el proceso de reconocer gestos. La primera refiere a la detección de las características de las imágenes, en la cual se realiza una identificación de los elementos relevantes; la segunda corresponde a la estimación de parámetros, para lo cual se utilizan los elementos relevantes obtenidos previamente. Esto se sintetiza en la Figura 15:

Figura 15

Etapas del reconocimiento gestual



Nota. Adaptado de *Sistema de reconocimiento gestual de lengua de señas chilena mediante cámara digital*, por González y Yimes, 2016, Pontificia Universidad Católica de Valparaíso.

Existen diferentes modelos de reconocimiento que han sido desarrollados. Entre estos, por ejemplo, se encuentra LeapMotion, dispositivo USB que utiliza cámaras LED infrarrojas que se adapta a un escritorio o mesa. Con él es posible realizar observaciones para navegar sitios web, zoom, dibujos, etc. Wisedge, por su parte, es un modelo que no requiere de dispositivos especiales, sino que corresponde a una interfaz que detecta gestos humanos mediante redes inalámbricas. Kinect es otro modelo de reconocimiento, desarrollado por Microsoft para videojuegos (González & Yimes, 2016). Existen otra serie de modelos que dan cuenta de la diversidad de aplicaciones que pueden tener las tecnologías de reconocimiento de gestos, y en particular aplicadas al reconocimiento de gestos de la palma de la mano.

Detección de los dedos de la mano

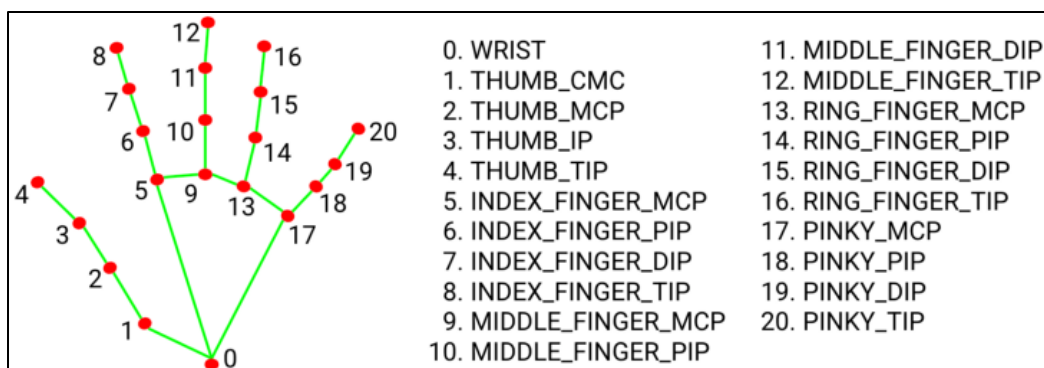
Cuando se ha detectado la posición inicial de la mano, es preciso que los modelos de detección puedan identificar sus partes. Para esto se realiza el reconocimiento de 21 *landmarks* (puntos de referencia). En general, esto se lleva a cabo mediante la implementación de modelos que permiten aprender cómo se representan las manos y sus posturas usuales, mediante la identificación de las coordenadas de cada uno de estos 21 puntos (x e y), además de categorizaciones binarias de las manos, como por ejemplo el ser una mano izquierda o derecha, etc. (Zhang, y otros, 2021).

MediaPipe permite identificar tanto la palma de la mano como los puntos de referencia de esta (*landmarks*) (Figura 16). Esto se realiza a través de un modelo holístico o de carácter múltiple basado en redes neuronales convolucionales. En estas capas convolucionales se evalúan los píxeles próximos, reduciendo con ello la capacidad de procesamiento necesaria.

Cuando ingresan imágenes desde fotogramas, el modelo permite rastrearlos para identificar nuevas posiciones. Así, el *tracking* es el seguimiento de los puntos de la mano, lo que permite no identificar en cada fotograma nuevamente la imagen y sus *landmarks*, sino rastrearlos para identificar cómo se mueve la mano en el espacio (MediaPipe, 2021). Este seguimiento manual (*MultiHandDetection*) incluye el módulo de detección de los dedos, *MultiHandLandmark*, además de *MultiHandRender*, que permite visualizar estos puntos en la imagen de salida (Doménech, 2020). La detección en particular del dedo índice se encuentra añadida una función que, una vez conectados los puntos entre sí, permite añadir las coordenadas particulares en la que se ubica el dedo índice (index finger), identificar su centro y modificar el color para representar un render que lo destaque.

Figura 16

Ubicación de puntos de referencia en las manos (landmarks)



Nota. Imagen tomada de *Visión por computadora usando Mediapipe*, por Ichi, 2021.

Recuperado de <https://ichi.pro/es/vision-por-computadora-usando-mediapipe-211738590199029>

Tarjeta NVIDIA Jetson Nano

Definición

NVIDIA® Jetson Nano™ corresponde a un kit de procesamiento que facilita la programación y entrenamiento de RNA paralelas para diversas aplicaciones, entre las que constan la identificación y clasificación de imágenes, detección de objetos y gestos y procesamiento del habla (NVIDIA, 2021).

Características

Este equipo de desarrollo se compone de un módulo Jetson P3448-0000 que se conecta a una placa base de referencia P3449-0000. Va acompañada además de NVIDIA® JetPack™ SDK, que incluye:

- Linux y sus controladores para NVIDIA
- Bibliotecas, API y Computer Vision
- Distintas herramientas de desarrollo
- Documentación necesaria y muestra del código

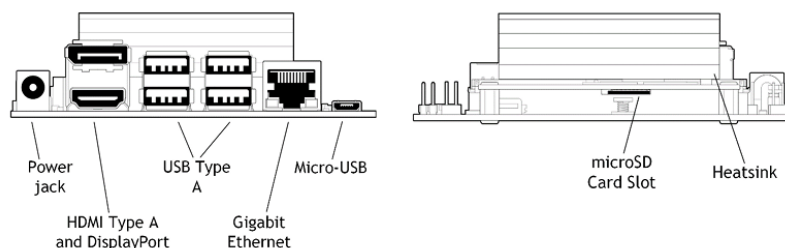
Como elementos que se incluyen en el kit, se presenta:

- Módulo Jetson Nano con disipador
- Placa base de referencia
- Paper card para inicio rápido y ayuda

Los módulos del kit se visualizan a continuación (Figura 17):

Figura 17

Vista frontal y trasera de la NVIDIA Jetson Nano



Nota. Vista frontal y trasera del kit de desarrollo Jetson Nano. Tomado de *Jetson Nano Developer Kit. User Guide*, por NVIDIA, 2020. Recuperado de https://developer.nvidia.com/embedded/dlc/Jetson_Nano_Developer_Kit_User_Guide

Software

Python

Corresponde a un lenguaje de programación que se sostiene en un principio de legibilidad de su código, lo que constituye parte fundamental de la filosofía de programación. Es de carácter multiparadigma, de manera que se compone de elementos de tipo imperativo, funcional y con orientación a objetos, los que se encuentran adecuadamente desacoplados, tal como plantean Troyano et al (2018). Esto implica que sea sencillo y progresivo el aprendizaje de este lenguaje, y, por tanto, el proceso se realiza progresivamente. Además, presenta un gran número de bibliotecas de herramientas para el desarrollo. Entre estas cuentan herramientas para lectura de ficheros de diversos formatos, desarrollo gráfico de interfaces, aneji3n a bases de datos, entre otras. Estos repositorios junto con la facilidad de implementaci3n lo vuelven un lenguaje sencillo y potente.

OpenCV

Corresponde a una librería o grupo de algoritmos que permiten la detección de objetos bajo tecnología de visión artificial, entre los que pueden contar distintas figuras geométricas, rostros, entre otros, el cual se encuentra disponible para Python. Además, permite realizar la detección de figuras a partir de la identificación de puntos específicos (Agustí, 2019). El sistema permite, además, binarizar imágenes; es decir, transformarlas a una escala de grises y, posteriormente, a blanco y negro (Martínez et al, 2021).

Mediapipe

Mediapipe es un framework multimodal de código abierto diseñado por Google Open Source, que permite poner en marcha proyectos en distintos dispositivos para tecnología *machine learning*. Permite la detección de gestos, rostros, manos, segmentos de cabello, objetos, entre otros, y presenta un muy buen desempeño. Se encuentra disponible en varios lenguajes, incluyendo Python (OMES, 2021).

gTTS

Corresponde a un paquete o librería completo para Python que utiliza API de Google TTS. Permite realizar síntesis de voz mediante tecnología TTS, que tiene amplia documentación y que permite ofrecer distintas funcionalidades al proyecto en desarrollo (Parzibyte, 2019).

Visual Studio Code

Corresponde a una plataforma de edición de código fuente realizada por Microsoft, la cual se encuentra disponible para varios lenguajes, entre ellos Python. Funciona en distintos S.O., y presenta diversas funcionalidades, como el soporte para

depuración, resaltado de sintaxis, entre otros (Microsoft, 2021). Si bien es gratuito, se descarga a través de software privativo.

Capítulo III

Metodología de desarrollo del proyecto

En el presente capítulo se llevará a cabo una descripción de la metodología que fue utilizada para el desarrollo del prototipo. Para esto, se apunta el problema a resolver con la herramienta diseñada y se describen las necesidades que busca satisfacer; posteriormente, se analiza el diagrama de bloques del sistema, sobre el cual se desarrollan cada una de sus etapas. En cada una de estos bloques se detalla lo realizado y las herramientas utilizadas para ello.

Problema y necesidades

La accesibilidad académica es un aspecto fundamental para cumplir con los deberes del Estado, y, por tanto, para garantizar el efectivo ejercicio de los derechos fundamentales de las personas. Esta noción implica la supresión de las barreras u obstáculos que no permiten el desarrollo adecuado de aprendizajes y competencias en la población discapacitada, y es imprescindible para generar una real integración de estas personas al ámbito académico (Palma, 2021). Sin embargo, las personas con discapacidad visual se ven continuamente impedidas o disuadidas de ingresar a la educación técnica o superior debido a la dificultad que representan los estudios de este tipo de áreas por la falta de herramientas tecnológicas que les permita estudiar otras carreras en diferentes áreas de conocimiento.

En particular, el área de las ingenierías y tecnicaturas en electrónica son áreas que requieren de aprendizajes basados en la identificación de elementos abstractos (símbolos) o teóricos. Esto constituye muchas veces un factor que disuade a las personas con discapacidad visual de estudiar este tipo de carreras, debido a que no

existen mecanismos ni estrategias que permitan a esta población acceder efectivamente a estos conocimientos.

Este proyecto es relevante porque ayuda a las personas con discapacidad visual en el reconocimiento de imágenes, siendo útil para el desenvolvimiento e inclusión a nivel educativo, social y laboral. Su aplicación en las unidades educativas podría ofrecer mayores alternativas de estudio a esta población, dado que facilitaría el acceso a los aprendizajes de este tipo. En carreras como Ingeniería Eléctrica, los circuitos eléctricos constituyen un aprendizaje esencial que deben adquirir los estudiantes y trabajadores del área de las ingenierías o tecnicaturas en electricidad, por tanto, este trabajo representa una herramienta de soporte para determinar los componentes de un circuito mediante la utilización de un prototipo programado en Python, con diversas aplicaciones potenciales.

Dado lo anterior, el prototipo desarrollado corresponde a un dispositivo de detección dactilar para el reconocimiento de los componentes de un circuito eléctrico para estudiantes no videntes. Este dispone de una cámara web en un pedestal a una altura adecuada para enfocar las imágenes y la mano de los usuarios. El prototipo consta de un software desarrollado en Python3, con lenguaje de código legible y con módulos principales OpenCV y MediaPipe, que permiten reconocer imágenes de componentes de un circuito eléctrico, como resistencias, capacitores, inductores y fuentes de alimentación y la posición de la mano sobre estos elementos. También dispone de un módulo de transformación texto-audio (gTTS) para emitir la información correspondiente al componente electrónico señalado, para lo que se utilizará como *hardware* una tarjeta Nvidia JETSON NANO.

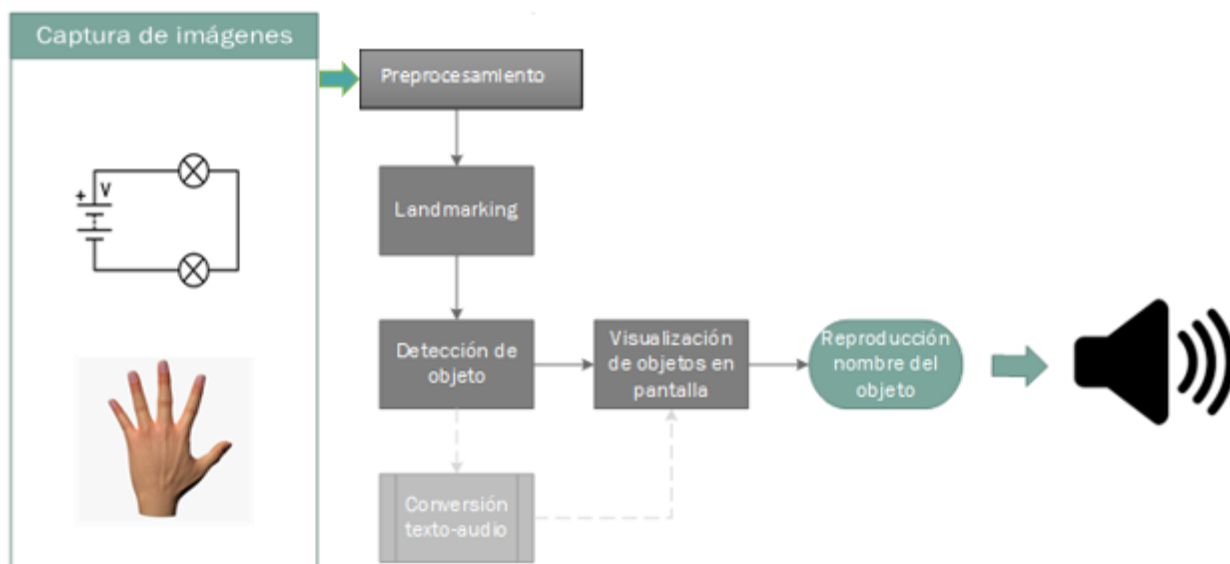
Para desarrollar el prototipo, se ocupó estudios realizados aplicados en el enfoque cuantitativo, de alcance aplicado y de corte transversal. El enfoque cuantitativo

corresponde a uno que se vale de métodos matemáticos y estadísticos para estudiar las características de una determinada población, ya sea directamente o a través del estudio en una muestra representativa (Hernández, Fernández, & Baptista, 2014). Por otra parte, el alcance de la investigación es aplicado, pues una vez estudiada la población de estudio se intervendrá directamente en la problemática mediante la propuesta de una solución. Por último, la investigación es de corte transversal, ya que se desarrolla en determinado periodo y no tiene por objetivo desarrollar un estudio temporal (Hernández, Fernández, & Baptista, 2014).

Diagrama general del sistema

Figura 18

Diagrama de bloques del sistema



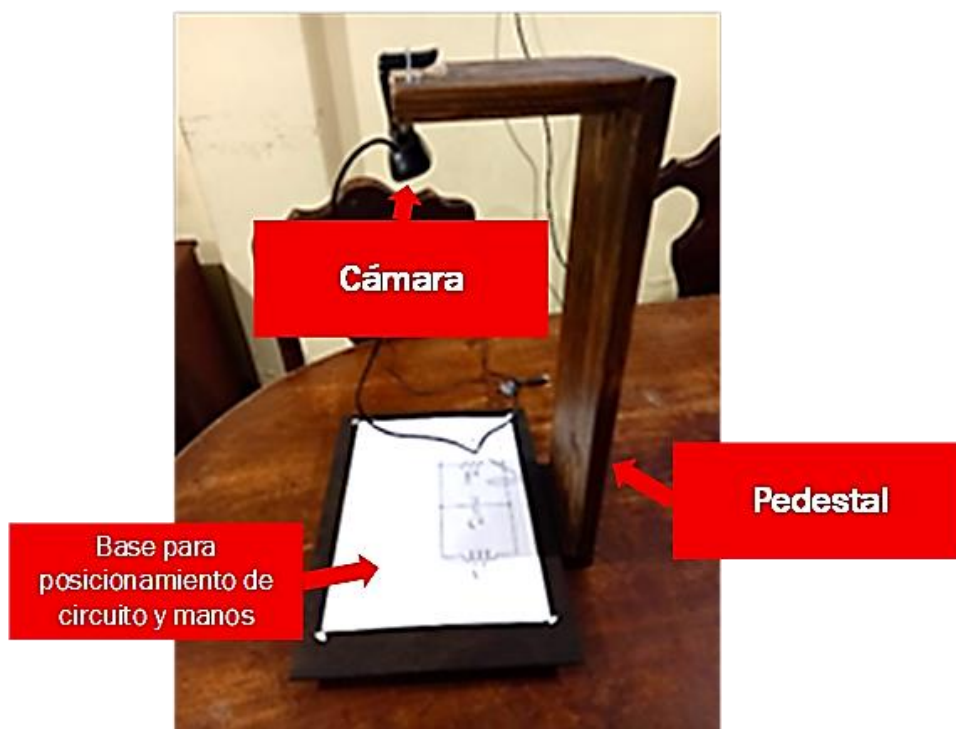
El código fuente completo del prototipo puede revisarse en detalle en el **¡Error! No se encuentra el origen de la referencia.** A continuación se describen cada una de las fases del sistema.

Captura

El dispositivo comienza a operar con la captura de una imagen con una cámara la cual está ubicada en un pedestal a 47 cm de su base de madera de 35 cm de ancho x 26 cm de largo, obteniendo la posición de la mano y de un determinado componente señalado sobre el circuito eléctrico.

Figura 19

Soporte del prototipo



En el pedestal (Figura 19) se ha dispuesto la cámara web (Figura 20 y Tabla 6), que permite la captura de imagen. En la base debe ubicarse el objeto a ser reconocido, en este caso un diagrama del circuito eléctrico.

Figura 20

Cámara



Tabla 6

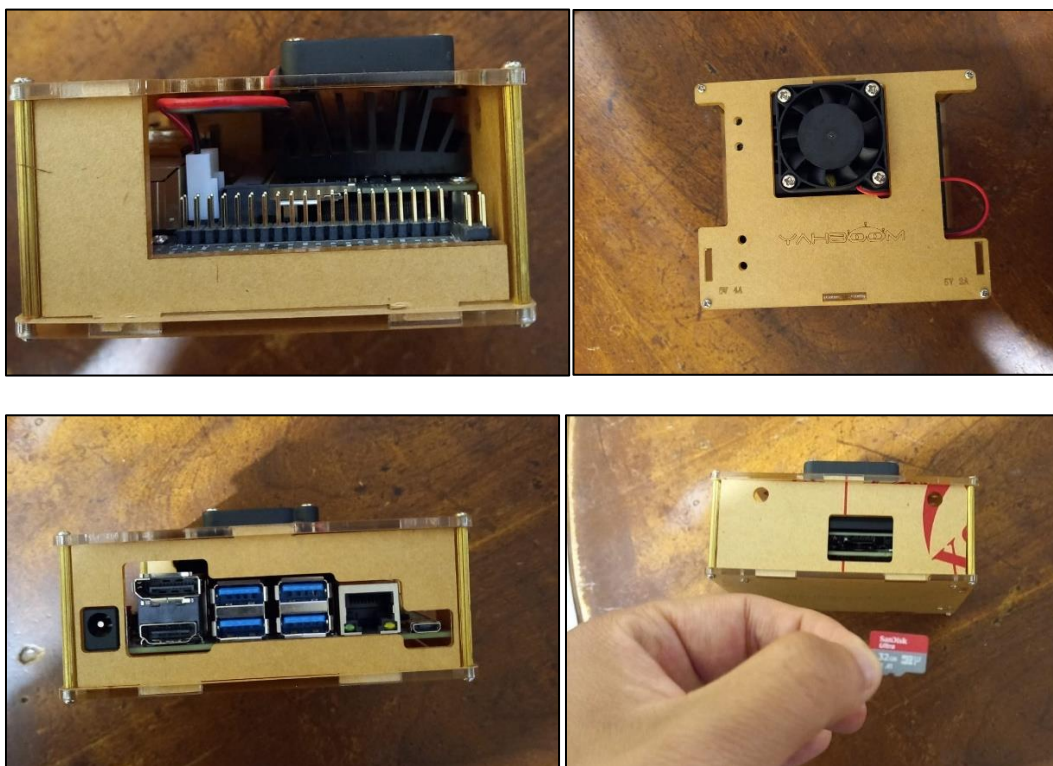
Características técnicas de la Cámara Web Logitech HD C270

Resolución	Campo visual diagonal	Iluminación automática	Micrófono	Conexión
HD 720p/30 FPS	55°	RightLight 2	Omnidireccional	Plug and Play USB-A

Fuente: (Logitech, 2022)

Preprocesamiento

En este bloque se describe el procesamiento que se realiza sobre la imagen capturada por la cámara web. Como hardware, se utilizó el Kit de procesamiento NVIDIA Jetson Nano (Figura 21), que facilita la programación y entrenamiento de RNA paralelas para diversas aplicaciones, entre las que constan la identificación y clasificación de imágenes, detección de objetos y gestos y procesamiento del habla.

Figura 21*Tarjeta Jetson Nano***Tabla 7***Características tarjeta JETSON NANO*

RAM	GPU	Núcleos	Frecuencia	Discos	Sistema operativo
4 GB	ARMv8 <i>processor</i> rev 1	4	1479 MHz por núcleo	64 GB	Ubuntu Server 16

Para el funcionamiento óptimo de la tarjeta, es preciso utilizar un jumper para la fuente de alimentación que se lo coloca en el lado donde se encuentra los pines de la

tarjeta, los dos pines que esta separados del resto ahí se coloca el jumper como se muestra en la siguiente figura (Figura 22), lo que garantiza un funcionamiento con el máximo rendimiento.

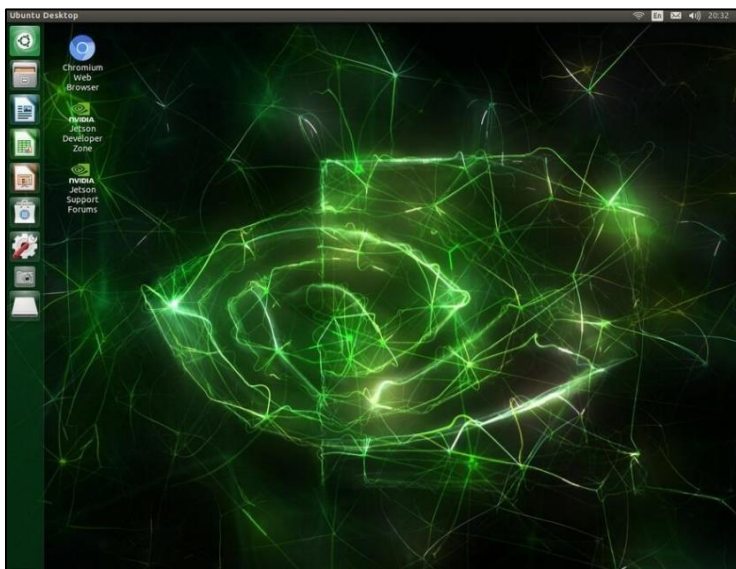
Figura 22

Jumper para fuente de alimentación



Figura 23

Escritorio JETSON NANO



De esta manera, para una implementación inicial, es preciso llamar a las bibliotecas a utilizar durante el proceso, entre las que tenemos cv2 el cual nos ayuda a utilizar la visión por computadora para el análisis y tratamiento de imágenes para detectar rostros, manos, objetos, etc., mediapipe el cual nos ayudara con la detección de las manos mediante su función hands, gtts el cual nos permite convertir un texto escrito en audio con la ayuda del traductor de google y por ultimo playsound el cual nos ayuda a reproducir un audio en formato .mp3 que se encuentra en la misma carpeta que el código:

```
import cv2
import mediapipe as mp
from gtts import gTTS
from playsound import playsound
```

De la biblioteca llamamos a dos funciones que vamos a utilizar para la detección de manos que es `mp_hands = mp.solutions.hands` el cual nos permite obtener la posición de la mano en toda la imagen limitándola en un recuadro , `mp_drawing = mp.solutions.drawing_utils` esta función nos ayudara para dibujar los 21 puntos encontradas en la mano y sus conexiones.

```
_hands = mp.solutions.hands
_drawing = mp.solutions.drawing_utils
```

Las imágenes que son capturadas a través de la cámara Logitech C270 con una resolución HD de 720p y utilizando 30 fps (Fotogramas por Segundo), las cuales son sometidas a un preprocesamiento que se realiza con el funcionamiento de la tarjeta Jetson Nano. Para comenzar el proceso, se da inicio a la captura de imagen a través de la cámara con el comando `cap = cv2.VideoCapture(0)` el parámetro 0 significa que va a utilizar la cámara conectada a la tarjeta, se ingresan los datos mediante los cuales se reconoce la palma de la mano, partiendo por el reconocimiento de una transmisión en vivo, se ingresan 3 parametros en la función `mp_hands.Hands`, la

primera es `static_image_mode=False` el parámetro `False` indica que los parámetros de entrada son de un video en vivo, si el parámetro fuera `True` indicaría al programa que se trabajaría con imágenes estáticas, `max_num_hands` indica el número máxima de manos por detectar, `min_detection_confidence=0.5` es el número mínimo de confianza para que la detección sea exitosa que varía entre 0 a 1, se utiliza el número de 0.5 para que la latencia del programa no sea tan alto y trabaje sin ningún problema:

```
with mp_hands.Hands(
    static_image_mode=False,
    max_num_hands=2,
    min_detection_confidence=0.5) as hands:
```

Con el uso de la función *while* es posible definir la transmisión continua de la imagen captada en la cámara (salvo en caso de interrupciones), con el parámetro `True` indicamos que no hay ninguna interrupción y continua la transmisión en vivo, en caso de que exista alguna interrupción el parámetro pasa a `False` e interrumpe el video en vivo. De igual manera, obtenemos las dimensiones de la imagen (ancho y altura) parámetros que se utilizaran más adelante en el código y orientarla con `cv2.flip(image, 1)` para que sea visualizada de forma correcta, ya que la imagen ingresada es por defecto una imagen refleja, el parámetro 1 indica que la imagen es girada horizontalmente y el parámetro 0 indica que la imagen es girada verticalmente.

```
while True:
    ret, image = cap.read()
    if ret == False:
        break
    height = image.shape[0]
    width = image.shape[1]
    image = cv2.flip(image, 0)
    image = cv2.flip(image, 1)
```

De igual manera, realizamos la binarización de la imagen recibida en la cámara que la detecta con el estándar BGR, es precisa su conversión BGR a RGB debido a que las detecciones se realizan con imágenes RGB , mediante

```
results = hands.process(cv2.cvtColor(image,
cv2.COLOR_BGR2RGB))
```

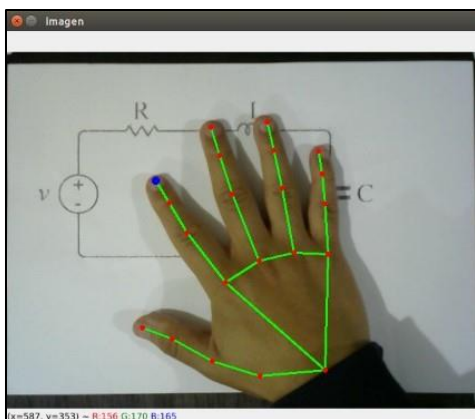
Detección de la mano (Landmarking)

Mediapipe, framework multimodal de código abierto, permite la detección de la mano junto con OpenCV. Se utilizó la mayoría del código proporcionado en Python, sobre el cual se realizaron leves modificaciones de adaptación a las necesidades particulares de los objetos a reconocer.

Para detectar en concreto la mano, se utilizan las dos funciones de Mediapipe: `mp_drawing = mp.solutions.drawing_utils` que permite dibujar los 21 puntos en la mano y la función `mp_hands = mp.solutions.hands` que permite localizar la palma de la mano (Figura 24) en cualquier ubicación dentro de la visión de la cámara que esta tenga y de sus dimensiones.

Figura 24

Detección de los puntos de la mano



El reconocimiento de los gestos de la mano se lleva a cabo en el proceso de reconocimiento de *landmarks* o esqueletización, que corresponden a un conjunto de marcas para referenciar la mano. Esto se lleva a cabo con la función `if`

`results.multi_hand_landmarks is not None`: esta línea de código nos indica que si detecta una mano puede continuar, caso contrario no continúa hasta que encuentre una mano, luego se llaman a subfunciones que me permite dibujar los 21 puntos de la mano y sus conexiones, mediante:

```
for hand_landmarks in results.multi_hand_landmarks:
    for hand_landmarks in
results.multi_hand_landmarks:
```

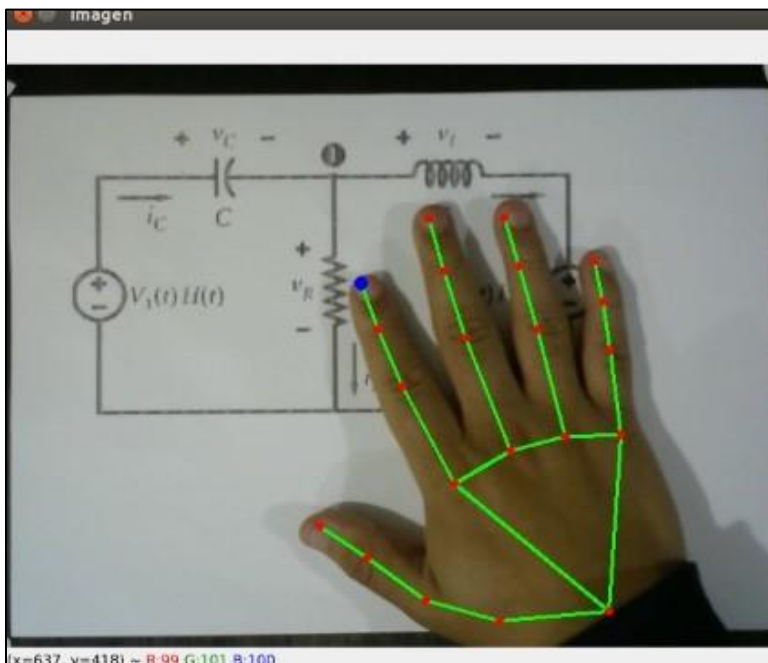
Se coloca 3 parámetros para dibujar la esqueletización de la mano, `image` que indica la imagen de la mano detectada, `hand_landmarks` dibuja los 21 puntos de la mano, `mp_hands.HAND_CONNECTIONS` une los 21 puntos de la mano mediante conexiones creando la esqueletización de la mano, mediante:

```
mp_drawing.draw_landmarks(
    image, hand_landmarks,
    mp_hands.HAND_CONNECTIONS)
```

Como se observa en la siguiente figura, el sistema es capaz de reconocer en la mano el dedo en particular que funciona como cursor. Este se indica en azul, a diferencia de los otros puntos (Figura 25).

Figura 25

Visualización en pantalla de la mano e identificación del dedo cursor



Ingresando:

```

x1 =
int(hand_landmarks.landmark[mp_hands.HandLandmark.INDEX_FINGER_TIP].x * width)
y1 =
int(hand_landmarks.landmark[mp_hands.HandLandmark.INDEX_FINGER_TIP].y * height)
cv2.circle(image, (x1, y1), 3, (255, 0, 0), 3)

```

se obtienen en particular las coordenadas de la punta del dedo que en particular realizará la identificación del circuito. Esto se realiza utilizando los nombres establecidos por MediaPipe, utilizamos `INDEX_FINGER_TIP` que corresponde al dedo índice de la mano, accedemos a la coordenada x multiplicando por el ancho de la imagen y la coordenada y multiplicando por el alto de la imagen, obteniendo las coordenadas del dedo índice (tanto x como y), y se termina por cambiar el color de la

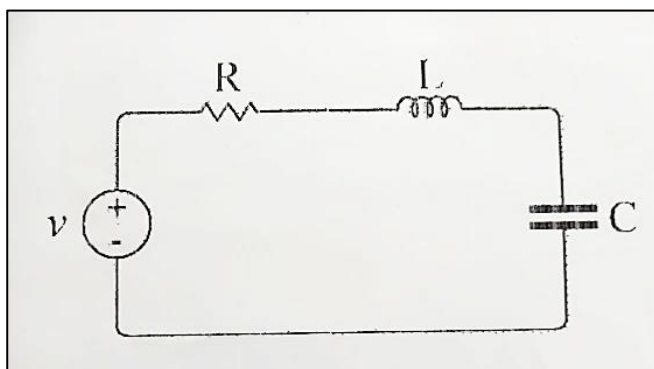
punta del dedo que vamos a utilizar por el color azul y cambiando el grosor del círculo para que se vea en la imagen en vivo cual dedo se está utilizando.

Detección del elemento (circuito)

La librería OpenCV permite la detección de objetos bajo tecnología de visión artificial, entre los que pueden contar distintas figuras geométricas, rostros, entre otros. En este caso, se adecuó para el reconocimiento de los distintos elementos de un circuito. Un ejemplo de circuito creado para el reconocimiento y sus componentes se observa a continuación (Figura 26)

Figura 26

Circuito ejemplo para el reconocimiento del sistema


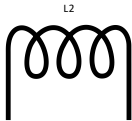

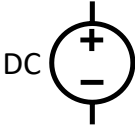


R=Resistencia

L=Inductor

C=Capacitor

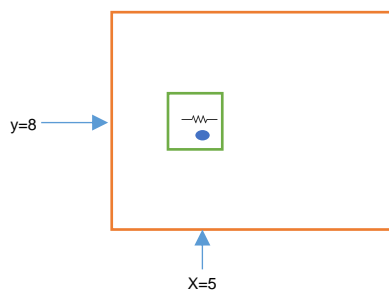
Tabla 8*Elementos del circuito reconocidos por el sistema*

Elementos	Símbolo(s)
Resistencia	
Inductor o bobina	
Capacitor	
Fuente	

Posteriormente, es preciso determinar de la misma manera la posición de los circuitos y sus componentes, la cual detallaremos en el siguiente ejemplo:

Figura 27

Ejemplo para el reconocimiento de coordenadas en el sistema



El cuadro tomate representa la visión total que tiene la cámara de video, el cuadro verde representa el componente eléctrico que se encuentra situado en el circuito eléctrico y como se observa en la parte posterior de la (Figura 25) se sabe con exactitud sus coordenadas (tanto x como y), dentro del cuadro se encuentre el componente eléctrico y el punto en azul representa la señalización de la punta del dedo índice de la mano en el circuito eléctrico, para el ejemplo se observa que el componente eléctrico se encuentra ubicado en las coordenadas $x = 5$ y $y = 8$, entonces cuando mi dedo este señalando esa coordenada en específico, el sistema dará una aviso por medio de un audio del objeto que se este tocando, en este ejemplo seria la resistencia.

En el sistema se realizó un rango de coordenadas tanto en x como en y debido al tamaño de cada componente eléctrico en cada circuito probado en el sistema, dado que van a tener diferente tamaño dependiendo cuantos elementos tenga el circuito, como se observa en la (Figura 29) solo hay dos elementos así que su tamaño es mayor, pero en la (Figura 33) hay 4 elementos entonces su tamaño va a ser menor y así cambiarían el rango de coordenadas por cada circuito probado, los rangos se expresan mediante.

```
if x1 > 65 and x1 < 67 and y1 > 147 and y1
< 176:
```

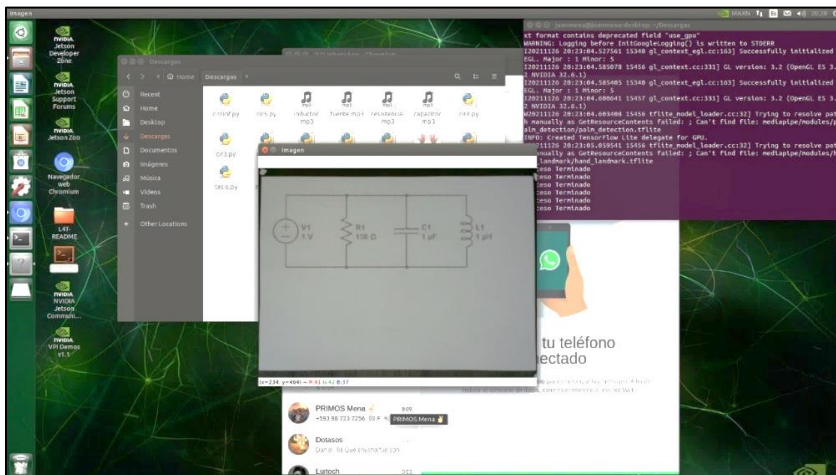
Visualización del objeto en pantalla

Para comprobar el funcionamiento del sistema se necesita visualizarlo para ello se utiliza la función `cv2.imshow('imagen', image)` que me indica en la pantalla el procesamiento del código y así verificar que esté ejecutando correctamente (

Figura 28).

Figura 28

Visualización en pantalla de un circuito



Reproducción del nombre del objeto

Finalmente, cuando se conoce el tamaño fijo del circuito y la posición de los elementos se programa la posición exacta de cada componente para que cuando el dedo pase por esa coordenada en específico pueda reconocer el elemento, se utiliza la función `if` para indicar la posición exacta que se encuentra cada elemento, con cada elemento que reconoce el sistema se almacenara en una variable tipo ("string") para cada componente del circuito eléctrico y al momento que esta el dedo sobre un determinado elemento se coge el texto de ese componente y se envía a la librería `gTTS` que nos permite realizar la transformación de texto a audio utilizando el Traductor de Google, esta prueba se realiza una sola vez ya que el archivo de audio queda grabado, para ello se necesitó poner de parámetros el idioma que vamos a utilizar, el texto el que vamos a transformar y la velocidad del audio transformado con el parámetro `slow`, en este proyecto no fue necesario utilizar este parámetro, debido que la velocidad de reproducción del audio fue la adecuada. Utilizamos la función `.save` que permite

guardar el audio una vez descargado, de manera que una vez ejecutada ya no sea necesario acceder a la red para obtener el audio. Este audio es almacenado en la carpeta donde se aloja el código, de modo que es a esta carpeta a la que se accede posteriormente para reproducirlo, se utiliza la librería `playsound` para reproducir el audio transformado y como último paso se muestra un mensaje (“Proceso Terminado”) para indicar que el sistema se ejecutó correctamente sin ningún error, este procedimiento se lo realiza mediante las siguientes líneas de código:

```

if x1 > 65 and x1 < 67 and y1 > 147 and y1
< 176:
    texto = "Esto es una fuente"
    lenguaje = 'es'

    audio = gTTS(text = texto, lang = lenguaje,
slow = False)
    audio.save('fuente.mp3')
    s = 'fuente.mp3'
    playsound(s)
    print("Proceso Terminado")
elif x1 > 201 and x1 < 206 and y1 > 145 and y1 <
172:
    texto = "Esto es un resistencia"
    lenguaje = 'es'

    audio = gTTS(text = texto, lang = lenguaje,
slow = False)
    audio.save('resistencia.mp3')
    s = 'resistencia.mp3'
    playsound(s)
    print("Proceso Terminado")
elif x1 > 337 and x1 < 340 and y1 > 137 and y1 <
148:
    texto = "Esto es un capacitor"
    lenguaje = 'es'

    audio = gTTS(text = texto, lang = lenguaje,
slow = False)
    audio.save('capacitor.mp3')
    s = 'capacitor.mp3'
    playsound(s)
    print("Proceso Terminado")
elif x1 > 479 and x1 < 482 and y1 > 147 and y1 <
179:
    texto = "Esto es un inductor"
    lenguaje = 'es'

```

```
slow = False)

audio = gTTS(text = texto, lang = lenguaje,

audio.save('inductor.mp3')
s = 'inductor.mp3'
playsound(s)
print("Proceso Terminado")
```

Capítulo IV

Implementación y resultados

Introducción

En el presente capítulo se expone la implementación del sistema diseñado y el resultado de las pruebas. Estas pruebas son, por una parte, una evaluación del funcionamiento del prototipo, la que se realizó con seis usuarios con distintos grados de discapacidad visual de la asociación de invidentes “Milton Vedado”, ubicado en la ciudad de Quito en la Av. Gral. Ignacio de Veintimilla frente al Cuerpo de Bomberos de la Veintimilla. Los usuarios que participaron testeando el prototipo se identifican en la Tabla 9.

Tabla 9

Identificación de usuarios

Id.	Edad	Grado de discapacidad visual (25, 50, 75 o 100%)	Nro. Pruebas realizadas
U 1	55	100	2
U 2	50	82	3
U 3	42	100	2
U 4	52	85	2
U 5	40	78	2
U 6	58	89	2

Primero se creó imágenes de alto relieve en que cada uno de los elementos fueron resaltados con un rectángulo alrededor de cada elemento, también se resaltó los hilos eléctricos que conectan cada elemento para que las personas no videntes tengan una guía de donde moverse en el circuito, de cada elemento para determinar si mi sistema es útil o no lo que se hizo es resaltar no el elemento, si no un recuadro alrededor de cada elemento de tal manera que ese recuadro sirva como guía para el no vidente y de esa forma se puede comprobar que se reconoce el elemento no por la figura en alto relieve de cada elemento, sino puede reconocer el elemento por la ayuda de mi sistema sabiendo la posición en la que se encuentra el dedo índice de la persona no vidente

Se tomó todas las medidas de bioseguridad correspondientes para realizar las pruebas, se les brindó una pequeña inducción previa a la electrónica, explicando los componentes eléctricos que se va a reconocer, una definición de cada elemento y cuál es su función en la electrónica, se les indicó imágenes en alto relieve de cada uno de los componentes eléctricos para que tengan conocimiento de cómo se les representan simbólicamente. Luego se les explico como se van a realizar las pruebas, primero se les indicó que van a tocar 5 circuitos diferentes como se observan en las figuras 31 a 35, los cuales tienen 4 o menos elementos que pueden ser: resistencia, capacitor, bobina, inductor y fuente, se les informo que los 5 circuitos los cuales se realizaran las pruebas son de 3 diferentes tipos: serie, paralelo y mixto.

Por último, se realizó un entrenamiento de lo que van a realizar para que sepan cómo tiene que moverse con la mano en el circuito para poder identificar los componentes, como tiene que posicionarse frente al pedestal (29 y 30) y se le indicó el audio de cada elemento para que se acostumbren al sonido de cada componente, se les indicó luego se procederá con las pruebas con cada uno de los integrantes de la

fundación, de cada circuito se les realizara una preguntas como se muestra en el (Anexo 2), luego de terminar las pruebas con todas las personas presentes se les hará una encuesta de conformidad como se muestra en el (Anexo3) y se preguntara por alguna mejora que crean conveniente realizar en trabajos futuros.

Figura 29

Posicionamiento del usuario en el prototipo

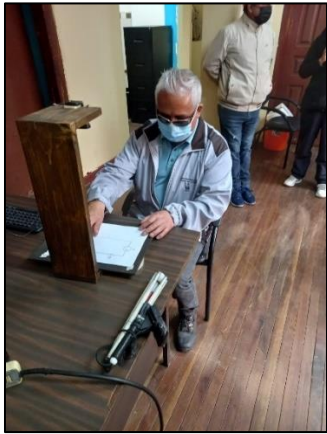


Figura 30

Indicaciones de uso del dispositivo



Prueba 1: Evaluación del funcionamiento del sistema

Esta prueba consistió en el uso del sistema para el reconocimiento de cinco tipos de circuitos y sus partes mediante el uso de la mano sobre la imagen del circuito como guía. Estos circuitos se observan en las figuras 31 a 35.

Figura 31

Visualización y reconocimiento en circuito 1

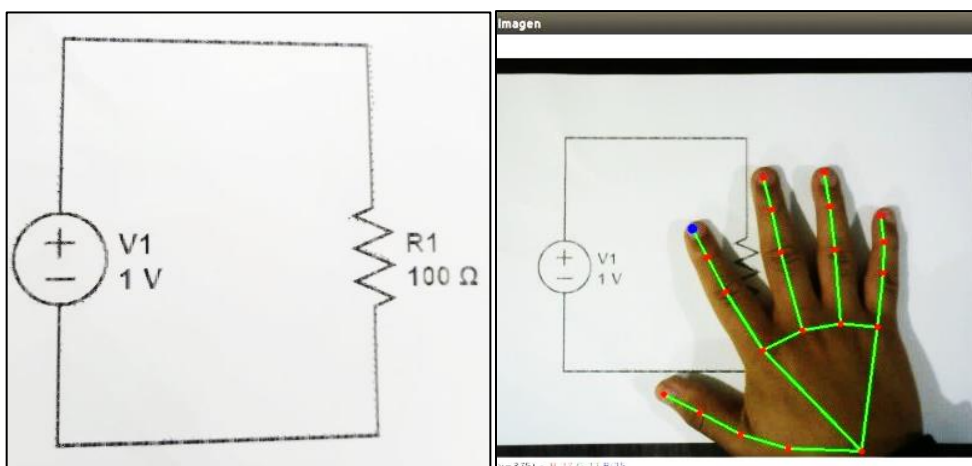


Figura 32

Visualización y reconocimiento en circuito 2

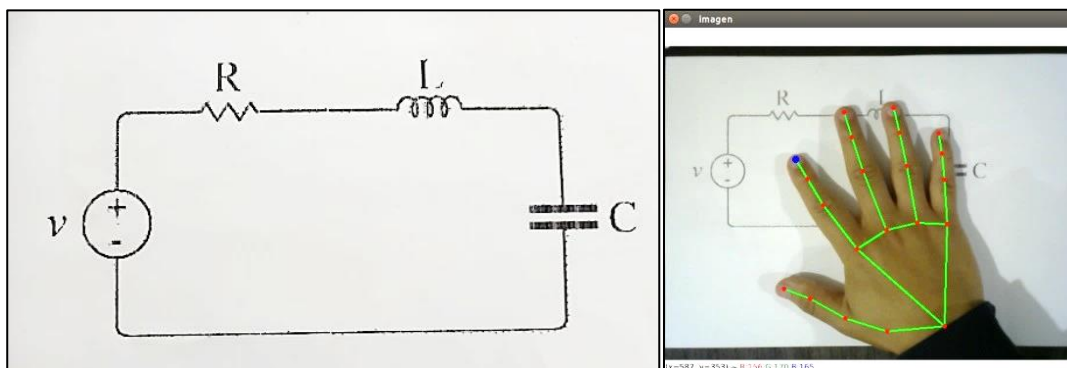
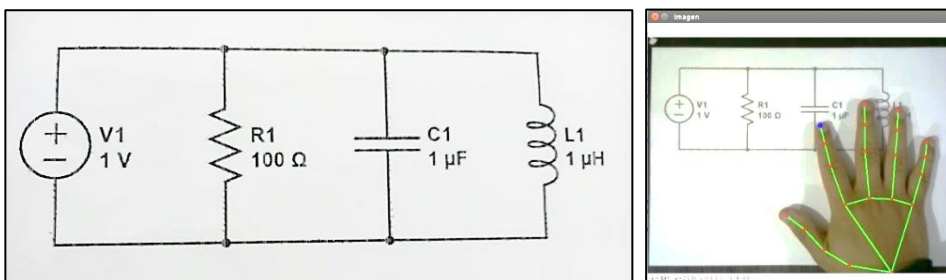
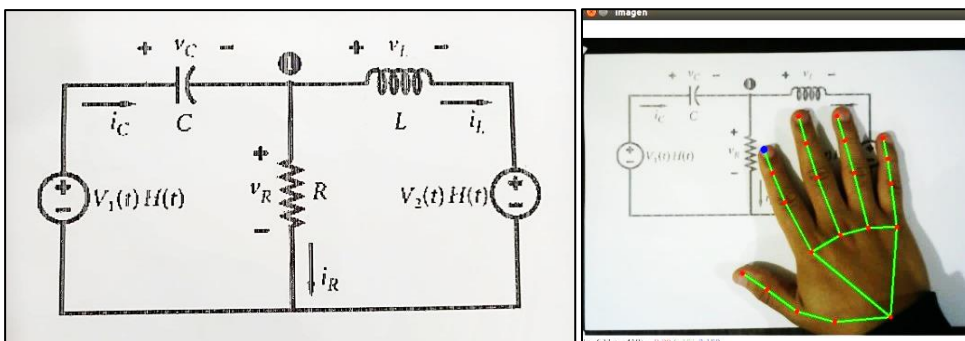
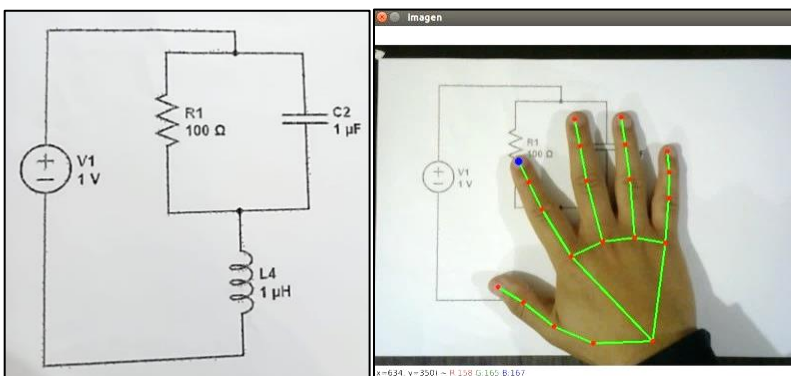
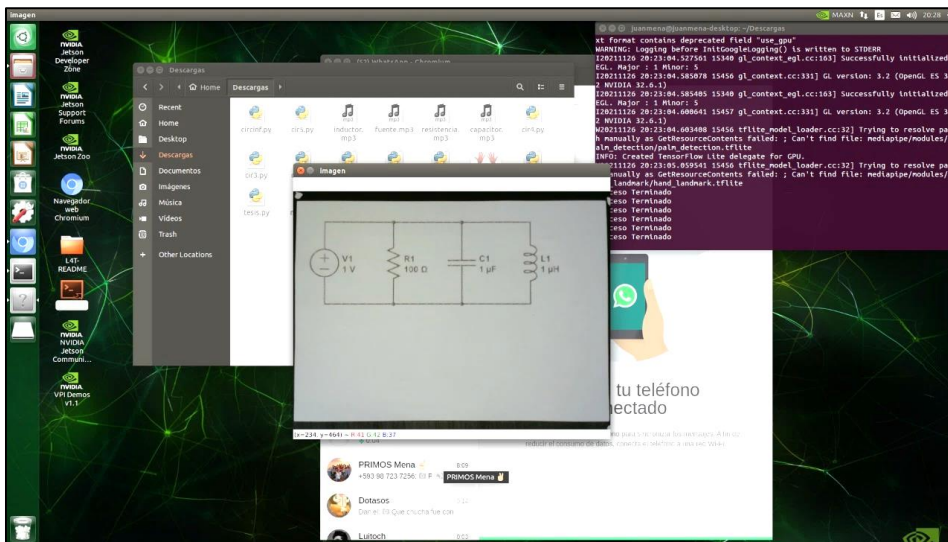


Figura 33*Visualización y reconocimiento en circuito 3***Figura 34***Visualización y reconocimiento en circuito 4***Figura 35***Visualización y reconocimiento en circuito 5*

Las imágenes procesadas se visualizaron de la siguiente manera en el escritorio de la tarjeta Nvidia JETSON NANO (36).

Figura 36

Visualización de objetos en pantalla



El dispositivo fue probado por turnos. Cada uno de los seis participantes pudo reconocer en los circuitos evaluados los elementos que lo componen a partir de la escucha del audio (Figura 37).

Figura 37

Testeo de usuarios



Se puede verificar videos de las pruebas que se realizaron en el siguiente código QR (Figura 38).

Figura 38

Código QR videos de implementación



Nota. Se puede acceder al video en el siguiente enlace <https://youtu.be/S7TtFmjEn9o>

Resultados de evaluación de funcionamiento

Una vez terminada las pruebas técnicas se realizó una encuesta que determine si lograron identificar todos los componentes eléctricos del circuito para ello preguntó a los usuarios por cada uno de los elementos del circuito que fueron capaces de reconocer con una encuesta (ver anexo 2). En la siguiente tabla se va a registrar los valores registradas por los usuarios siendo C1(Circuito 1), C2(Circuito 2), C3(Circuito3), C4(Circuito 4) y C5(Circuito 5); la prueba consistió en que cada usuario va identificar 5 circuitos, por cada circuito se realiza una encuesta para determinar cuántos bobinas, resistencias, capacitores y fuentes identifico, cuantos elementos totales identifico en el circuito y cuantas veces necesito realizar la prueba para poder contestar las preguntas

antes mencionadas, obteniendo los siguientes resultados que se muestran en la (Tabla 10)

Tabla 10

Encuesta de evaluación de resultados del sistema

Usuario 1		Circuitos				
Preguntas		C 1	C 2	C 3	C 4	C 5
1	¿Cuántos elementos tiene el circuito eléctrico?	2	3	4	5	4
2	¿Cuántos capacitores tiene el circuito eléctrico?	0	1	1	1	1
3	¿Cuántas resistencias tiene el circuito eléctrico?	1	1	1	1	1
4	¿Cuántos inductores tiene el circuito eléctrico?	0	0	1	1	1
5	¿Cuántas fuentes tiene el circuito eléctrico?	1	1	1	1	1
6	¿Cuántos intentos necesito para identificar los elementos del circuito eléctrico?	1	1	2	2	1

Usuario 2						
Preguntas		C 1	C 2	C 3	C 4	C 5
1	¿Cuántos elementos tiene el circuito eléctrico?	2	3	4	5	4
2	¿Cuántos capacitores tiene el circuito eléctrico?	0	1	1	1	1
3	¿Cuántas resistencias tiene el circuito eléctrico?	1	1	1	1	0
4	¿Cuántos inductores tiene el circuito eléctrico?	0	1	1	1	2

5	¿Cuántas fuentes tiene el circuito eléctrico?	1	0	1	2	1
6	¿Cuántos intentos necesito para identificar los elementos del circuito eléctrico?	1	3	1	2	2

Usuario 3						
Preguntas		C 1	C 2	C 3	C 4	C 5
1	¿Cuántos elementos tiene el circuito eléctrico?	2	4	4	5	4
2	¿Cuántos capacitores tiene el circuito eléctrico?	0	1	1	1	1
3	¿Cuántas resistencias tiene el circuito eléctrico?	1	1	1	1	1
4	¿Cuántos inductores tiene el circuito eléctrico?	0	1	1	1	1
5	¿Cuántas fuentes tiene el circuito eléctrico?	1	1	1	2	1
6	¿Cuántos intentos necesito para identificar los elementos del circuito eléctrico?	1	2	1	3	1

Usuario 4						
Preguntas		C 1	C 2	C 3	C 4	C 5
1	¿Cuántos elementos tiene el circuito eléctrico?	2	4	3	5	4
2	¿Cuántos capacitores tiene el circuito eléctrico?	0	2	1	1	0
3	¿Cuántas resistencias tiene el circuito eléctrico?	1	1	1	1	2
4	¿Cuántos inductores tiene el circuito eléctrico?	0	0	0	1	1
5	¿Cuántas fuentes tiene el circuito eléctrico?	1	1	1	2	1

6	el circuito eléctrico? ¿Cuántos intentos necesito para identificar los elementos del circuito eléctrico?	1	2	2	2	2
---	---	---	---	---	---	---

Usuario 5						
Preguntas		C 1	C 2	C 3	C 4	C 5
1	¿Cuántos elementos tiene el circuito eléctrico?	2	4	4	5	4
2	¿Cuántos capacitores tiene el circuito eléctrico?	0	1	1	1	1
3	¿Cuántas resistencias tiene el circuito eléctrico?	1	1	1	1	1
4	¿Cuántos inductores tiene el circuito eléctrico?	0	1	1	1	1
5	¿Cuántas fuentes tiene el circuito eléctrico?	1	1	1	2	1
6	¿Cuántos intentos necesito para identificar los elementos del circuito eléctrico?	1	2	2	2	2

Usuario 6						
Preguntas		C 1	C 2	C 3	C 4	C 5
1	¿Cuántos elementos tiene el circuito eléctrico?	2	4	4	5	4
2	¿Cuántos capacitores tiene el circuito eléctrico?	0	1	1	1	1
3	¿Cuántas resistencias tiene el circuito eléctrico?	1	1	1	1	1
4	¿Cuántos inductores tiene el circuito eléctrico?	0	1	1	1	1
5	¿Cuántas fuentes tiene el circuito eléctrico?	1	1	1	2	1

6	¿Cuántos intentos necesito para identificar los elementos del circuito eléctrico?	2	2	2	2	2
---	---	---	---	---	---	---

Como se observa en la Tabla 10, la mayoría de los usuarios reconoció adecuadamente los elementos del sistema. Respecto al número de elementos reconocidos, los usuarios 3, 5 y 6 reconocieron la totalidad de los elementos de los circuitos en todas las pruebas realizadas, mientras que los usuarios 1 y 2 reconocieron todos los componentes, salvo en el circuito nro. 2, para el que solo reconocieron 3 de sus componentes. Por último, el usuario 4 solo presentó problemas con el circuito nro. 3.

Respecto a los capacitores, todos los usuarios indicaron que este se encuentra presente en los circuitos 2, 3, 4 y 5 en una unidad, lo que corresponde efectivamente con los circuitos analizados. Solo el usuario 4 tiene dificultades para reconocer los capacitores en las pruebas, indicando más de uno en el circuito 2 y ninguno en el circuito 5.

El número de resistencias identificadas también se mostró en su mayoría correcto en todas las pruebas. Los usuarios 1, 3, 5 y 6 identificaron una resistencia en todos los circuitos. El usuario 2 no identificó resistencias en el circuito nro. 5 y el usuario 4 identificó dos resistencias en este mismo circuito.

Para el caso de los inductores, los resultados muestran igualmente una tendencia positiva. Los usuarios 3, 5 y 6 respondieron que en los circuitos 2 a 5 existe un inductor, mientras que para los usuarios 1, 2 y 4 se presenta la información distinta. Estos reconocen de forma heterogénea los inductores: para el usuario 1 solo hay

inductores en los circuitos 3 a 5; para el usuario 2 hay más de un inductor en el circuito 5 y para el usuario 4 solo se presentan inductores en los circuitos 4 y 5.

En lo que respecta a las fuentes en los circuitos, los usuarios 3, 4, 5 y 6 responden de la misma manera, indicando que en todos existe una fuente, salvo en el circuito 4, en que se presentan 2. Por el contrario, los usuarios 1 y 2 muestran resultados disímiles: mientras que el usuario 1 encuentra para todos los circuitos una fuente, salvo en 3 y 4 que evidencia 2, mientras que el usuario 2 considera que el circuito nro. 2 no tiene fuente alguna.

Por último, muchos usuarios realizaron una prueba o necesitaron una segunda oportunidad para moverse en el circuito al cual se lo denomina como intento, el usuario 1 declara necesitar un intento para todos los circuitos salvo el 2 y el 4. El usuario 2 considera que para los circuitos 1 y 3 necesitó solo un intento, dos intentos para el 4 y el 5, y tres intentos para el circuito 2. El usuario 3 necesitó un intento para los circuitos 1, 3 y 5, mientras que para el circuito 2 y 4 necesitó dos y tres intentos, respectivamente. Los usuarios 4 y 5 necesitaron en su mayoría dos intentos (en los circuitos 2 a 5), mientras que solo uno para el primer circuito. Por último, el usuario 6 necesitó dos intentos para cada circuito.

En términos generales se observa una buena interacción usuario-sistema por los resultados obtenidos. Cabe mencionar que hubo usuarios que realizaron comentarios del tamaño de los circuitos, del audio, de la velocidad de reproducción, pero todos esos parámetros pueden cambiarse en el software de acuerdo con la necesidad de cada usuario, el sistema tiene parámetros estándares para que todos los usuarios puedan utilizarlo, pero eso no quiere decir que se puede realizar modificaciones de acuerdo a la comodidad de cada usuario.

Por otra parte, unos usuarios dijeron que podría hacer interesante agregar una segunda mano para el reconocimiento lo cual para este prototipo no se realizó, pero en el software puede ser configurado en el momento que sea necesario.

Prueba 2: Valoración del rendimiento

Una vez terminadas las pruebas para determinada la usabilidad del sistema se hizo una encuesta de valoración del rendimiento la cual se encuentra en el (Anexo 3). Este correspondió a una escala tipo Likert de 5 alternativas, en donde

1 = Totalmente en desacuerdo

2 = Algo en desacuerdo

3 = Ni de acuerdo ni en desacuerdo

4 = Algo de acuerdo

5 = Totalmente de acuerdo

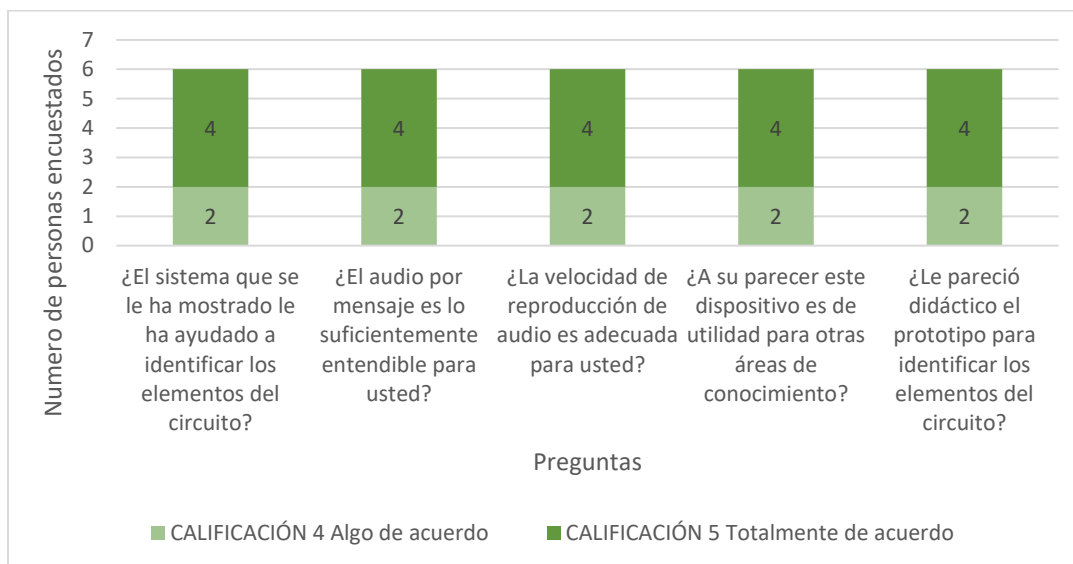
Así, las preguntas se centraron en la eficacia del sistema para la identificación de los elementos de los circuitos estudiados, en la legibilidad del audio reproducido, en su velocidad de reproducción, en la potencialidad del prototipo para su uso en otras áreas y en su funcionalidad con el objetivo de poder usar el sistema en otras áreas de conocimiento.

Resultados de valoración del rendimiento

Una vez terminada de realizar la encuesta que está en el (Anexo 3) se presentan a continuación los resultados obtenidos en la Tabla 11 y Figura 39 a continuación:

Tabla 11*Evaluación de resultados del sistema*

	PREGUNTAS	CALIFICACIÓN				
		1	2	3	4	5
		Totalmente en desacuerdo	Algo en desacuerdo	Ni de acuerdo ni en desacuerdo	Algo de acuerdo	Totalmente de acuerdo
1	¿El sistema que se le ha mostrado le ha ayudado a identificar los elementos del circuito?	0	0	0	2	4
2	¿El audio por mensaje es lo suficientemente entendible para usted?	0	0	0	2	4
3	¿La velocidad de reproducción de audio es adecuada para usted?	0	0	0	2	4
4	¿A su parecer este dispositivo es de utilidad para otras áreas de conocimiento?	0	0	0	2	4
5	¿Le pareció didáctico el prototipo para identificar los elementos del circuito?	0	0	0	2	4

Figura 39*Registro de sesión de pruebas*

Como se observa en los resultados, la evaluación es positiva. La mayoría planteó que está totalmente de acuerdo con los resultados obtenidos, y, por tanto, con que el sistema es adecuado para lo que ha sido diseñado. No obstante, dos usuarios valoraron todos los parámetros con “algo de acuerdo”, indicando que en su conjunto debe ser optimizado para ajustarse a las necesidades particulares de esta población, pero, para este prototipo se tomaron necesidades genéricas, pero las necesidades particulares pueden ser modificadas en el software.

En definitiva, el sistema puede adecuarse a diferentes características de cada una de las personas con deficiencia visual, constituyendo una solución para la problemática en los procesos de enseñanza aprendizaje que presentan las personas con discapacidad visual en el Ecuador. El sistema puede convertirse en una herramienta que aporte efectivamente para promover los aprendizajes de esta

población vulnerable, lo que coadyuva en la garantía de los derechos de las personas a la educación.

Finalmente colocamos una figura (Figura 40) en el cual se muestra el grupo de personas con las cuales se realizó las pruebas pertenecientes a la asociación de invidentes “Milton Vedado”.

Figura 40

Registro de sesión de pruebas



Capítulo V

Conclusiones y recomendaciones

Conclusiones

En esta tesis se desarrolló un prototipo que brinda accesibilidad a imágenes de circuitos eléctricos para personas con discapacidad visual mediante el reconocimiento de la posición dactilar usando procesamiento digital de imágenes mediante librerías de Python 3 ejecutados en una tarjeta Jetson NANO y este sistema va ser probado en una fundación para personas no videntes obteniendo resultados positivos.

Se revisaron trabajos previos desarrollados en torno al tema, y se encontró literatura que aportó tanto teórica como metodológicamente para el desarrollo, lo que quedó establecido en el estado del arte del presente trabajo, y permitió el desarrollo del marco teórico. Muchos de estos trabajos utilizaron también redes neuronales convolucionales mediante la librería MediaPipe, en mi trabajo se adecuó la librería MediaPipe y sus funciones consecutivas para determinar la posición de los componentes de un circuito eléctrico.

Para el desarrollo del prototipo se investigó el funcionamiento de la tarjeta Nvidia JETSON NANO, la cual presenta un óptimo desempeño para el trabajo con IA. De igual manera, fue necesario realizar una investigación sobre librerías, módulos y *frameworks* disponibles para este tipo de reconocimiento.

Así, se evidenció que tanto OpenCV como MediaPipe son dos de los componentes que se presentan como idóneos para trabajar sobre el reconocimiento de objetos y gestos de la mano.

Se determinó la posición del dedo índice de la mano sobre una imagen en tiempo real capturada por una cámara utilizando el módulo MediaPipe, que permite identificar la posición de la mano y formar un esqueleto de 21 puntos para identificar que dedo que se desea utilizar.

Una vez reconocidas las imágenes capturadas por la cámara web, el prototipo reproduce un audio de carácter descriptivo con el nombre en particular del objeto eléctrico reconocido, lo que vuelve accesible a las personas con discapacidad visual el conocimiento sobre los componentes de un circuito eléctrico.

Se implementó un prototipo que permita identificar la posición dactilar sobre la imagen de un circuito eléctrico construyendo un pedestal de 45 cm de alto con una cámara web que transmite en tiempo real las imágenes captadas. Este procesamiento está realizado en base al aprendizaje profundo de redes neuronales convolucionales, de manera que puede ser entrenado para su mejora continua.

Se realizaron pruebas del prototipo en la asociación para personas no videntes "Milton Vedado", debido a la pandemia actual solo se pudo realizar la prueba con 6 usuarios con distintos grados de discapacidad. A medida que se utilizó el prototipo, las personas no videntes fueron adquiriendo un mejor desempeño al reconocer los elementos eléctricos con la posición del dedo índice, entendiendo de mejor manera el funcionamiento del dispositivo.

De igual manera, los usuarios evaluaron de manera positiva el prototipo, e identificaron un potencial de desarrollo que puede ofrecerles un medio para adquirir aprendizajes y conocimientos con mayor facilidad en otras áreas de conocimiento en el sistema educativo.

La tarjeta Jetson NANO funciona adecuadamente para trabajos de Machine Learning debido a que posee una GPU que cuenta con 128 núcleos para poder trabajar paralelamente a nivel de hilos con múltiples programas al mismo tiempo y nos ayuda a evitar retrasos en la ejecución del sistema.

La librería MediaPipe posee las librerías y bases de datos necesarias para la identificación de la mano.

Recomendaciones

Para utilizar la tarjeta Nvidia JETSON NANO es recomendable colocar un jumper en la placa para poder utilizar la fuente, de modo de suministrar más amperaje y voltaje a la tarjeta. Esto permite el uso de los 4 núcleos disponibles y tener un menor tiempo de procesamiento al utilizar el dispositivo, obteniendo con ello el máximo rendimiento.

Los circuitos deben ser de un tamaño adecuado de tal manera de que permita mover la posición de la mano sin confundir los elementos, estos elementos deben ser colocados en alto relieve que permita que las personas no videntes puedan ubicarse de elemento a elemento dentro de un circuito.

Respecto al trabajo con personas con distintos grados de discapacidad visual, es importante desarrollar pruebas en entornos diseñados para la accesibilidad, de manera de llevar a cabo la prueba lo más adecuadamente posible. Asimismo, es preciso al realizar las pruebas respetar todas las medidas de bioseguridad correspondientes para cuidar la integridad de todas las personas presentes.

Trabajos futuros

Por otra parte, también podría constituir un enorme avance el conocer otros aspectos de desarrollo personal fundamentales que, hasta el momento, se vean dificultados en las personas con discapacidad visual. Esto permitiría diseñar con la misma tecnología otro tipo de aplicaciones que mejoren su calidad de vida de manera sustancial. Por último, se sugiere para futuros estudios profundizar en las distintas librerías actuales, de modo de utilizar su potencial adecuadamente.

Se sugiere ocupar una base de datos mucha más amplia para mejorar los resultados. Utilizar una base de datos que utilice más elementos como transistores transformadores entre otros elementos de electrónica para ampliar el conocimiento de circuitos eléctricos. Además, parece pertinente el añadir otros soportes de imágenes y/o objetos, de manera que no solo puedan ser reconocidos los elementos en imágenes 2D.

Por último, se sugiere indagar más en las necesidades de la población con discapacidad visual. Ya que esta se presenta en distintos grados, es posible que se requiera de distintos tipos de herramientas que se enfoquen en particular en las dificultades que estos grupos presentan.

Bibliografía

- Acevedo, E., Serna, A., & Serna, E. (2017). Principios y características de las redes neuronales artificiales. En E. (. Serna, *Desarrollo e innovación en Ingeniería* (pág. 174). Medellín: Editorial Instituto Antioqueño de Investigación.
- Agustí, M. (2019). *Introducción a la detección de puntos característicos con OpenCV*. Universitat Politècnica de València. Recuperado de <http://hdl.handle.net/10251/123298>.
- Akanksha, D., & Dhole, A. (2018). KNN Based Hand Drawn Electrical Circuit Recognition. *International Journal for Research in Applied Science & Engineering Technology (IJRASET)*, 6(1), 1111-1115.
- Aldaz, A. (2016). *Sistema electrónico para la enseñanza del lenguaje Braille a personas invidentes*. Ambato: Universidad Técnica de Ambato.
- Arias, J., Llano, Y., Astudillo, E., & Suárez, J. (2018). Caracterización clínica y etiología de baja visión y ceguera en una población adulta con discapacidad visual. *Revista Mexicana de Oftamología*, 1(92), 201-208.
- Big Data. (15 de 11 de 2021). *Redes Neuronales Convolucionales*. Obtenido de Juanbarrios.com/: <https://www.juanbarrios.com/redes-neurales-convolucionales/>
- Cabello, M. (2019). *Circuitos eléctricos básicos I (IEI)* (Primera ed.). Madrid: Editex.
- Cachimuel, J. (2021). *Desarrollo de una aplicación para describir los caracteres matemáticos de las ecuaciones mediante audio 3D utilizando earcons y spearcons dirigida a personas con deficiencia visual*. Quito: ESPE.
- Catalán, A. (2019). Técnicas de procesamiento digital de imágenes. *Revista de Marina*, 136(969), 13-24.

- Chandan, G., Jain, A., & Jain, H. (2018). Real Time Object Detection and Tracking Using Deep Learning and OpenCV. *Proceedings of the International Conference on Inventive Research in Computing Applications (ICIRCA 2018)* (págs. 1305-1308). IEEE.
- Cinjordiz, C. (15 de Diciembre de 2015). *¿Qué es un circuito eléctrico?* Recuperado el 11 de Noviembre de 2021, de <https://www.infootec.net/que-es-un-circuito-electrico/>
- Código Fuente. (11 de 15 de 2021). *Redes neuronales profundas – Tipos y Características*. Obtenido de [CodigoFuente.org](https://www.codigofuente.org): <https://www.codigofuente.org/redes-neuronales-profundas-tipos-caracteristicas/>
- CONADIS. (30 de Septiembre de 2021). *Estadísticas de Discapacidad*. Obtenido de <https://www.consejodiscapacidades.gob.ec/estadisticas-de-discapacidad/>
- Constitución de la República del Ecuador. (2008). *Registro Oficial 449 de la Constitución de la República del Ecuador*. Montecristi: Asamblea Nacional del Ecuador.
- Cubero, E., & Poler, R. (2020). Aplicación de algoritmos de aprendizaje automático a la programación de órdenes de producción en talleres de trabajo: una revisión de la literatura reciente. *Dirección y Organización* núm. 72, DOI: <https://doi.org/10.37610/dyo.v0i72.588>.
- Dabian, D., & Peña, F. (2020). Prevalencia y causas de ceguera y discapacidad visual en Colombia. *Ciencia para la Tecnología Salud Visual y Ocular*, 18(2), 20-30.
- Doménech, A. (2020). *Sign Language Recognition. ASL Recognition with MediaPipe and Recurrent Neural Networks*. [TFM] Universitat Politècnica de Catalunya. En <https://upcommons.upc.edu/handle/2117/343984>.

- Duarte, O., & González, J. (2017). *Aplicaciones de procesamiento digital de imágenes y videos con propósitos académicos*. Lima: Universidad Nacional de Ingeniería.
- Duxans, H., & Costa, M. (2017). *Síntesis del habla*. Catalunya: Universitat Oberta de Catalunya.
- Erzin, Y., & Turkoz, D. (2016). Use of neural networks for the prediction of the CBR value of some Aegean sands. *Neural Comput. Appl.* 27(5), DOI: 10.1007/s00521-015-1943-7.
- Esparza, J. (2020). *Redes neuronales convolucionales en la detección de afecciones respiratorias de los sistemas locales de salud: una revisión sistemática entre los años 2010 y 2020*. [TFG] UPN. En <https://repositorio.upn.edu.pe/handle/11537/25838>.
- Fernández, J. (12 de 11 de 2021). *Resistencia Eléctrica*. Obtenido de FisicaLab.com: https://www.fisicalab.com/apartado/resistencia-electrica-conductor#resistencia_circuitos
- Floyd, T. (2007). *Principios de circuitos eléctricos* (Octava ed.). CDMX: Pearson Educación.
- García, V. (2019). *Adaptación del sistema de conversión de texto a voz extremo a extremo Tacotron a otras lenguas*. Bilbao: Universidad del País Vasco. Disponible en https://addi.ehu.es/bitstream/handle/10810/34823/TFM_Victor_Garcia_Romillo.pdf?sequence=3&isAllowed=y.

- González, C., & Yimes, F. (2016). *Sistema de reconocimiento gestual de lengua de señas chilena mediante cámara digital*. Pontificia Universidad Católica de Valparaíso.
- González, D. (2016). *Procesamiento de imágenes*. Salamanca: Universidad de Salamanca.
- Hernández, R., Fernández, C., & Baptista, P. (2014). *Metodología de la investigación*. México: McGrawHill Educación.
- Ichi. (10 de 11 de 2021). *Visión por computadora usando Mediapipe*. Obtenido de Ichi.Pro: <https://ichi.pro/es/vision-por-computadora-usando-mediapipe-211738590199029>
- Jones, T. (04 de 10 de 2017). *Arquitecturas de aprendizaje profundo*. Obtenido de IBM Developer: <https://developer.ibm.com/es/articles/cc-machine-learning-deep-learning-architectures/>
- Langford, J., Alencastre, L., Muñoz, L., Navarro, O., Echeverría, G., Manrique, J., & Maqueo, M. (2016). Detección y seguimiento de palmas y puntas de los dedos en tiempo real basado en imágenes de profundidad para aplicaciones interactivas. *Research in Computing Science* 114 , 137–149.
- Lirusso, F. (15 de 10 de 2021). *Circuito eléctrico, serie y paralelo*. Obtenido de Google Sites. Franco Lirusso: <https://sites.google.com/site/francolirusso/circuito-electrico-serie-y-paralelo>
- Llamazares, J., Arias, A., & Melcon, M. (2017). Revisión teórica de la discapacidad visual, estudio sobre la importancia de la creatividad en la educación. *Sophia*, 13(2), 96-109.

- Llisterri, J. (2017). *Las etapas en la conversión del texto en habla*. Barcelona: Universidad Autónoma de Barcelona.
- Loayza, M. (2016). *Investigación de un sistema electrónico con detección de obstáculos para personas no videntes*. Latacunga: ESPE.
- Logitech. (10 de 01 de 2022). *Cámara web HD C270*. Obtenido de Logitech.com: <https://www.logitech.com/es-roam/products/webcams/c270-hd-webcam.960-000947.html>
- Martínez, A., Rojas, J., Sanjuan, R., & Ochoa, M. (2021). Detección de tornillos en ensamble de ECU mediante Python y OpenCV en Raspberry Pi. *Reacción Vol. 9 Núm. 1*, Recuperado de http://reaxion.utleon.edu.mx/Art_Deteccion_de_tornillos_en_ensamble_de_ECU_mediante_Python_y_OpenCV_en_Raspberry_Pi.html.
- MediaPipe. (15 de 10 de 2021). *Inicio*. Obtenido de Mediapipe.dev: <https://mediapipe.dev/>
- Mejía, J. (2017). *Apuntes de Procesamiento Digital de Imágenes*. San Luis Potosí: UASLP.
- Microsoft. (15 de 11 de 2021). *Así es como se crea software*. Obtenido de VisualStudio.microsoft.com: <https://visualstudio.microsoft.com/es/>
- Millán, M. (2018). *Electricidad básica*. Madrid: McMillan Education.
- Ministerio de Educación Chilena. (2016). *Guía de apoyo técnico-pedagógico: Necesidades educativas especiales en el nivel de educación palvularia*. Santiago: Ministerio de Educación.

- Miqueli, M., López, S., & Rodríguez, S. (2016). Baja visión y envejecimiento de la población. *Revista Cubana de Oftalmología*, 29(3), 492-501.
- Moya, D. (2021). *Desarrollo de una herramienta para la conversión de voz a texto*. Madrid: Universidad Politécnica de Madrid. Disponible en http://oa.upm.es/66298/1/TFG_DIEGO_MOYA_CORDOBA.pdf.
- Muñoz, A. (2021). *Reconocimiento de los elementos del diagrama de un circuito eléctrico básico mediante reconocimiento de imágenes*. Quito: ESPE.
- NVIDIA. (15 de 11 de 2021). *Jetson Nano Developer Kit*. Obtenido de Developer-Nvidia.com: https://developer-nvidia.com.translate.google.com/embedded/jetson-nano-developer-kit?_x_tr_sl=en&_x_tr_tl=es&_x_tr_hl=es&_x_tr_pto=nui,op,sc
- OMES. (15 de 11 de 2021). *Como instalar MEDIAPIPE | Pytho*. Obtenido de OMES-va.org: <https://omes-va.com/como-instalar-mediapipe-python/>
- OMS. (26 de febrero de 2019). *Ceguera y discapacidad visual*. Recuperado el 10 de febrero de 2021, de <https://www.who.int/es/news-room/factsheets/detail/blindness-and-visual-impairment>
- OpenCV. (10 de 10 de 2021). *Inicio*. Obtenido de OpenCV.org: <https://opencv.org/>
- Palma, K. (2021). Una Propuesta Práctica para la Inclusión de Estudiantes Ciegos en una Carrera de Ingeniería: Un Estudio de Caso. *Rev. latinoam. educ. inclusiva vol. 15 no. 1*, DOI: <http://dx.doi.org/10.4067/S0718-73782021000100139> .
- Parzibyte. (06 de 07 de 2019). *Conversión de texto a voz (TTS) con Python y gTTS*. Obtenido de Parzibyte.me: <https://parzibyte.me/blog/2019/07/06/conversion-texto-voz-tts-python-gtts/>

- Phyton. (20 de noviembre de 2021). *Lanzamiento de Python 3.0*. Recuperado el 12 de marzo de 2021, de <https://www.python.org/download/releases/3.0/>
- Proyecto Ingeni@. (15 de 10 de 2015). *Tecnología-Proyecto Ingeni@*. Obtenido de [edu.xunta.gal](https://www.edu.xunta.gal):
<https://www.edu.xunta.gal/centros/cafi/aulavirtual/mod/page/view.php?id=25247&lang=pt>
- Rocha, A., Segundo, A., & Hernández, J. (2015). Estrategia para Detección de la Mano y Reconocimiento de Gestos con los Dedos. *Memorias del Congreso Internacional de Investigación* (pág. https://www.teczamora.mx/documentos/posgrado_investigacion/articulos/Estrategia%20para%20Detecci%C3%B3n%20de%20la%20Mano%20y%20Reconocimiento%20de%20Gestos%20con%20los%20dedos.PDF). México: Academia Journals en Tecnologías.
- Russell, R. (2018). *Redes Neuronales: Guía Sencilla de Redes Neuronales Artificiales*. Madrid: Independent Publishing Platform.
- SENPLADES. (2017). *Plan Nacional del Buen Vivir*. Quito: Secretaría Nacional de Planificación y Desarrollo .
- Three Points The School for Digital Business. (26 de Enero de 2021). *Redes neuronales: definición y características*. Recuperado el 12 de Noviembre de 2021, de <https://www.threepoints.com/int/redes-neuronales-definici%C3%B3n-caracter%C3%ADsticas>
- Troyano, J., Cruz, F., González, M., & Vallejo, C. (2018). Introducción a la Programación con Python, Computación Interactiva y Aprendizaje Significativo. *Actas de las Jenuj, vol. 3,* 223-230.

Valderrama, F., Chávez, D., Muñoz, S., Tuesta, V., & Mejía, I. (2021). Importancia de las redes neuronales artificiales en la ingeniería civil: Una revisión sistemática de la literatura. *Revista ITECKNE Vol. 18 N° 1*, DOI: <https://doi.org/10.15332/iteckne.v18i1.2542>.

Valverde, W. (2016). *Diseño de un sistema de procesamiento de imágenes mediante programación de instrumentación virtual LabView*. Guayaquil : UCSG.

Zhang, F., Bazarevsky, V., Vakunov, A., Tkachenka, A., Sung, G., Chang, C., & Grundmann, M. (2021). MediaPipe Hands: On-device Real-time Hand Tracking. *arXiv:2006.10214 (cs)*, Disponible en <https://arxiv.org/abs/2006.10214>.

Anexos