

VisualVoice - Reconocimiento de Objetos y Sintetización de Voz

Joel Juan Pablo Gramajo Chan

Facultad de Ingeniería

Universidad Mesoamericana

Quetzaltenango, Guatemala, C.A.

juanpablogramajo@umes.edu.gt

Juan Carlos Neil Palacios Escobar

Facultad de Ingeniería

Universidad Mesoamericana

Quetzaltenango, Guatemala, C.A.

juancarlosneil@umes.edu.gt

Resumen—El proyecto consiste en la creación de un dispositivo con la capacidad de reconocer al menos cinco tipos de objetos que se encuentren frente a una cámara y deberá sintetizar una voz humana que indique por medio de audio qué objeto ha reconocido y a qué distancia se encuentra aproximadamente.

Palabras clave—Sintetizar, API, Consola, Backend, Modelos, Frontend, Frameworks, Inteligencia Artificial, IoT, Machine Learning.

I. INTRODUCCIÓN

El reconocimiento de objetos es una tecnología que permite a las máquinas identificar y clasificar objetos en imágenes o vídeos. En este caso, utilizarías una cámara para capturar imágenes de los objetos frente a ella. El software de reconocimiento de objetos procesaría estas imágenes y utilizaría algoritmos de aprendizaje automático para identificar los objetos presentes en la escena.

Existen varias técnicas para el reconocimiento de objetos, pero una de las más utilizadas es el aprendizaje profundo (deep learning). Mediante el uso de redes neuronales convolucionales (CNN), estas técnicas pueden aprender características visuales de los objetos y realizar predicciones precisas sobre su identidad.

Una vez que el software haya identificado los objetos, puedes combinarlo con la síntesis de voz para proporcionar información auditiva al usuario. La síntesis de voz es una tecnología que convierte texto en voz humana. Utilizando algoritmos de procesamiento del lenguaje natural (NLP), el software puede generar una descripción verbal de los objetos reconocidos.

II. TECNOLOGÍAS UTILIZADAS

A continuación se presentan las tecnologías y conceptos utilizados para la realización de este proyecto, junto con una descripción de cada uno.

II-A. Python

Python es un lenguaje de alto nivel de programación interpretado cuya filosofía hace hincapié en la legibilidad de su código, se utiliza para desarrollar aplicaciones de todo tipo, ejemplos: Instagram, Netflix, Spotify, Panda3D, entre otros.

Administrado por Python Software Foundation, posee una licencia de código abierto, denominada Python Software Foundation License. Python se clasifica constantemente como uno de los lenguajes de programación más populares.

II-B. React JS

ReactJS es una biblioteca de JavaScript desarrollada por Facebook para construir interfaces de usuario interactivas y reactivas. Su principal enfoque es la creación de componentes reutilizables que encapsulan tanto la estructura como el comportamiento de la interfaz. Utilizando JSX, una sintaxis especial que combina JavaScript y HTML, ReactJS facilita la construcción de interfaces dinámicas y mantenibles. Con su algoritmo de DOM virtual, ReactJS minimiza las actualizaciones del DOM real, lo que resulta en una interfaz más eficiente y rápida. Además, su flujo de datos unidireccional y el uso de props permiten una comunicación efectiva entre los componentes y la reutilización lógica.

El ecosistema de ReactJS es otro aspecto destacado. Existen numerosas bibliotecas y herramientas disponibles que complementan sus capacidades, como React Router para el enrutamiento, Redux para la administración del estado y Axios para hacer solicitudes HTTP. Este ecosistema activo y en constante crecimiento amplía las posibilidades y facilita el desarrollo de aplicaciones web complejas.

II-C. OpenCV

OpenCV (Open Source Computer Vision Library) es una biblioteca de visión por computadora de código abierto ampliamente utilizada y altamente popular. Está diseñada para proporcionar una amplia gama de herramientas y algoritmos

relacionados con la visión artificial y el procesamiento de imágenes.

OpenCV es compatible con varios lenguajes de programación, como C++, Python y Java, lo que la hace accesible y adaptable a diferentes entornos de desarrollo. Ofrece una amplia gama de funciones para el procesamiento de imágenes, como manipulación, filtrado, detección de bordes, segmentación y reconocimiento de objetos.

La biblioteca también proporciona algoritmos avanzados para tareas más complejas, como reconocimiento facial, seguimiento de objetos, calibración de cámaras, reconstrucción 3D y más. Estas capacidades hacen que OpenCV sea una herramienta versátil para aplicaciones en diversos campos, incluyendo la robótica, la seguridad, la medicina, la industria automotriz y la realidad aumentada.

OpenCV se beneficia de una comunidad activa de desarrolladores y ofrece una documentación completa y ejemplos de código para facilitar su aprendizaje y uso. Además, cuenta con enlaces a otras bibliotecas populares, como NumPy y SciPy, lo que permite realizar operaciones avanzadas en matrices y análisis numérico.

II-D. Numpy

NumPy es una biblioteca de Python que se utiliza para realizar cálculos numéricos en arrays multidimensionales. Fue desarrollada para proporcionar una alternativa eficiente a las estructuras de datos nativas de Python, como las listas y las tuplas, para el manejo de grandes cantidades de datos numéricos.

Entre las principales características de NumPy se encuentran su capacidad para realizar operaciones matemáticas en arrays multidimensionales de manera eficiente, la inclusión de funciones matemáticas y estadísticas para el análisis de datos, y su integración con otras bibliotecas de Python, como Pandas y Matplotlib.

Entre los usos comunes de NumPy se incluyen la exploración y análisis de datos, la creación de modelos matemáticos y estadísticos, la implementación de algoritmos de machine learning y la visualización de datos.

II-E. TensorFlow

TensorFlow es una biblioteca de código abierto desarrollada por Google que se utiliza para crear y entrenar modelos de aprendizaje automático. Es una de las bibliotecas más populares y ampliamente utilizadas en el campo de la inteligencia artificial.

TensorFlow se basa en la idea de un grafo de flujo de datos, donde los nodos representan operaciones matemáticas y las aristas representan los datos multidimensionales, conocidos como "tensores", que fluyen entre ellos. Esto permite construir modelos de aprendizaje automático de manera eficiente, ya que las operaciones se pueden ejecutar en paralelo y optimizar para aprovechar las capacidades de hardware, como las GPU.

Una de las principales características de TensorFlow es su flexibilidad. Permite construir modelos de aprendizaje automático para una amplia variedad de aplicaciones, desde el reconocimiento de imágenes y el procesamiento del lenguaje

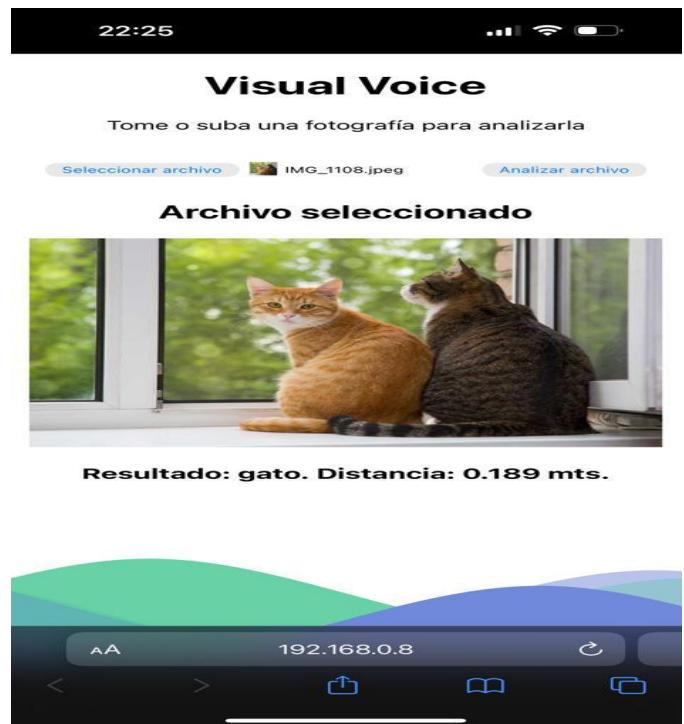


Figura 1: Programa.

natural hasta la generación de texto y la traducción automática. TensorFlow también ofrece una amplia gama de capas y herramientas para simplificar el desarrollo de modelos complejos, como redes neuronales convolucionales (CNN), redes neuronales recurrentes (RNN) y redes neuronales generativas adversarias (GAN), entre otros.

II-F. Flask

Flask es un framework de desarrollo web en Python que permite construir aplicaciones web de manera rápida y sencilla. Es ligero, flexible y minimalista, lo que lo convierte en una opción popular para desarrolladores que buscan una solución simple pero potente.

El objetivo principal de Flask es proporcionar las características esenciales para el desarrollo web sin imponer una estructura rígida o complicada. Esto significa que Flask es altamente flexible y permite a los desarrolladores organizar su código de la manera que mejor se adapte a sus necesidades.

Flask se destaca por su facilidad de uso y su curva de aprendizaje accesible para principiantes. Con unas pocas líneas de código, es posible crear una aplicación web básica en Flask. La biblioteca proporciona una API intuitiva y clara que facilita el desarrollo y permite a los desarrolladores enfocarse en la lógica de su aplicación en lugar de preocuparse por detalles técnicos complejos.

III. RESULTADOS

Estos son los resultados obtenidos al momento de realizar el software.

```

File Edit Selection View Go Run ... ⌛ ⌛ ⌛ ⌛ ⌛ ⌛ ⌛ ⌛ ...
EXPLORER ... main.py x model.py
➤ CIAR10DETCT
➤ _pycache_
➤ main.py
model.py
main.py ...
1 import matplotlib.pyplot as plt
2 import numpy as np
3 import tensorflow as tf
4 import tensorflow_datasets as tfds
5 import model
6
7 # print(tfds.list_builders())
8
9 ds = tfds.load('cifar10', split='train')
10 m = model.Classifier()
11
12 m.fit(ds, batch_size=10000, lr=0.001, epochs=10)
13 m.save_weights(model.h5")
14

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL python +v ✘ 〔 ... ^ x
Epoca: 8/100 - Lot: 1/5 - Loss: 2.104541778564453
Epoca: 8/100 - Lot: 2/5 - Loss: 2.107485717191939
Epoca: 8/100 - Lot: 3/5 - Loss: 2.0874549989163574
Epoca: 8/100 - Lot: 4/5 - Loss: 2.084797515137695
Epoca: 8/100 - Lot: 5/5 - Loss: 2.0633066363749
Epoca: 9/100 - Lot: 1/5 - Loss: 2.077080496584863
Epoca: 9/100 - Lot: 2/5 - Loss: 2.0616995344424
Epoca: 9/100 - Lot: 3/5 - Loss: 2.055388649769756
Epoca: 9/100 - Lot: 4/5 - Loss: 2.04805267359044
Epoca: 9/100 - Lot: 5/5 - Loss: 2.0414282153287354

```

Figura 2: Entrenamiento de modelos.

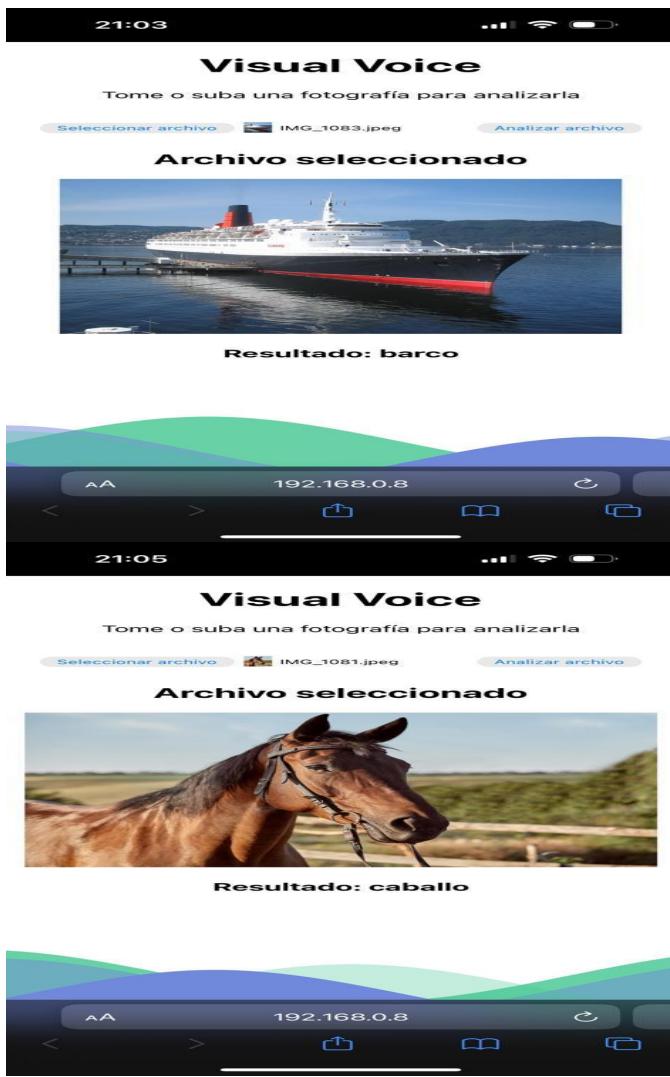


Figura 3: Reconocimiento.

IV. CONCLUSIONES

- Las tecnologías utilizadas en este proyecto, han permitido la creación de un sistema funcional y eficiente en el reconocimiento de objetos y sintetización de voz.
- El uso de Python, con sus librerías como Matplotlib, Tensorflow y Numpy, ha facilitado entrenamiento de los modelos con inteligencia artificial para la detección de objetos.
- Por otro lado, la utilización de hardware como Raspberry Pi, NVIDIA Jetson AGX Xavier y Arduino, ha permitido la integración de diversos componentes electrónicos en el proyecto del audiometro, como los auriculares y los altavoces.
- La combinación de estas tecnologías ha logrado un correcto funcionamiento del software, permitiendo la detección precisa de los objetos previstos, y con una reducción significativa en los costos de los equipos utilizados en estos diagnósticos.
- El proyecto demuestra cómo la tecnología moderna, puede utilizarse para resolver problemas reales en el ámbito de la salud, en este caso en la detección de objetos para personas no videntes.



Joel Juan Pablo Gramajo Chan nació en Quetzaltenango, Guatemala, el 27 de Noviembre de 2001. Se graduó del Colegio Salesiano Liceo Guatemala en Octubre del 2019 y actualmente está estudiando ingeniería en sistemas en la Universidad Mesoamericana de Quetzaltenango.

Ejerció prácticas profesionales en la Cervecería Nacional S.A. Entre sus campos de interés están la inteligencia artificial, las matemáticas y desarrollo de software.

Joel Gramajo Chan recibió títulos honoríficos en las instituciones de enseñanza básica y diversificada, obteniendo el Galardón de la Riva a la excelencia estudiantil. En el 2016 perteneció al Parlamento Juvenil de Guatemala, participando en la aprobación de un Decreto relacionado con "La necesidad pública de la creación de espacios para la construcción de una ciudadanía en la juventud guatemalteca".



Juan Carlos Neil Palacios Escobar nació en el departamento de Suchitepéquez, Guatemala el 10 de Septiembre de 2003. Cuenta con un bachiller en ciencias y letras con orientación científica, actualmente estudio ingeniería en sistemas en la Universidad Mesoamericana Sede Quetzaltenango.

Juan Carlos Neil Palacios Escobar recibió galardones en su institución por haber obtenido el primer lugar en la Feria de la Ciencia ganando el Primer Lugar así mismo por haber obtenido a nivel local y departamental ganando el Primer Lugar en el concurso de Matemáticas y Física.

Entre sus campos de interés están la inteligencia artificial, la ciencia de datos, machine learning, física, desarrollo de software, la programación web (Backend Developer).