

RECONOCEDOR DE LESCO

2/15/2024

SISTEMAS EXPERTOS

*Diego Duarte, Dylan Castillo,
Kendel Amador, Esteban Zúñiga*

Rolando Herrera Sánchez

Tabla de contenidos

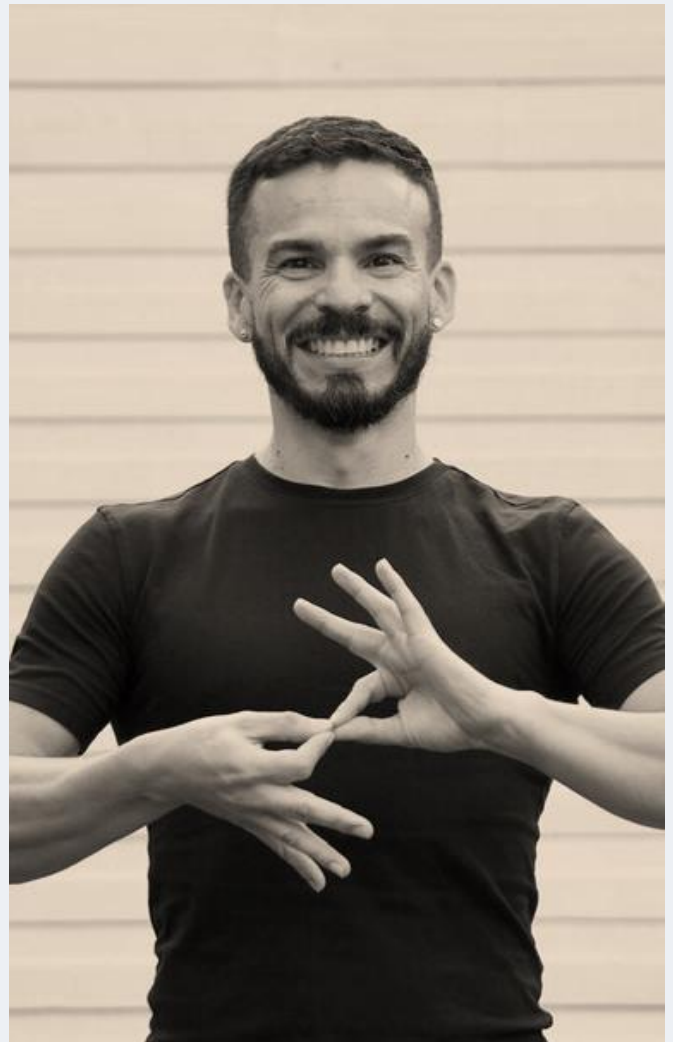
RECONOCEDOR DE LESCO	1
Introducción	3
Indicaciones Técnicas.....	4
Acerca de este proyecto	5
Construcción del proyecto.....	5
Limitantes del proyecto.....	7
Vocales en lescó	7
Utilidades de la herramienta.....	8
Glosario.....	9
Conclusión	10
Referencias	11

Introducción

A lo largo de la evolución tecnológica el humano siempre ha buscado cómo hacer para que esta facilite los diferentes procesos que debe realizar día tras día, así mismo, con la llegada de la inteligencia artificial se ha buscado la manera de que esta haga procesos por nosotros, la idea de que la IA procese lenguaje natural ha innovado tanto que se han desarrollado diversas formas de procesar lenguaje.

Este documento tiene como objetivo explicar de la mejor manera la construcción de un detector de señas LESCO el cual está delimitado para que detecte las vocales y permita señalar de cuál letra se trata.

De esta manera se logra apoyar de una manera moderna a la comunidad de hablantes LESCO, permitiendo así una herramienta que puede estar al alcance de muchos y que de alguna u otra forma sea de utilidad para las personas con discapacidad auditiva.



INDICACIONES TÉCNICAS

Como parte de las tecnologías que se utilizaron para realizar este proyecto se tiene:

Módulo / Tecnología	Versión	Propósito
Python	3.8.10	Es un lenguaje de programación con bastante soporte para inteligencia artificial
Open-CV	4.9.80	Permite el procesamiento de imágenes y videos, provee una amplia gama de funciones que detección de objetos, seguimiento de movimientos, reconocimiento facial, etc.
Imutils	0.5.4	Son una serie de funciones que permiten que el uso de open-cv sea más fácil.
Pytttsx	2.90	Permite convertir texto a voz
Mediapipe	0.8.10.1	Proporciona una amplia gama de soluciones pre-entrenadas para tareas como detección de manos, detección de caras, seguimiento de poses humanas, entre otros.
Yolo	8	Red neuronal que por medio de entrenamiento permite detectar objetos en tiempo real

ACERCA DE ESTE PROYECTO

A continuación, se detallará de qué manera fue creado el proyecto, así como también las fuentes:

Construcción del proyecto

Como parte de la elaboración del proyecto se ha tomado como referencia un canal de YouTube el cuál su dueño es especialista en inteligencia artificial y justamente posee un proyecto en donde se explica cómo crear un reconocedor de señas LESCO, sin embargo, el código no se encontraba en ningún repositorio, por lo que se tuvo que transcribir y hacer algunas modificaciones con forme se iba entrenando el modelo. El proyecto consta de múltiples partes, a pesar de esto, los archivos más importantes son **inferencia.py** y **seguimientoManos.py** los cuales contienen respectivamente el código de detectar las manos y poder realizar la lectura de los archivos necesarios tales como el archivo .pt el cuál es el que tiene los datos para la ejecución, por así decirlo el resultado del entrenamiento y así mismo el archivo **seguimientoManos.py** proporciona una clase donde tiene todos los métodos necesarios para que **inferencia.py** pueda hacer consumo de esas funciones.

La clase **seguimientoManos.py** se compone de los siguientes métodos:

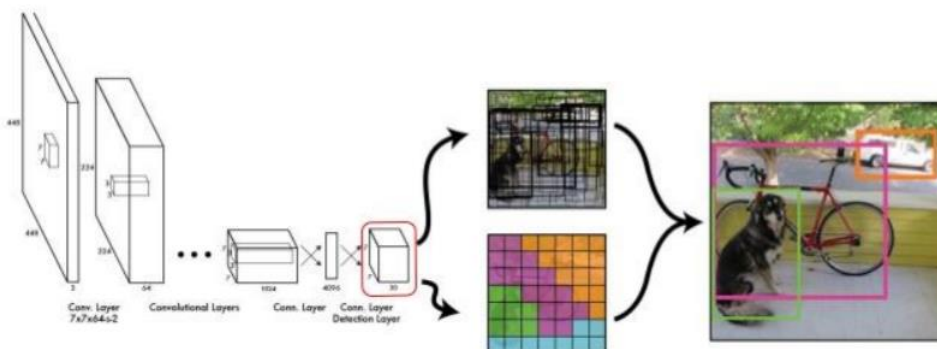
Método	Parámetros	Descripción
__init__	mode, max_manos, model_complexity, conf_detection, conf_segui	Este es el método constructor, se encarga de inicializar todas las variables necesarias para el funcionamiento del modelo
encontrarManos	Frame, dibujar	Este método permite encontrar la mano en cámara y hacer un dibujo de la misma, con el fin de poder marcar la detección. Al tener el parámetro dibujar el valor de True, dibuja los puntos de referencia y las conexiones de la mano en el fotograma.
encontrarPosicion	Frame, manoNum, dibujarPuntos, dibujarBox, color	Se utiliza para encontrar la posición específica de una mano en el fotograma, incluyendo la delimitación y los puntos de referencia.

dedosArriba		Está diseñado para detectar cuantos dedos están levantados en una mano detectada.
distancia	p1, p2, frame, dibujar, r, t	Este método calcula la distancia entre dos puntos de referencia en una mano detectada en un fotograma de entrada, y opcionalmente dibuja las líneas y círculos correspondientes en el fotograma.

La clase **inferencia.py** hace uso de métodos de la clase anterior, permitiendo así separar el código de lo más general al código relacionado con el modelo de inteligencia artificial, a continuación, se detallará un poco sobre el código de este archivo ya que como tal no tiene métodos, pero es la fuente principal de ejecución y hacer funcionar la detección de señas.

En este código caben varias importaciones como lo serían los módulos cv2 y “ultralytics” que específicamente utiliza el componente **YOLO**, así mismo la inicialización de cv2 para poder hacer la lectura de la cámara del dispositivo en el que se está corriendo, cabe destacar que también permite reconocer cámaras externas como las del teléfono celular, una vez hecho esto, se declaran las resoluciones de la cámara. Seguido de esto se procede a leer el modelo utilizando YOLO y un archivo llamado **vocales.pt** este archivo en particular se utiliza para cargar modelos de redes neuronales, junto con sus parámetros entrenados los archivos **.pt** son útiles porque permiten guardar el estado completo de un modelo de **PyTorch** de una manera conveniente y compacta. Esto significa que puedes entrenar un modelo en una máquina y luego guardar el archivo .pt para usarlo en otra máquina, o incluso para usarlo en el futuro sin tener que volver a entrenar el modelo desde cero.

YOLO: You Only Look Once



Complementando esto se llega a la parte donde todo ocurre de manera recurrente, utilizando un ciclo que siempre se mantendrá para mantener la cámara encendida y detectando la mano que se le coloca al frente, de esta manera se lanza una capturadora, esta envía un frame al método de encontrarmanos, así mismo se hace el posicionamiento de la mano (en este caso está delimitado a una sola mano) y de este modo al detectar la mano se condiciona el ciclo, en caso de que haya encontrado una mano se extraerá información del recuadro dibujado y se asignan márgenes, así con esto se crea un recorte y posteriormente se redimensiona para finalmente predecir el resultado que en este caso sería la devolución de la etiqueta, es decir, a que letra pertenece la seña, nuevamente condicionando si hay resultados fijará las coordenadas y las etiquetas para poder visualizar la detección en pantalla, de lo contrario se mantendrá en busca de la seña LESCO que se le pueda mostrar. Una vez habiendo detectado la seña, se le envía la información al módulo locutor, este módulo lo que contiene es un método que hace utilidad de la librería pyttsx3 el cual permite recibir un texto como parámetro y este por medio del altavoz del computador que en el que se está ejecutando va a reproducir el texto en voz alta. Este apartado es un agregado extra que se le quiso dar al proyecto meramente por estética.

Limitantes del proyecto

Este proyecto fue creado en base a otro ya existente, sin embargo, por tiempo se tuvo que delimitar tanto a nivel de **dataset** como de letras a reconocer, específicamente se reconocen las letras de las vocales (A-E-I-O-U), así mismo la cantidad de entrenamiento fue limitado a la capacidad que tenía el computador más potente a disposición del grupo de desarrollo, aún con estos obstáculos se logró un resultado aceptable al reconocer las diferentes señalizaciones **lesco**.

Vocales en lesco

Como anteriormente se menciona, el modelo fue entrenado únicamente para reconocer las letras vocales:



Utilidades de la herramienta

Como parte de este proyecto se enlistan algunas utilidades que se le puede dar al proyecto a la hora e implementarlo en un ámbito u otro dejando como consecuencia la virtualización de esta lengua y así mismo el apoyo digital hacía esta comunidad.

- **Comunicación inclusiva:** Facilitaría la comunicación entre personas sordas que utilizan señas LESCO y personas que no conocen el lenguaje de señas.
- **Acceso a la educación:** Un modelo de reconocimiento de señas LESCO podría integrarse en plataformas de aprendizaje en línea o aplicaciones educativas para ayudar a las personas sordas a acceder a contenido educativo en su lengua nativa.
- **Integración de dispositivos tecnológicos:** Un modelo de reconocimiento de señas LESCO podría integrarse en dispositivos tecnológicos, como teléfonos inteligentes, tabletas o dispositivos de asistente virtual, para permitir a las personas sordas interactuar con tecnología de forma más natural y eficiente.
- **Apoyo en la traducción del contenido multimedia:** Podría utilizarse en la traducción automática de contenido multimedia, como videos en línea o programas de televisión, para proporcionar subtítulos o interpretación en tiempo real en señas LESCO, lo que facilitaría el acceso a la información y el entretenimiento para personas sordas.

Glosario

Glosario de palabras clave para mayor guía al lector:

Palabra	Significado
LESCO	Lenguaje de señas costarricense
YOLO	You Only Look Once por sus siglas en inglés, es un modelo de inteligencia artificial, más concretamente una red neuronal.
DataSet	Un dataset es un conjunto bastante grande de datos, ya sea datos de texto, imágenes, música, entre otros.
Clase	Una clase es un componente de programación orientada a objetos que permite moldear un esquema en el que se tendrá métodos y atributos para una determinada razón
Métodos	Acciones que realiza un objeto instanciado de una clase
Etiquetas	Una etiqueta es un par clave valor que se define para un determinado objeto, su función es clasificar ese objeto con tal etiqueta, ejemplo una foto de la seña “A” con la etiqueta “A”

Conclusión

En conclusión, el desarrollo de este proyecto en apoyo de un modelo de inteligencia artificial para detectar señas LESCO (Lengua de Señas Colombiana) representa un avance significativo en la promoción de la inclusión y accesibilidad para personas sordas en diversos aspectos de la vida cotidiana. Al proporcionar una herramienta que puede reconocer y comprender el lenguaje de señas, este proyecto busca facilitar la comunicación fluida entre personas sordas y la sociedad en general, así como mejorar su acceso a la educación, servicios y tecnología. La aplicación práctica de este modelo abarca desde la integración en dispositivos tecnológicos hasta su uso en servicios de atención al cliente y traducción de contenido multimedia, demostrando su potencial para promover la igualdad de oportunidades y la inclusión social. Además, este proyecto destaca el poder transformador de la inteligencia artificial para abordar desafíos sociales y mejorar la calidad de vida de diversas comunidades.

Referencias

Sanchez, S. (2023, Junio 19). *Youtube*. Retrieved from Youtube:
<https://www.youtube.com/watch?v=rk7zOBRJWCc&t=1218s>

Sánchez, S. (2023, Julio). *Youtube*. Retrieved from Youtube:
<https://www.youtube.com/watch?v=8cbCruS5Z-E&t=2638s>