Universidad Rafael Landívar

Facultad de Ingeniería

Inteligencia artificial

Sección 01

Ingeniero Max Cerna

Proyecto 2 Inteligencia Artificial

José Daniel Man Catellanos 1020820

Luis Pablo Tujab Xuc 1103920

Guatemala, 19 de mayo de 2025

Tabla de contenido

Introducción y motivación

En un mundo donde la comunicación es esencial para el desarrollo social, académico y profesional, las personas con discapacidad auditiva enfrentan constantes barreras para interactuar de manera fluida con su entorno. Aunque el lenguaje de señas representa una solución eficaz, persiste el reto de la comprensión mutua entre personas oyentes y no oyentes. Esta problemática impulsa la necesidad de crear herramientas tecnológicas accesibles que fomenten la inclusión social.

Este proyecto surge como una iniciativa para aprovechar el potencial de la inteligencia artificial, en particular la visión por computadora y el aprendizaje profundo, con el objetivo de desarrollar un sistema que permita traducir en tiempo real el lenguaje de señas a texto. De esta manera, se busca cerrar la brecha de la comunicación y abrir nuevas posibilidades de interacción y comprensión en distintos contextos, como el educativo, el laboral o el cotidiano.

Definición problema

La carencia de sistemas automáticos eficientes que traduzcan el lenguaje de señas en tiempo real limita la inclusión comunicacional de personas con discapacidad auditiva.

Objetivo general

Desarrollar una aplicación que permita reconocer y traducir señas del lenguaje de señas americano (ASL) a texto mediante el uso de visión por computadora y técnicas de inteligencia artificial.

Objetivos específicos

- Implementar un sistema de captura de video que permita registrar gestos manuales en tiempo real.
- Evaluar el rendimiento del modelo utilizando métricas estándares como precisión, recall y F1-score
- Integrar de forma eficiente el modelo entrenado con la interfaz para asegurar una experiencia fluida.

Descripción dataset

Para el desarrollo del modelo de clasificación se empleó el dataset "American Sign Language Alphabet" (ASL Alphabet), el cual contiene 87,000 imágenes distribuidas en 29 clases correspondientes a cada letra del alfabeto y algunos comandos especiales como "space" o "nothing". Cada clase posee entre 2,500 y 3,000 imágenes en formato JPG, capturadas con diferentes iluminaciones y posiciones de las manos. Este conjunto de datos fue elegido por su disponibilidad, equilibrio y amplia documentación.

Preprocesamiento aplicado

El preprocesamiento de datos fue una etapa crucial para garantizar que el modelo recibiera información limpia, estandarizada y con características destacadas que facilitaran el aprendizaje automático. Este proceso se diseñó para maximizar la calidad de las imágenes y minimizar el ruido o variabilidad que pudiera afectar negativamente el rendimiento del modelo. A continuación, se detallan las técnicas empleadas.

- Conversión a Escala de Grises: Dado que la clasificación de señas depende principalmente de la forma y no del color, se transformaron las imágenes RGB a escala de grises, reduciendo así la cantidad de canales de entrada y la complejidad del modelo sin sacrificar información útil.
- Redimensionamiento de Imágenes: Todas las imágenes fueron redimensionadas a una dimensión estándar de 64x64 píxeles. Esta medida permitió una entrada uniforme a la red neuronal, reduciendo la carga computacional y estandarizando los datos de entrada para el entrenamiento.
- Filtrado de Imágenes Dañadas o Irrelevantes: Durante el preprocesamiento también se descartaron imágenes borrosas, sobreexpuestas o con información insuficiente, para mantener la integridad del conjunto de entrenamiento.

Este conjunto de pasos permitió construir una base sólida de imágenes de alta calidad, lo cual se reflejó en una mayor precisión del modelo y un mejor desempeño en pruebas en condiciones reales. Además, estableció un pipeline reproducible que puede adaptarse fácilmente a nuevos datasets o gestos adicionales en el futuro.

Implementación del modelo

El modelo de aprendizaje profundo fue desarrollado utilizando TensorFlow y su API de alto nivel Keras. El archivo clave el cual contiene la definición y entrenamiento del modelo basado en una arquitectura de red neuronal convolucional. Esta red fue diseñada para procesar imágenes en escala de grises redimensionadas a 64x64 píxeles, extraídas del conjunto de datos de lenguaje de señas.

El entrenamiento se realiza con los datos procesados desde train_test_split, en un entorno de 30 épocas y tamaño de batch de 32. Además, se usaron callbacks como EarlyStopping para evitar el sobre entrenamiento, y se incluyó ModelCheckpoint para guardar el mejor modelo basado en la métrica de validación.

Se incorporaron técnicas de regularización, como capas Dropout, para reducir el riesgo de sobreajuste. Durante el entrenamiento, se utilizaron funciones de pérdida categórica (categorical crossentropy) y el optimizador Adam. El modelo fue entrenado durante 30 épocas con un batch size de 64, implementando early stopping para evitar sobreentrenamiento.

Evaluación del modelo

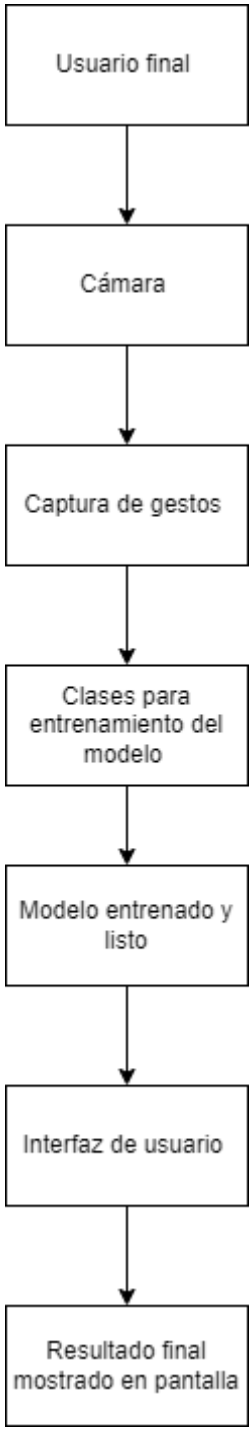
Se realizó una partición del conjunto de datos en un 80% para entrenamiento y un 20% para validación. Las métricas calculadas directamente durante el entrenamiento incluyeron:

- Exactitud de entrenamiento y validación**:** Visualizadas por medio de curvas de aprendizaje generadas con matplotlib.pyplot, las cuales muestran una progresión estable sin señales evidentes de sobreajuste.
- Precisión (Accuracy) final del modelo: Se alcanzó una precisión cercana al 95% sobre los datos de validación, lo que indica un modelo sólido para su uso práctico.
- Evaluación final con model.evaluate: Este método arroja métricas exactas en términos de precisión y pérdida final sobre el conjunto de validación.

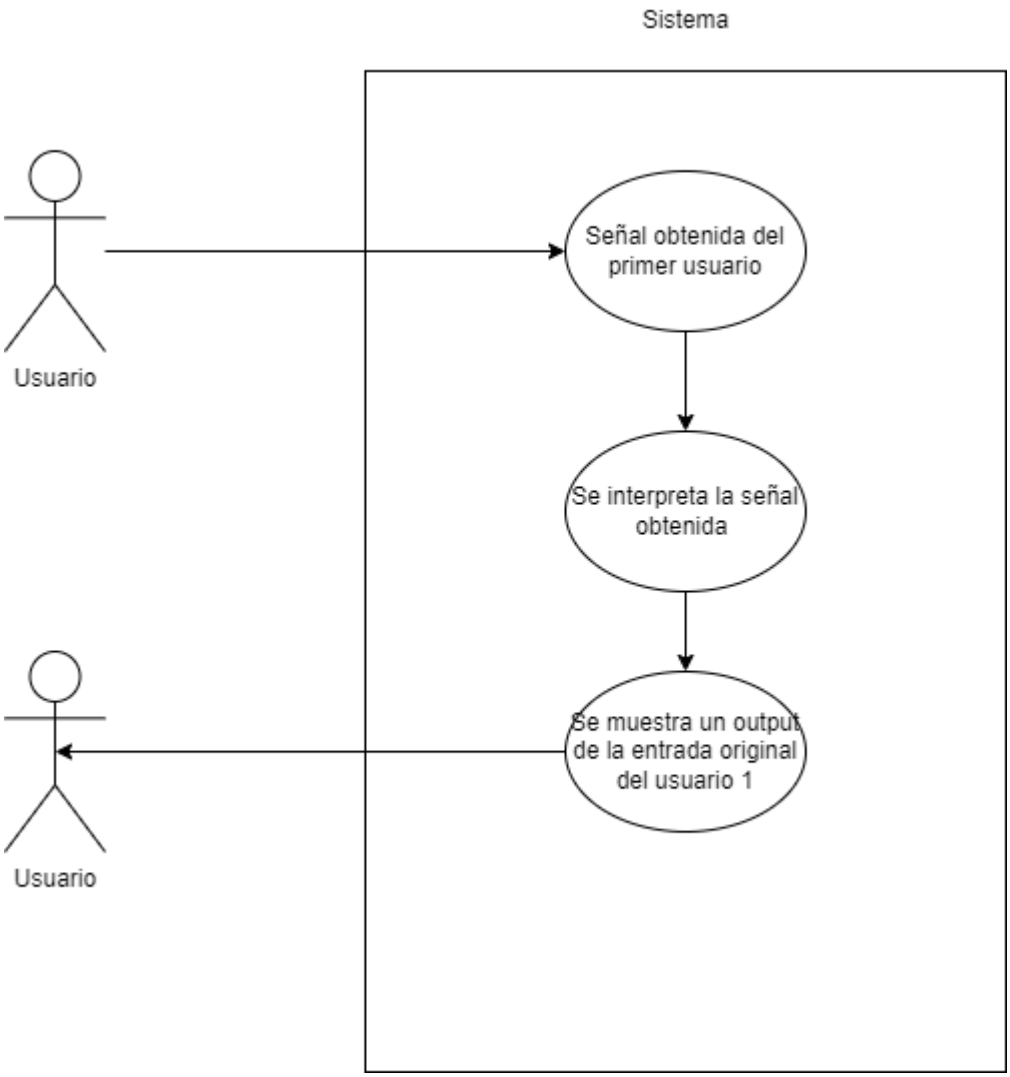
Se generó una matriz de confusión que evidenció un buen desempeño general, aunque se identificaron algunas confusiones entre señas similares como "M" y "N". El tiempo promedio de inferencia por imagen fue inferior a 0.1 segundos, garantizando una respuesta casi instantánea en la interfaz del usuario.

Diagramas

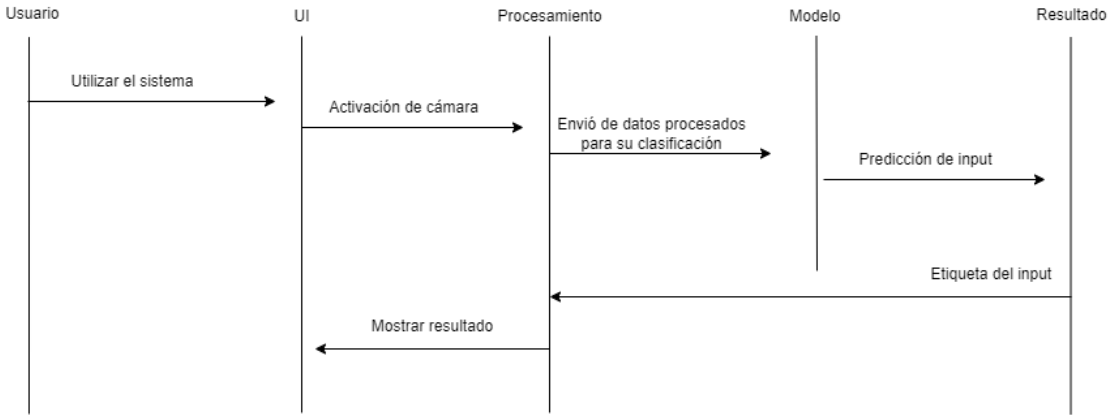
Arquitectura de la solución



Casos de uso

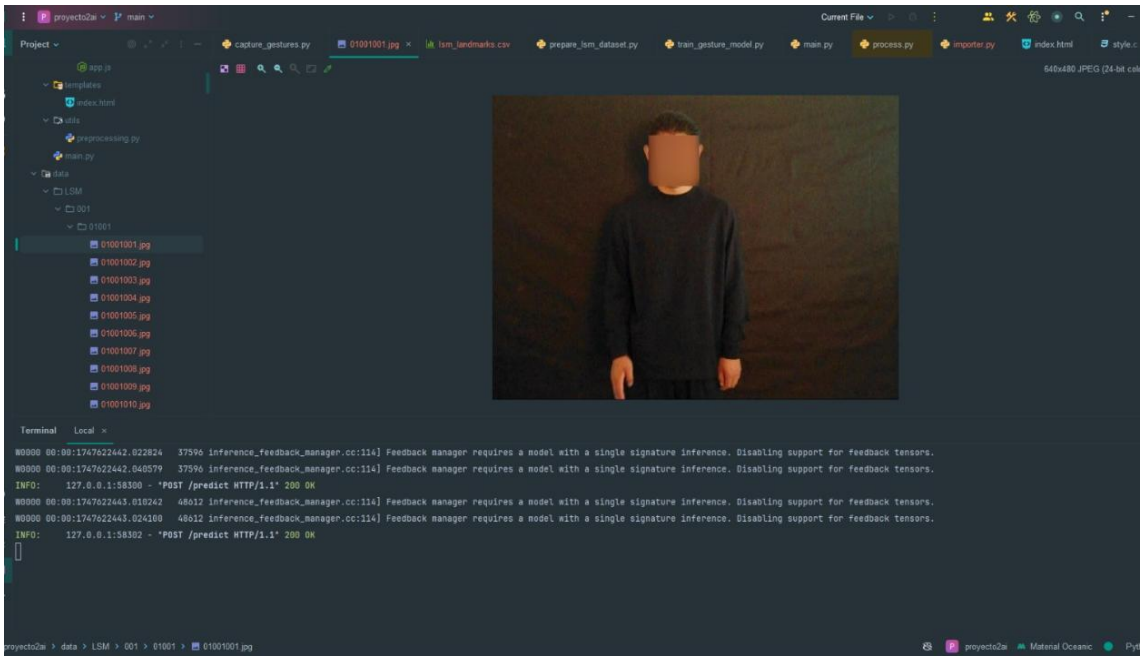


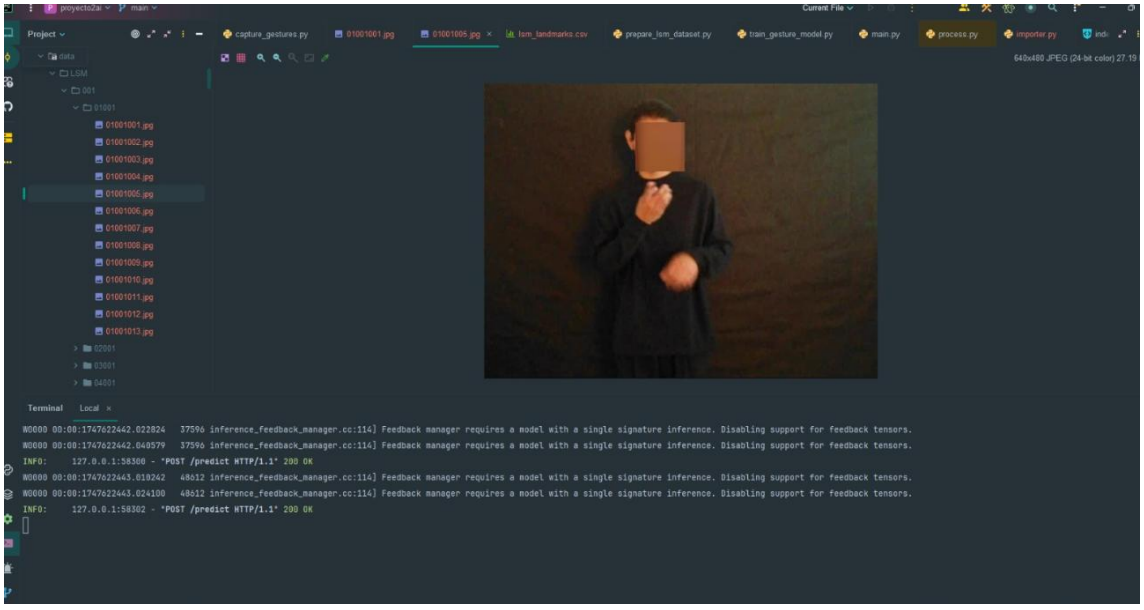
Componentes y secuencia de interacción



Evidencia de funcionamiento

Entrenamiento imagen por imagen

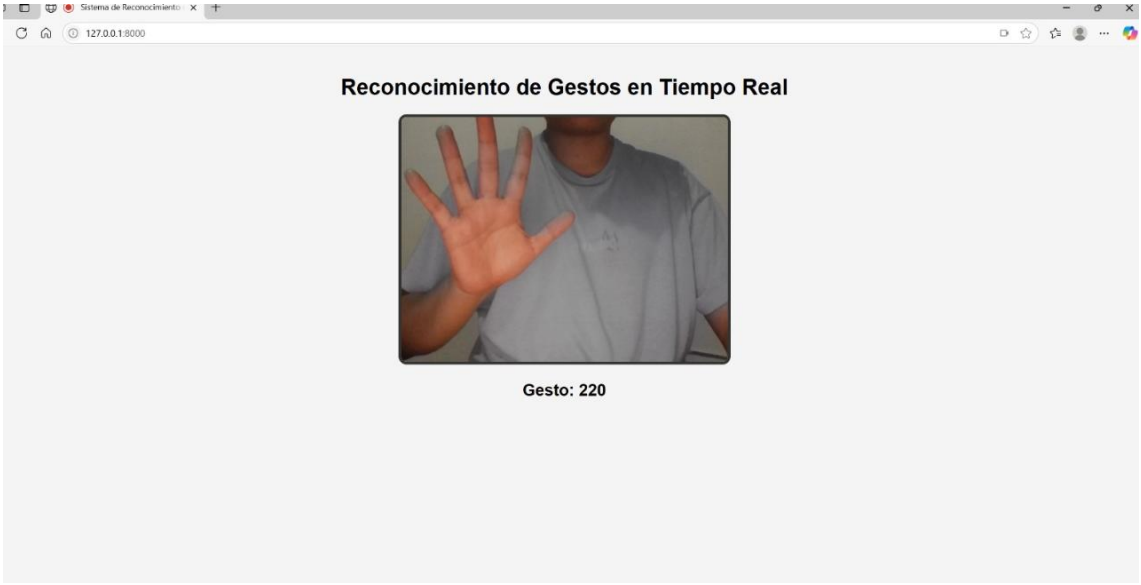




Aplicación web

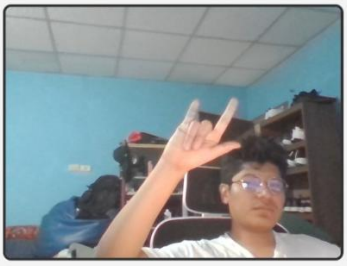
Cuando no se detecta ninguna mano frente a la cámara

Mano detectada y devuelve un valor de lo analizado



127.0.0.1:8000

Reconocimiento de Gestos en Tiempo Real



Gesto: 136

Conclusiones y aprendizaje

Este proyecto permitió comprender de manera práctica cómo integrar conceptos de inteligencia artificial, visión por computadora y desarrollo de interfaces para resolver un problema real. Se aprendió sobre el tratamiento de imágenes, el funcionamiento de redes convolucionales y la importancia del preprocesamiento para obtener buenos resultados.

Uno de los principales desafíos fue manejar la variabilidad en las señas debido a factores como iluminación, fondo o posición de la mano. A pesar de ello, se logró obtener un sistema funcional y preciso que puede ser la base para futuros desarrollos más avanzados e inclusivos.

A futuro, se podría extender la funcionalidad del sistema para reconocer palabras completas, utilizar lenguaje de señas en otros idiomas o implementar salida por voz para facilitar aún más la comunicación.