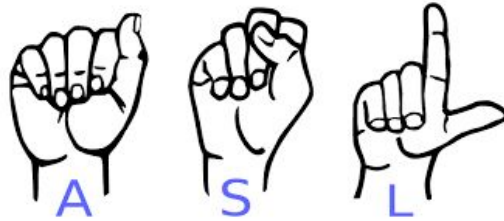


Práctica final de Visión Artificial
2021-2022



ASL detector

Patricia Cuesta Ruiz
Laura Mambrilla Moreno

ASL detector

Índice

Introducción: ASL

1. Lenguas de señas
2. ASL
3. Problema y solución

ASL detector

4. Entorno
5. Librerías
6. Transfer learning
7. Obtención del dataset
8. Tratamiento de datos
9. Ajuste de la red utilizada
10. Desarrollo
11. Resultados

Ideas descartadas

12. Otras librerías
13. Captura de imágenes

Conclusiones

14. Conclusiones

Bibliografía

15. Bibliografía
-



Introducción: ASL



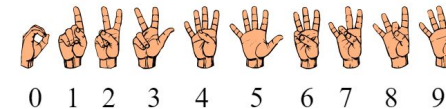
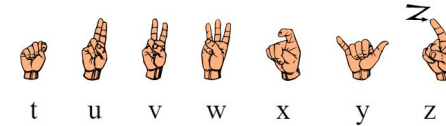
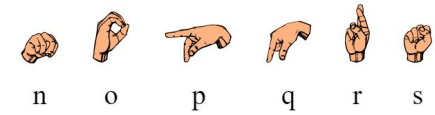
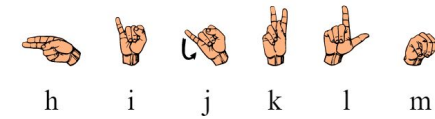
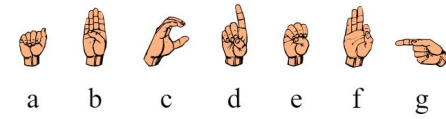
1. Lenguas de señas

- Lengua de señas:
 - americana (**ASL**, American Sign Language)
 - mexicana (LSM)
 - francesa (LSF)
 - italiana (LIS)
 - de Irlanda (IRSL)
 - española (LSE)
 - catalana (LSC)
 - británica (BSL)
 - de Nueva Zelanda (NZSL)
 - de Irlanda del Norte (NIRSL)
 - alemana (DGS),
 - de la Suiza alemana (DSGS)
 - austríaca (ÖGS)
 - de señas israelí (ISL)



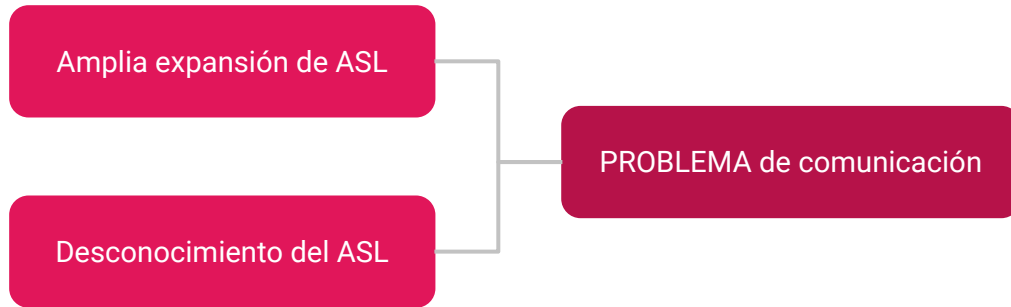
2. ASL

- ~ 1 millón de hablantes
- Estados Unidos, Canadá, México
 - Filipinas, Malasia, China
 - República Dominicana, El Salvador, Haití, Puerto Rico, Costa Rica
 - Costa de Marfil, Burkina Faso, Ghana, Togo, Benín, Nigeria, Chad, Gabón, Congo-Kinsasa, República Centroafricana, Mauritania, Kenia, Madagascar y Zimbabue

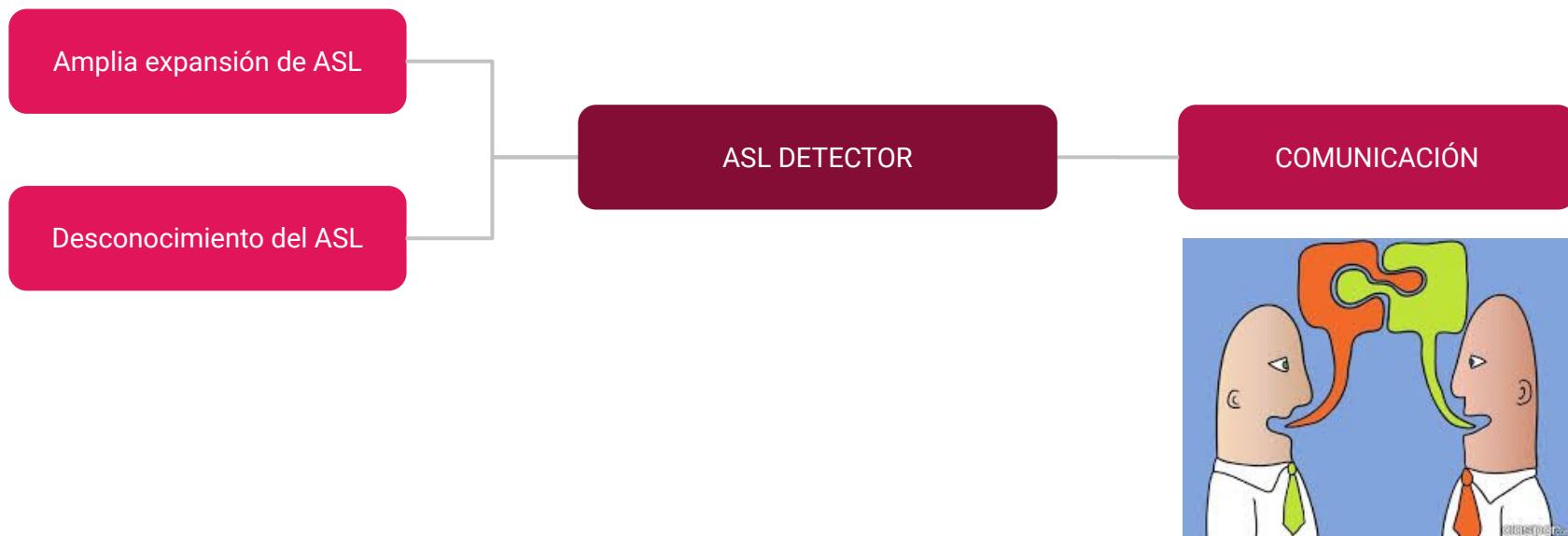




3. Problema y solución: ASL detector



3. Problema y solución: ASL detector





ASL detector
nuestro hijo



4. Entorno



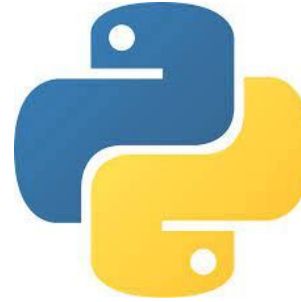
Google Colab

- Permite programar en Python de forma dinámica.
- No requiere configuración.
- Da acceso gratuito a GPUs.
- Permite compartir contenido fácilmente.



Jupyter Notebooks

- Permite desarrollar código en Python de manera dinámica, integrando bloques de texto, gráficas o imágenes.



Python

- Lenguaje multiparadigma.



Anaconda

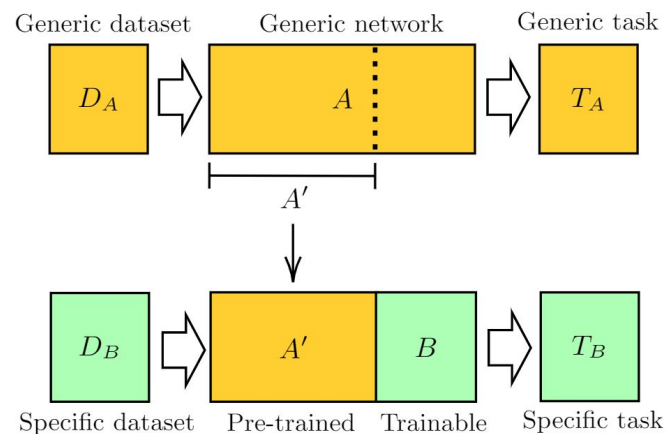
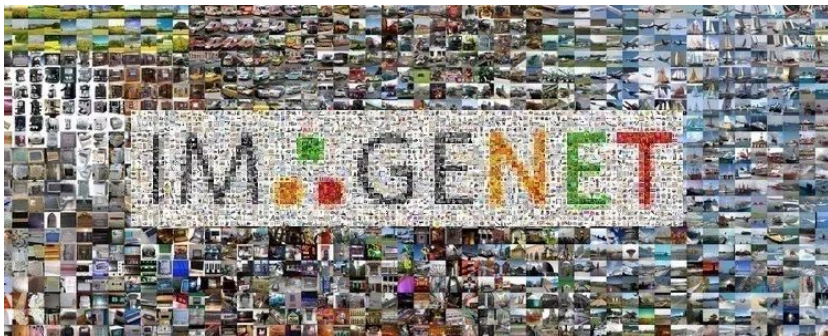
- Utilizado para crear entornos virtuales en Python.

5. Librerías



6. Transfer Learning

- Redes neuronales entrenadas anteriormente.



7. Obtención del dataset

kaggle

<https://www.kaggle.com/grassknotted/asl-alphabet>

roboflow

<https://public.roboflow.com/object-detection/american-sign-language-letters>

8. Tratamiento de datos

Augmentation Options

Randomly applied to images in your training set

Flip

Horizontal

Crop

0% Minimum Zoom, 20% Maximum Zoom

Rotation

Between -5° and $+5^\circ$

Shear

$\pm 5^\circ$ Horizontal, $\pm 5^\circ$ Vertical

Grayscale

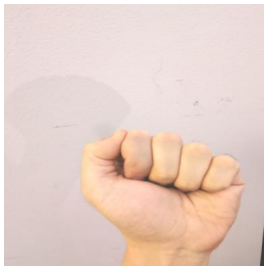
Apply to 10% of images

Brightness

Between -25% and +25%

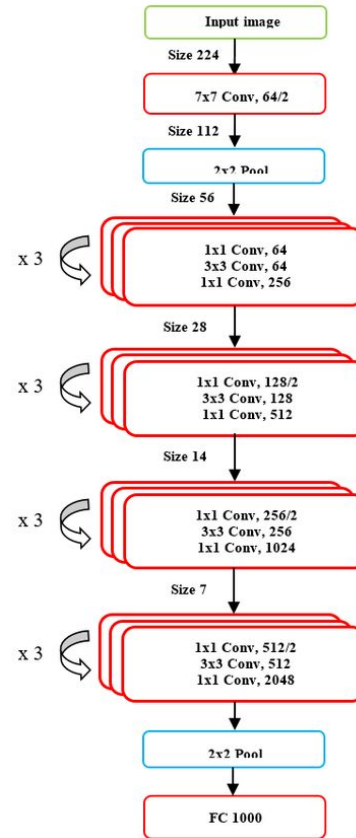
Blur

Up to 1.25px



9. Ajuste de la red utilizada

- ResNet 50
- ¿Congelar los pesos?



10. Desarrollo



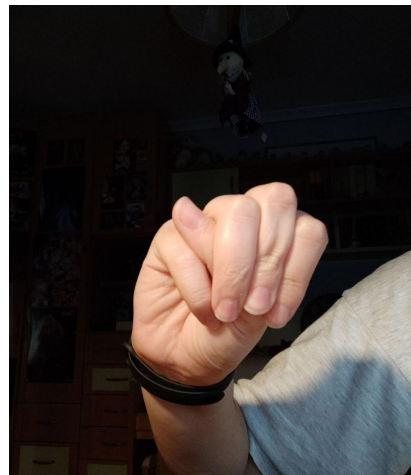
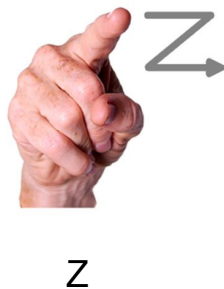
kaggle



roboflow

11. Resultados

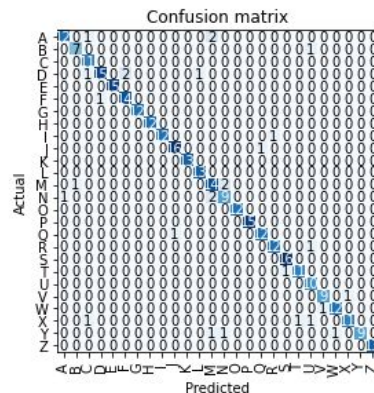
- Error Cuadrático Medio.
- Errores en letras similares.
- Errores en letras con movimiento.



M



N





Ideas descartadas

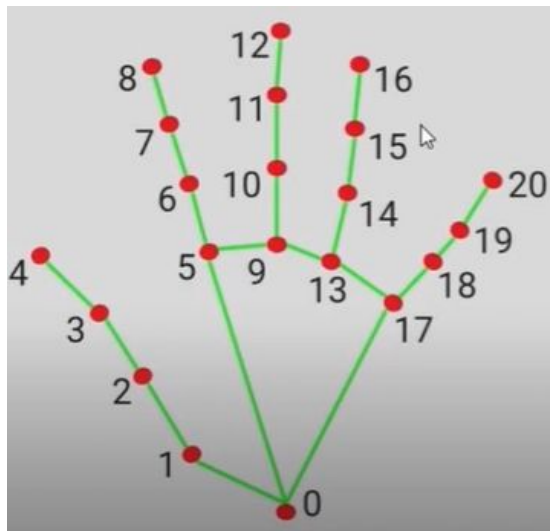


12. Otras librerías



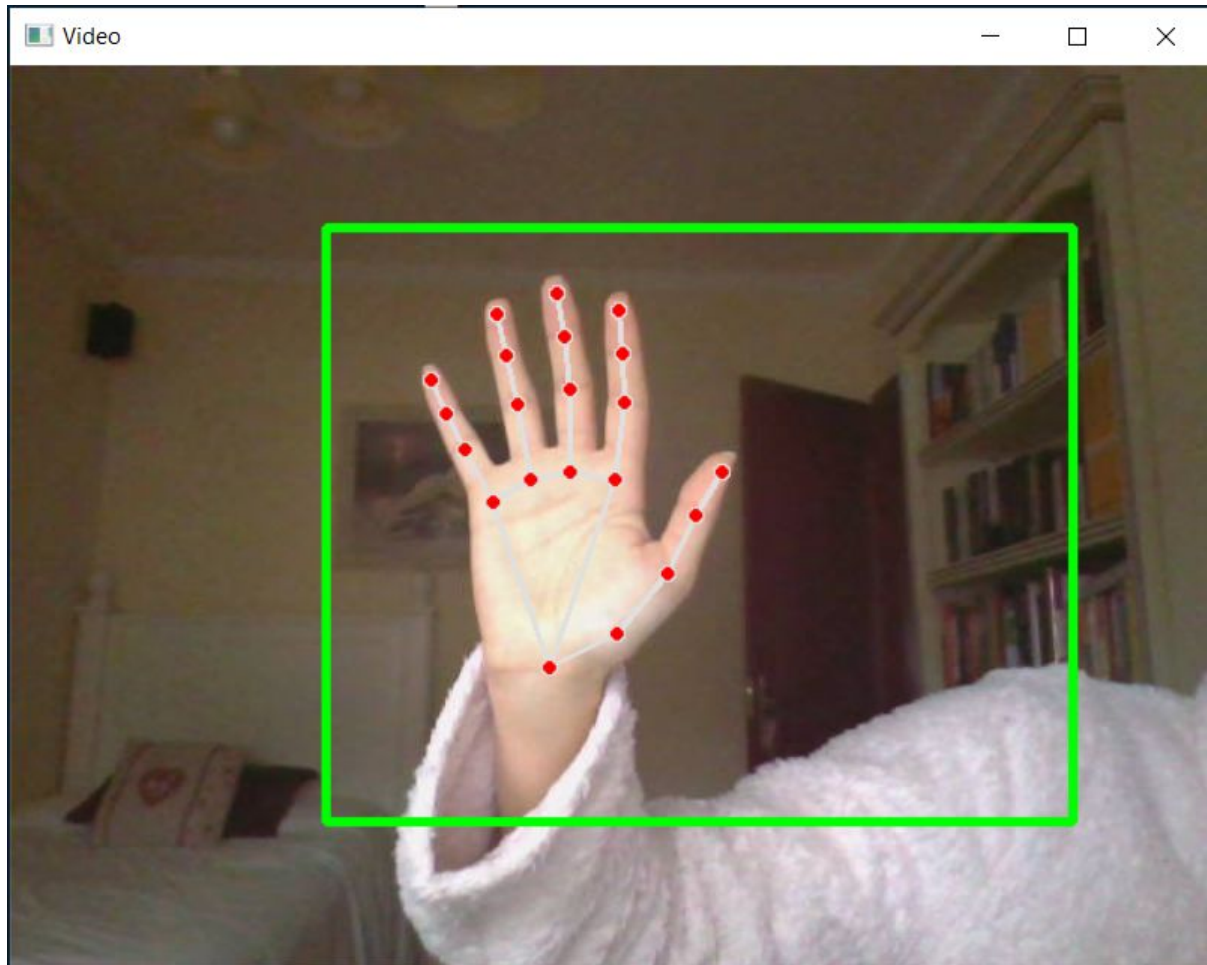
13. Creación de un dataset manualmente

- Librerías: cv2 y mediapipe



0. WRIST
1. THUMB_CMC
2. THUMB_MCP
3. THUMB_IP
4. THUMB_TIP
5. INDEX_FINGER_MCP
6. INDEX_FINGER_PIP
7. INDEX_FINGER_DIP
8. INDEX_FINGER_TIP
9. MIDDLE_FINGER_MCP
10. MIDDLE_FINGER_PIP

11. MIDDLE_FINGER_DIP
12. MIDDLE_FINGER_TIP
13. RING_FINGER_MCP
14. RING_FINGER_PIP
15. RING_FINGER_DIP
16. RING_FINGER_TIP
17. PINKY_MCP
18. PINKY_PIP
19. PINKY_DIP
20. PINKY_TIP



Conclusiones

14. Conclusiones

- Dataset con más de 1700 imágenes y más variedad.
- Letras que no se parecen a otras realiza buena clasificación.
- Siguiendo paso → robustez del modelo.



Bibliografía



15. Bibliografía

- *fastbook/ at master · fastai/fastbook · GitHub* (Available at: <https://github.com/fastai/fastbook>)
- *Fast AI Video Viewer* (Available at: <https://course.fast.ai/videos/>)