

DETECCIÓN DE TAPABOCAS EN TIEMPO REAL CON SSD



LOZANO. E, URREA. J
UNIVERSIDAD DE LOS ANDES | FACULTAD DE INGENIERÍA |
DEPARTAMENTO DE INGENIERÍA BIOMÉDICA

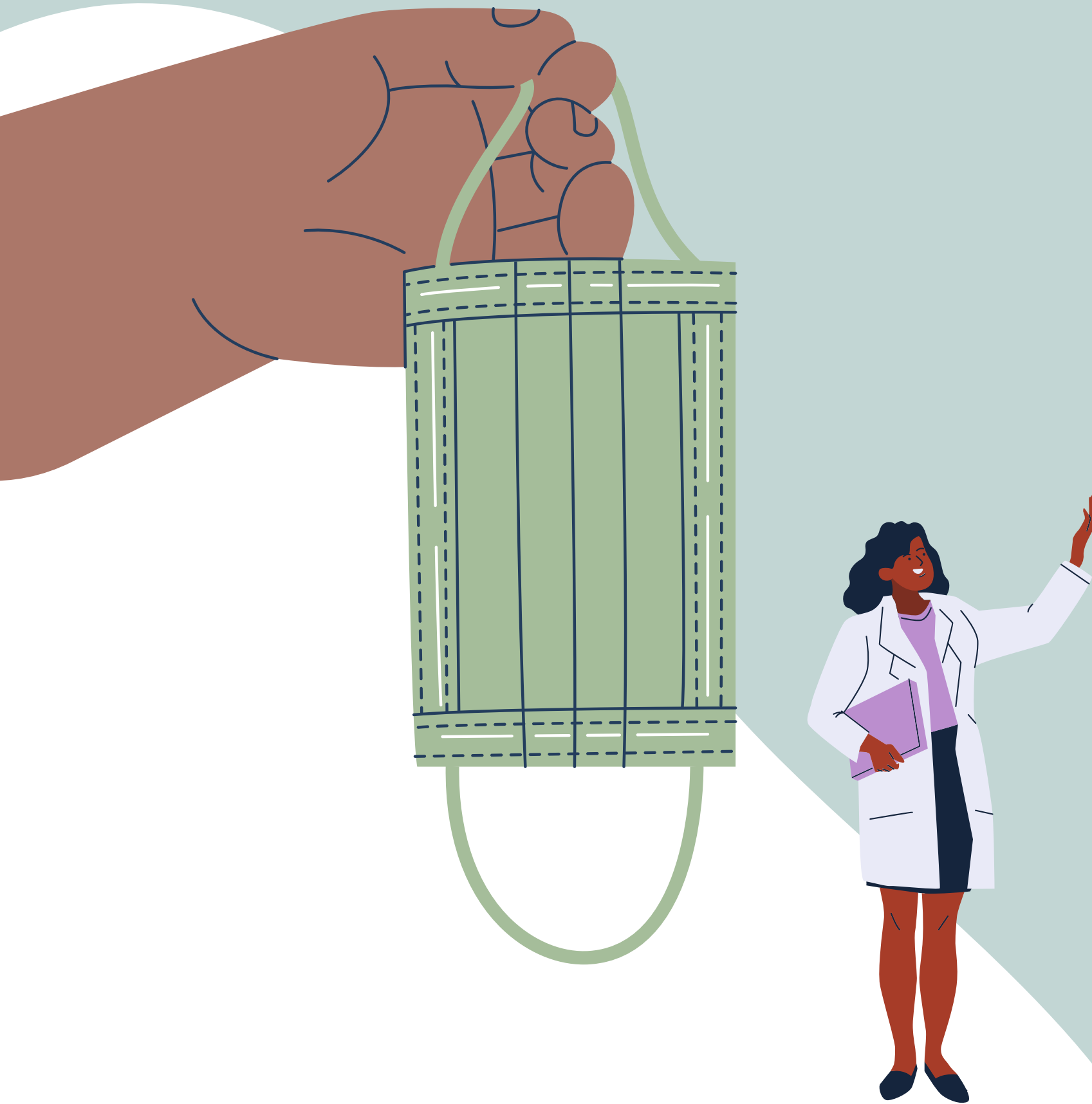
MOTIVACIÓN



El uso de tapabocas se ha vuelto parte de la vida cotidiana.

Es la principal medida de adaptación que se ha tomado a nivel mundial para impedir la transmisión de COVID-19

La consciencia social respecto al uso del tapabocas no se ha desarrollado plenamente, pues es común encontrar en sitios públicos personas con el tapabocas mal puesto o incluso sin este.





Se hace necesario automatizar el proceso de revisión del uso adecuado del tapabocas para evitar posibles contagios.





METODOLOGÍA

Dataset

Categorías:

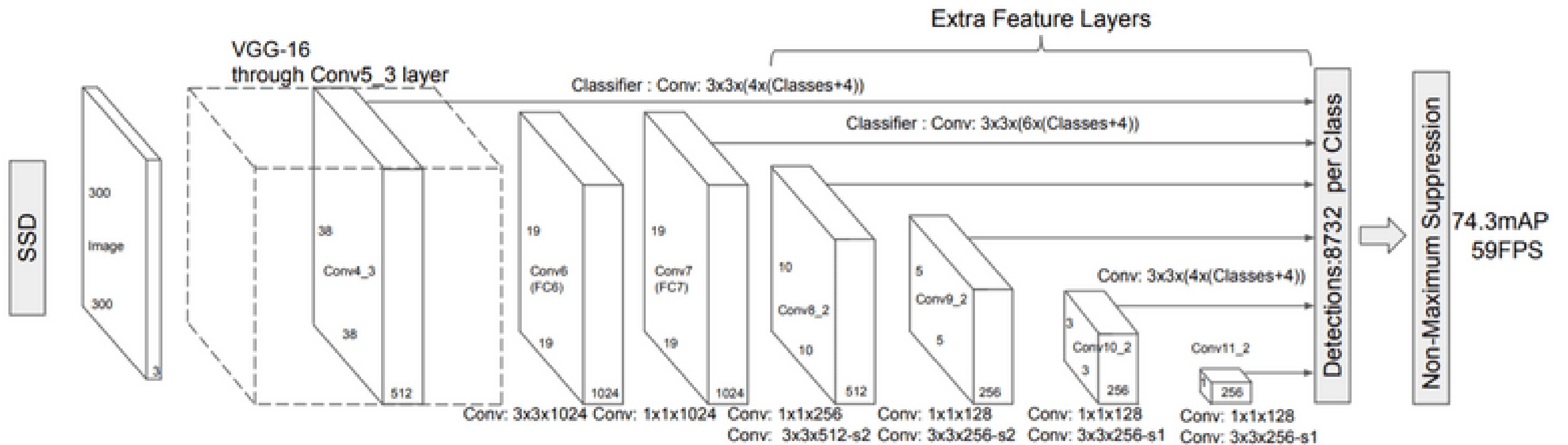
- Con tapabocas (80%)
- Sin tapabocas (17%)
- Puesto incorrectamente (3%)

División:

- Entrenamiento (70%)
- Validación (15%)
- Prueba (15%)

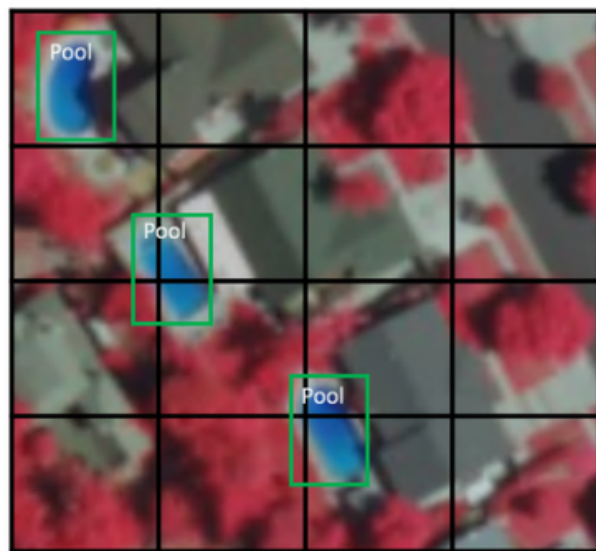


ARQUITECTURA SINGLE SHOT DETECTOR

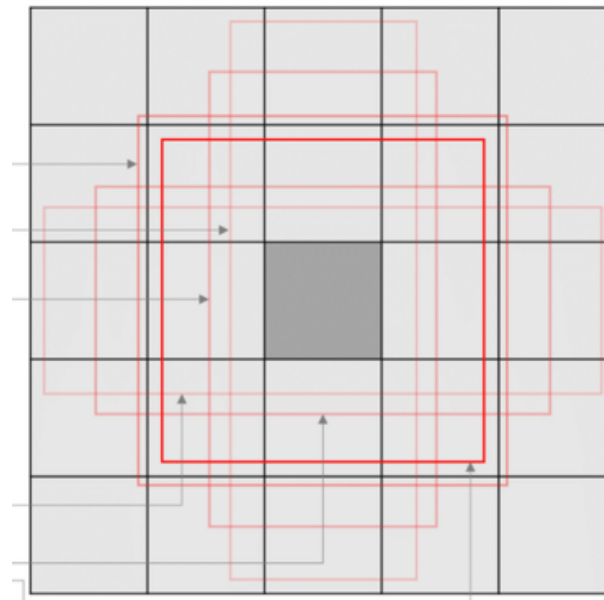


FUNCIONAMIENTO SINGLE SHOT DETECTOR

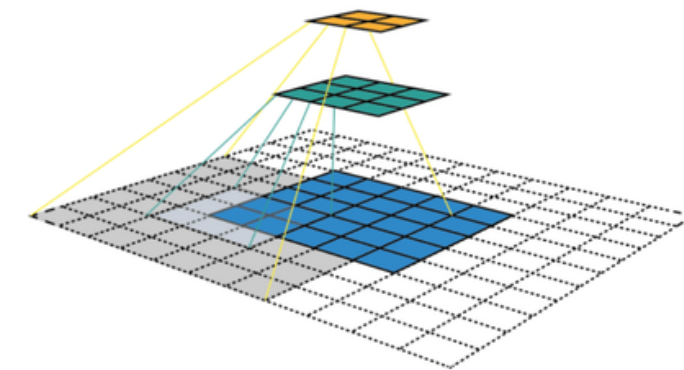
Grid cell



Anchor box



Respective field



VENTAJAS DE SSD



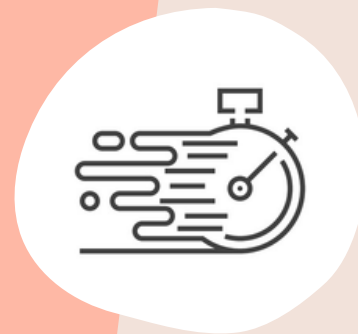
Es una red neuronal end-to-end: todos los parámetros son ajustados por los procesos de optimización



Tiene mejor rendimiento que otras arquitecturas como YOLO y RCNN para la misma tarea (PASCAL VOC 2007)



Debido a su arquitectura es un método que tiene un procesamiento más rápido que otros algoritmos. Por esta razón es útil para la detección en tiempo real



Tiene un buen funcionamiento con datasets pequeños y con clases desbalanceadas

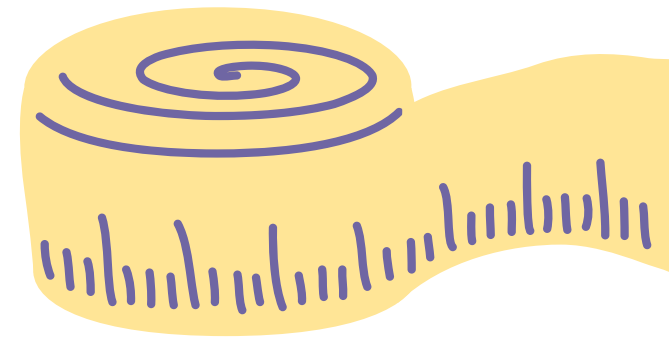


EXPERIMENTOS

Se probaron parámetros de aprendizaje y de la función de pérdida del modelo (MultiBox Loss):

- Negative/Positive Ratio
- Learning Rate
- Cross Entropy Weights
- Alpha
- Optimizador

Se usa como criterio el **mAP**



RESULTADOS



Experimentos de validación

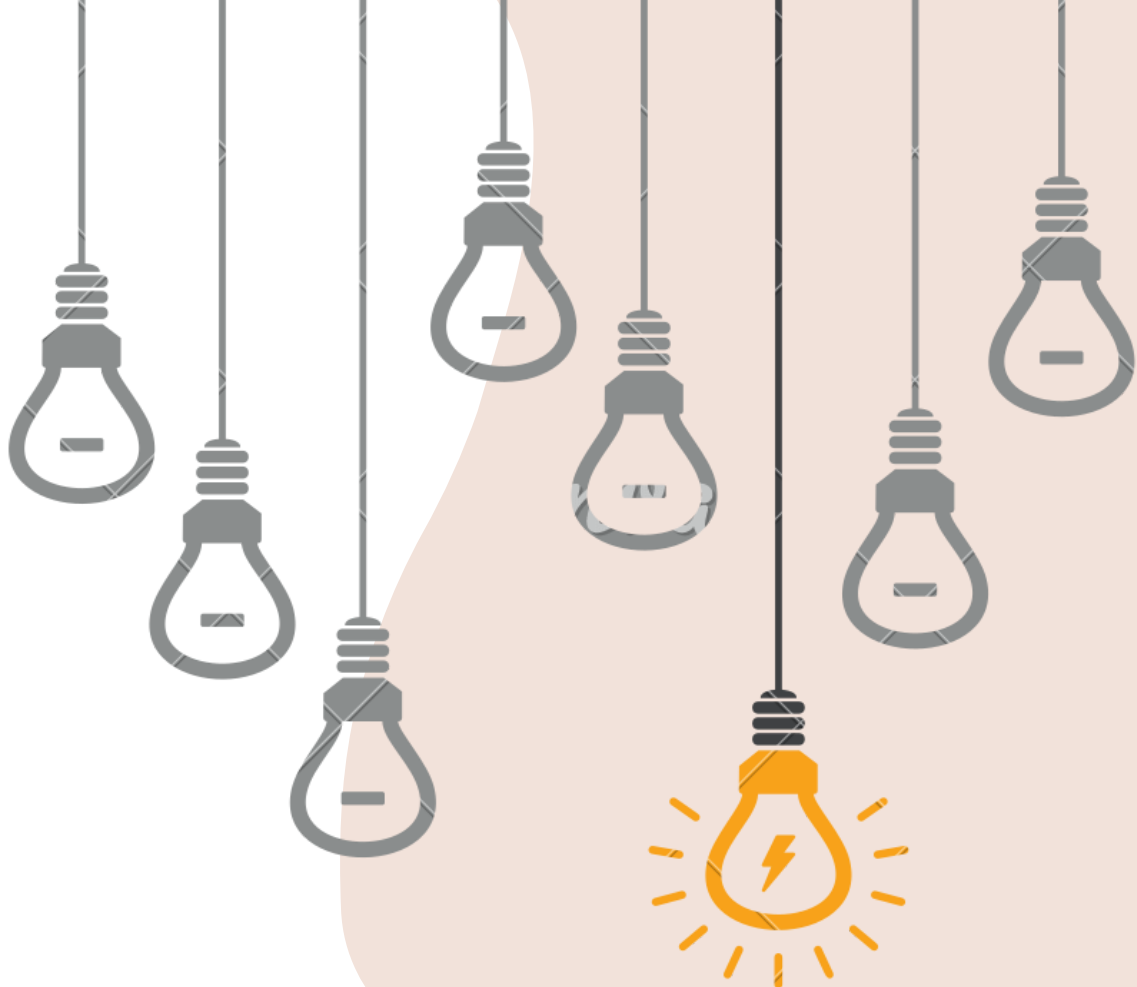
Parameter	mAP	WOM	MWI	WM
Baseline	55.6	54.4	42.6	69.8
Neg/Pos Ratio	57.5	54.3	47.3	70.9
Learning Rate	63.9	57.1	56.0	78.4
Cross Entropy W	68.0	63.1	64.3	76.5
Alpha	70.2	62.4	71.4	76.8
Optimizador*	61.1	54.2	53.6	75.4



Resultados finales

Se obtuvo un mAP de **66.7**
en el conjunto de prueba:

- Con tapabocas: **69.2**
- Sin tapabocas: **67.2**
- Puesto incorrect..: **63.6**



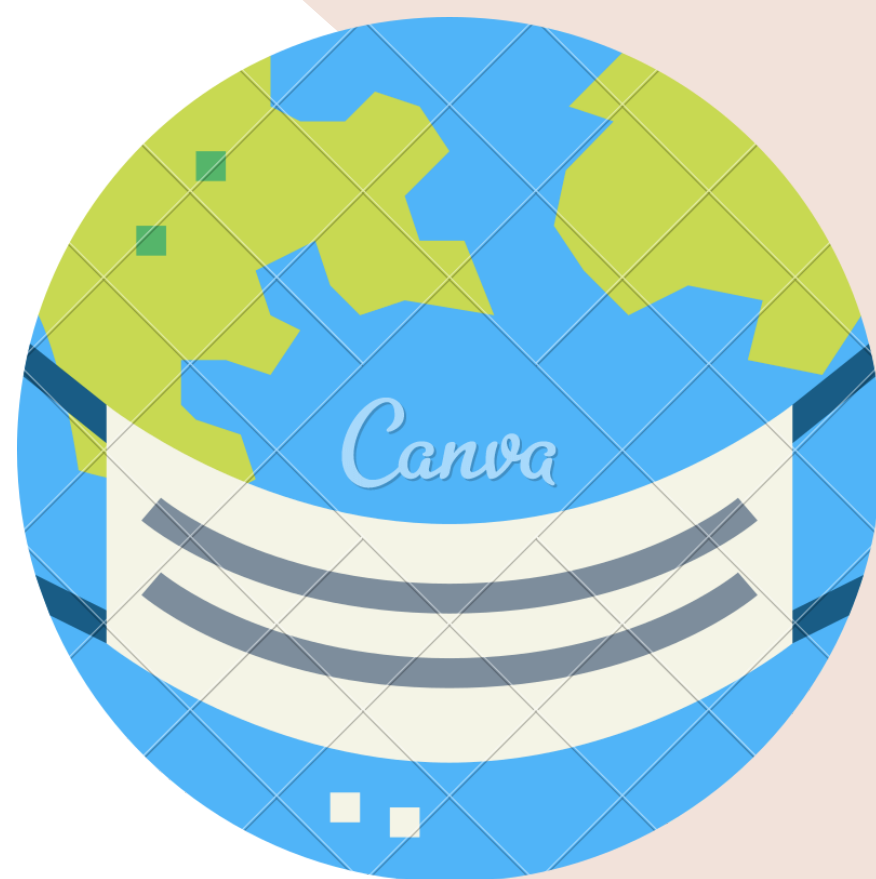
Problema 1

El modelo detecta demasiados positivos de caras pequeñas



Problema 2

El modelo se basa en la nariz y en la boca para clasificar, por lo que puede ser fácilmente engañado, por ejemplo, por una persona que se cubra con la mano



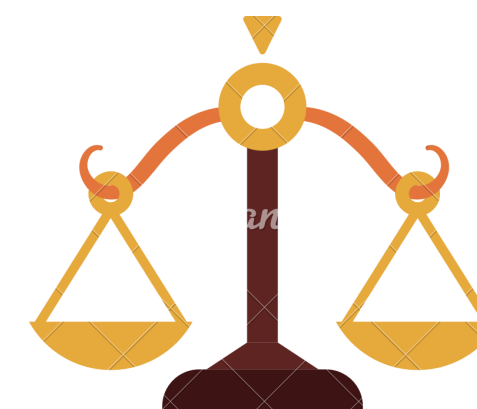
Por mejorar

Eficacia de Supresión de No Máximos, ya que a veces se detectan varias categorías en una misma cara.

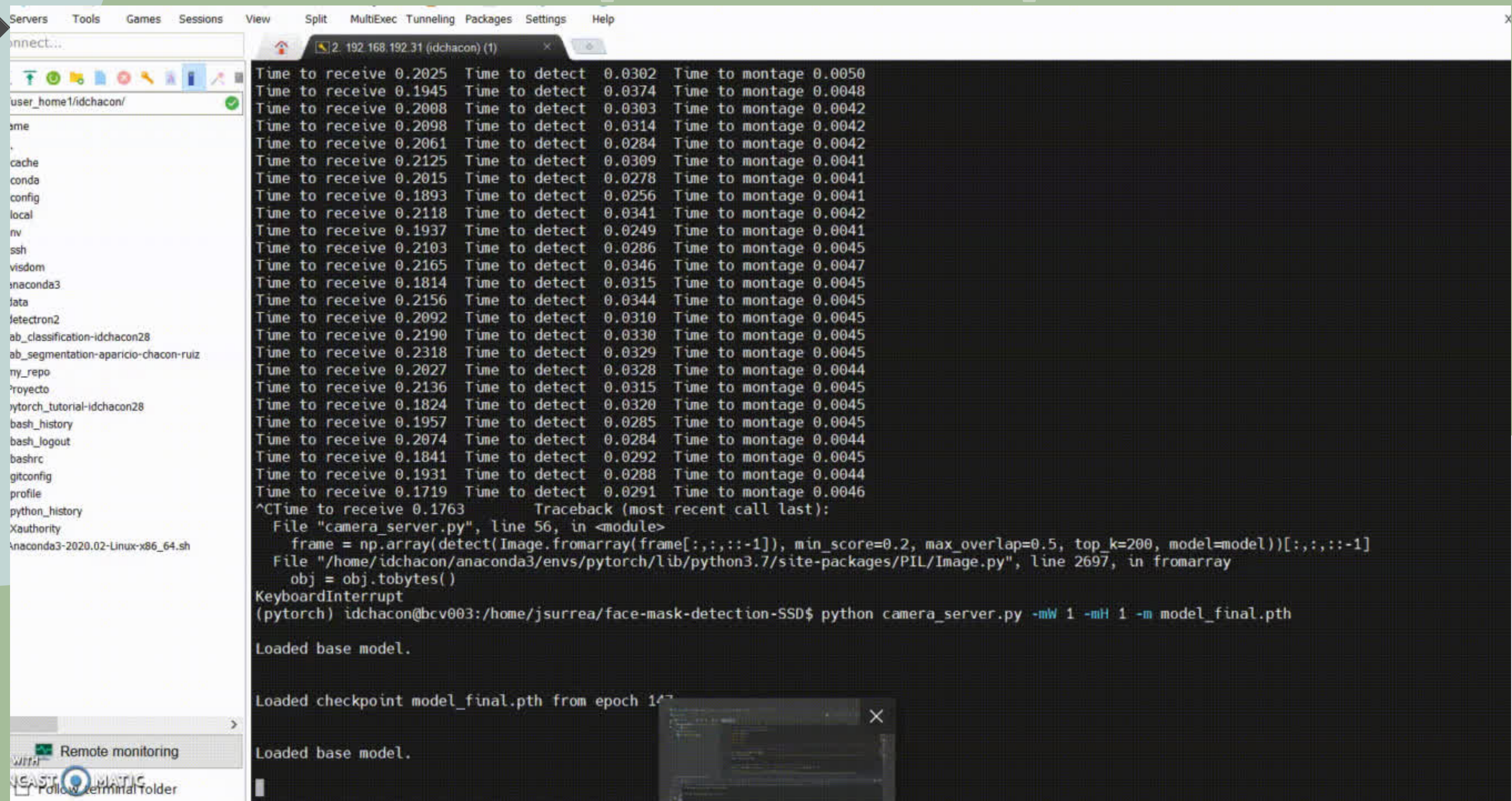


Por mejorar

Conseguir una mayor variedad de datos



Prototipo en tiempo en real



The screenshot displays a remote terminal session with a file explorer on the left and a terminal window on the right. The file explorer shows a directory structure for a user named 'idchacon' on a remote host. The terminal window shows a table of performance metrics for a face mask detection prototype, followed by a traceback error message and a command to run the prototype.

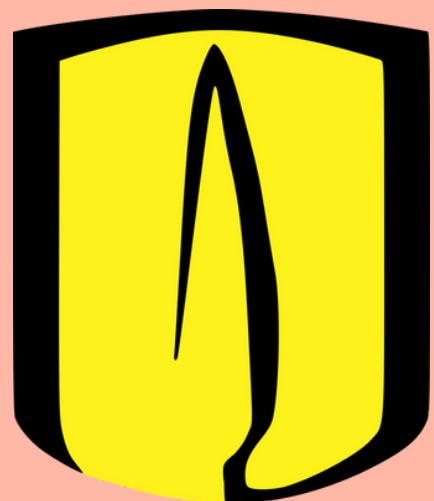
Time to receive	Time to detect	Time to montage
0.2025	0.0302	0.0050
0.1945	0.0374	0.0048
0.2008	0.0303	0.0042
0.2098	0.0314	0.0042
0.2061	0.0284	0.0042
0.2125	0.0309	0.0041
0.2015	0.0278	0.0041
0.1893	0.0256	0.0041
0.2118	0.0341	0.0042
0.1937	0.0249	0.0041
0.2103	0.0286	0.0045
0.2165	0.0346	0.0047
0.1814	0.0315	0.0045
0.2156	0.0344	0.0045
0.2092	0.0310	0.0045
0.2190	0.0330	0.0045
0.2318	0.0329	0.0045
0.2027	0.0328	0.0044
0.2136	0.0315	0.0045
0.1824	0.0320	0.0045
0.1957	0.0285	0.0045
0.2074	0.0284	0.0044
0.1841	0.0292	0.0045
0.1931	0.0288	0.0044
0.1719	0.0291	0.0046

Traceback (most recent call last):
File "camera_server.py", line 56, in <module>
frame = np.array(detect(Image.fromarray(frame[:, :, :-1]), min_score=0.2, max_overlap=0.5, top_k=200, model=model))[:, :, :-1])
File "/home/idchacon/anaconda3/envs/pytorch/lib/python3.7/site-packages/PIL/Image.py", line 2697, in fromarray
obj = obj.tobytes()
KeyboardInterrupt
(pytorch) idchacon@bcv003:/home/jsurrea/face-mask-detection-SSD\$ python camera_server.py -mW 1 -mH 1 -m model_final.pth

Loaded base model.

Loaded checkpoint model_final.pth from epoch 147

Loaded base model.



Presentado por:

Juan Sebastian Urrea López

Estudiante de Ingeniería Biomédica e
Ingeniería Industrial



Erick Sebastián Lozano Roa

Estudiante de Ingeniería Biomédica e
Ingeniería Electrónica

