DIPLOMADO EN DESARROLLO DE APLICACIONES
CON INTELIGENCIA ARTIFICIAL

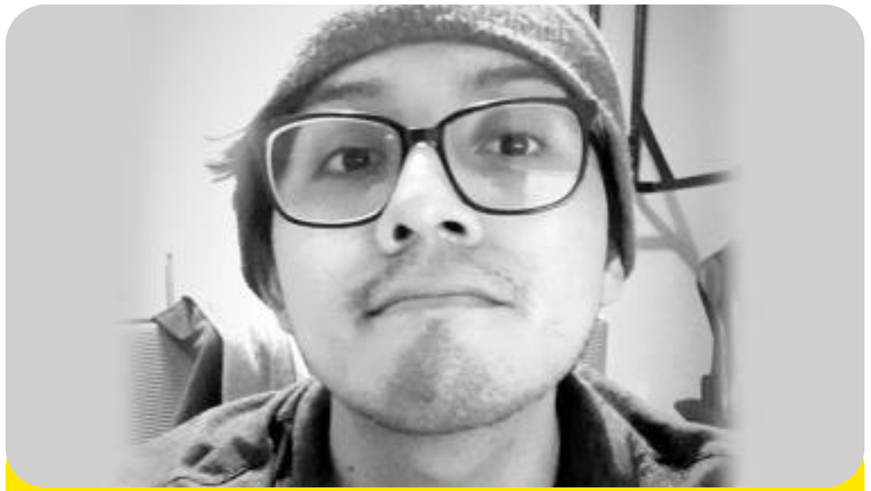
PUCP

DETECCIÓN DE MASCARILLAS EN TIEMPO REAL

Desarrollo en aplicaciones con visión artificial

GRUPO DE TRABAJO

TAMARA
Reategui



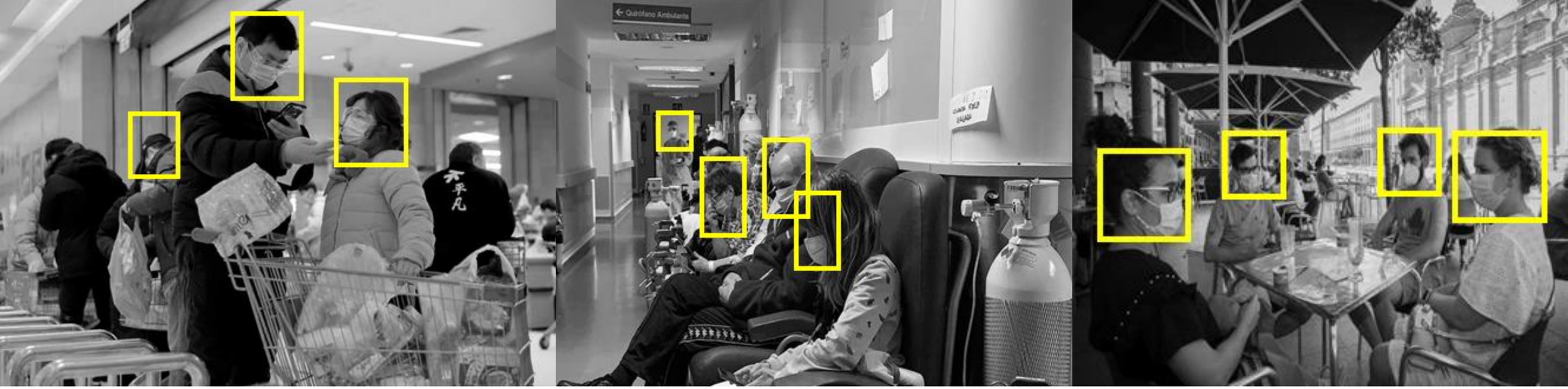
ALEXIS
Coronado

2020

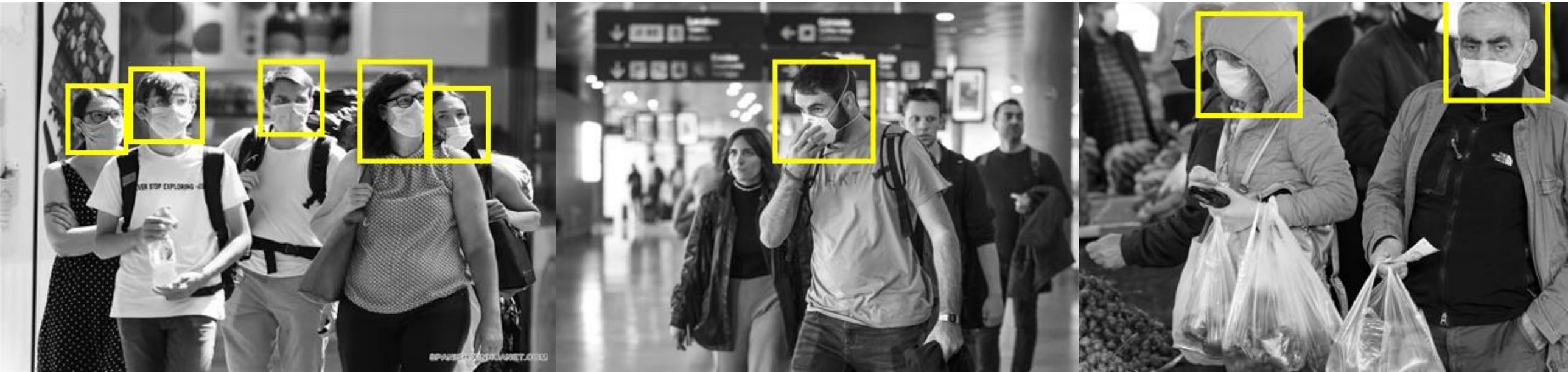
SUCESOS

- COVID 19
- Iniciativas en distintos rubros
- Crecimiento de la conectividad y aplicaciones.
- Aplicaciones usando inteligencia artificial.

GLOBAL PANDEMIC



Detección de mascarillas en ambientes de alta concurrencia



DESARROLLOS

LOGMASK: Servicios para la deteccion. de personas con mascarillas

https://www.logmask.com/es#mask_detection

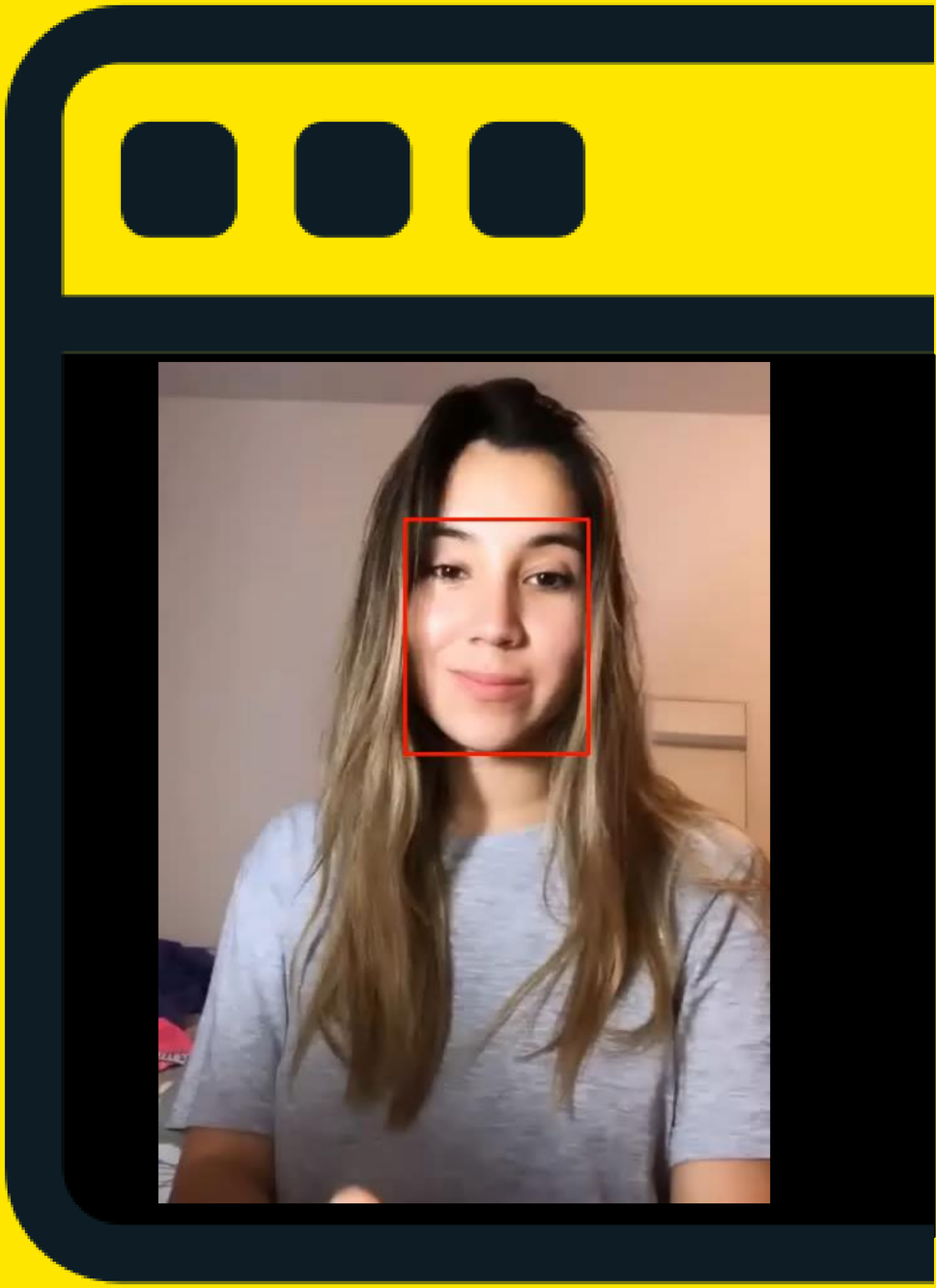
TRYO.LAB: Detección de personas con mascarillas en calles

https://youtu.be/SQCHaHkPoSQ

Solución

Detector de mascarillas en tiempo real

- Estructura MobileNetV2
- Estructura R CNN



1

MobileNetV2

CóMO FUNCIONA

In a nutshell

1

Leer frame del video

Lectura de frame en el que se analizara la presencia de rostros con mascarilla

2

Detectar objeto

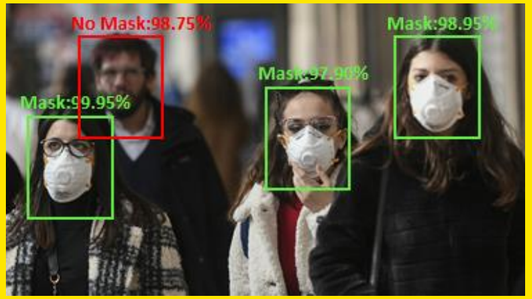
Face detector

Modelo : SSD, red neuronal pre entrenado para la detección de rostros.

Output: Array con ROIs de la imagen (faces) y sus ubicaciones (locs) que generan los bounding boxes

3

Clasificar



Mask detector

Modelo : MobileNetV2, red neuronal que clasifica en dos clases with mask o without mask

Output: Score entre 0 y 1 que significa la probabilidad del uso o no de la mascarilla en ese ROI

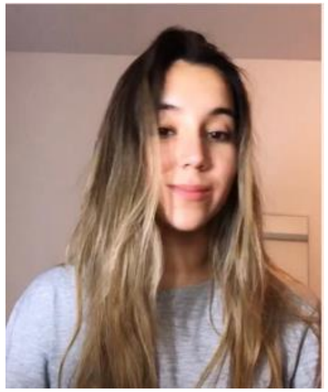
4

Repetir el proceso para cada frame

Repetir el proceso en cada frame al detectar al menos un rostro hasta el break

Arquitectura

Frame



Face detector

- Modelo: SSD
- Files

. **prototxt :** define el modelo de la arquitectura (ejem. las capas)
. **caffemodel:** contiene los pesos de las capas.

- Cargar

facenet = cv2.dnn.readnet (prototxtPath, weightsPath)

Mask detector

TRAINING

- Dataset
- Data augmentation
- Fine tuning set up
  - Cargar MobilenetV2
  - Crear nueva capa FC
  - Freeze
- Compilar
  - Optimizador (ADAM/SGD)
  - Loss: Binary-crossentropy
- Entrenamiento
- predicción
- Plot

APPLYING

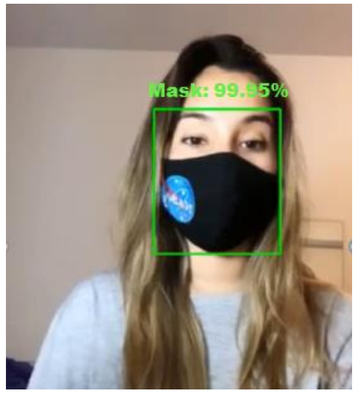
Aplicar el clasificador de deteccion de mascarillas a cada ROI para determinar la existencia de rostros mascarillas.

Despliegue

La función detect-and-predict mask acepta tres paramatros:

- Frame: un frame del stream
- faceNet: modelo usado para detectar rostros.
- maskNet: modelo para clasificador la detección de mascarillas

Results



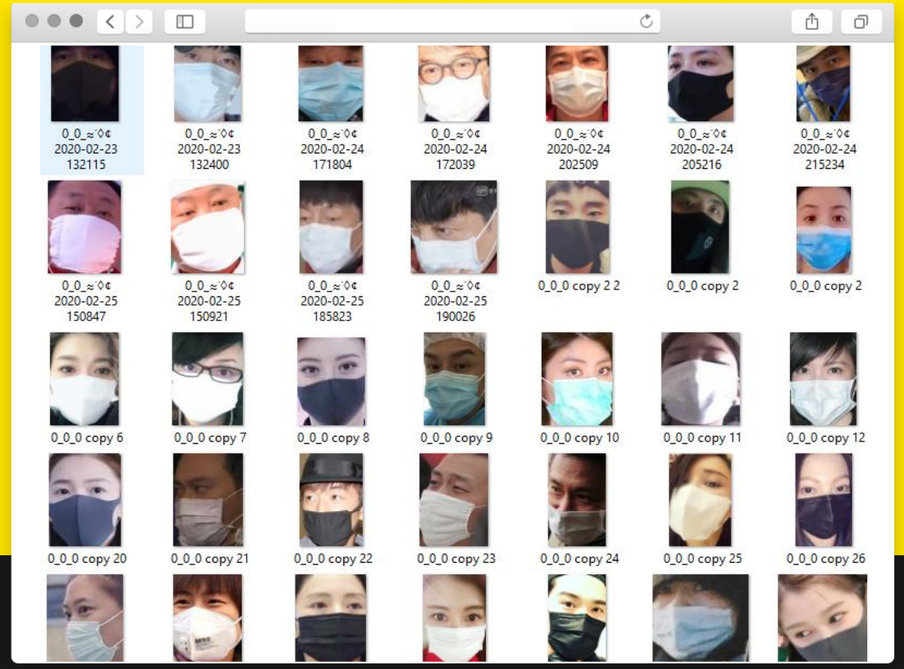
TRAINING

▶ DATASETS

1915

Imágenes con mascarilla

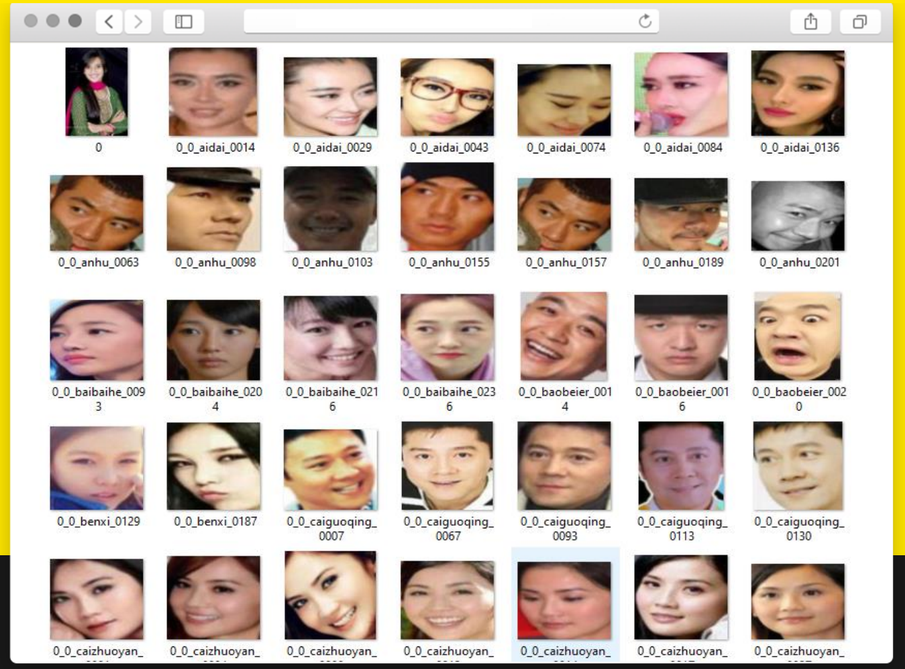
test size = 0.2



1918

Imágenes sin mascarilla

test size = 0.2



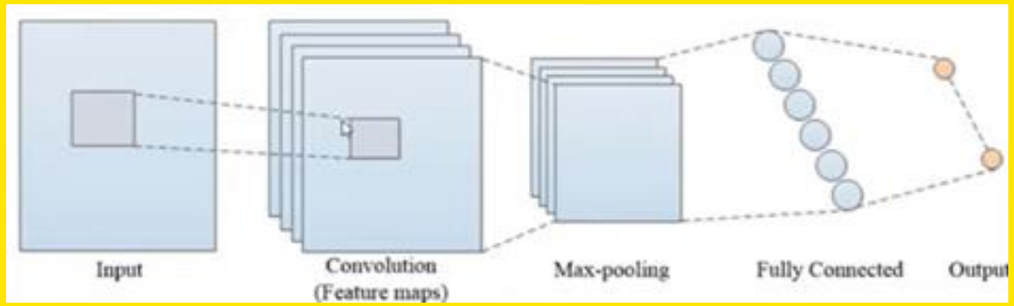
TRAINING

▶ Classification report

TRAINING

- Dataset
- Data augmentation
- Fine tuning set up
  - Cargar MobilenetV2
  - Crear nueva capa FC
  - Freeze
- Compilar
  - Optimizador
  - Loss: Binary-crossentropy
- Entrenamiento
- predicción
- Plot

.



COMPILAR:

```
Epoch 1/20
34/34 [==============================] - 30s 885ms/step - loss: 0.6431 - accuracy: 0.6676 - val_loss: 0.3696 - val_accuracy: 0.8242
Epoch 2/20
34/34 [==============================] - 29s 853ms/step - loss: 0.3507 - accuracy: 0.8567 - val_loss: 0.1964 - val_accuracy: 0.9375
Epoch 3/20
34/34 [==============================] - 27s 800ms/step - loss: 0.2792 - accuracy: 0.8820 - val_loss: 0.1383 - val_accuracy: 0.9531
Epoch 4/20
34/34 [==============================] - 28s 814ms/step - loss: 0.2196 - accuracy: 0.9148 - val_loss: 0.1306 - val_accuracy: 0.9492
Epoch 5/20
34/34 [==============================] - 27s 792ms/step - loss: 0.2006 - accuracy: 0.9213 - val_loss: 0.0863 - val_accuracy: 0.9688
...
Epoch 16/20
34/34 [==============================] - 27s 801ms/step - loss: 0.0767 - accuracy: 0.9766 - val_loss: 0.0291 - val_accuracy: 0.9922
Epoch 17/20
34/34 [==============================] - 27s 795ms/step - loss: 0.1042 - accuracy: 0.9616 - val_loss: 0.0243 - val_accuracy: 1.0000
Epoch 18/20
34/34 [==============================] - 27s 796ms/step - loss: 0.0804 - accuracy: 0.9672 - val_loss: 0.0244 - val_accuracy: 0.9961
Epoch 19/20
34/34 [==============================] - 27s 793ms/step - loss: 0.0836 - accuracy: 0.9710 - val_loss: 0.0440 - val_accuracy: 0.9883
Epoch 20/20
34/34 [==============================] - 28s 838ms/step - loss: 0.0717 - accuracy: 0.9710 - val_loss: 0.0270 - val_accuracy: 0.9922
```

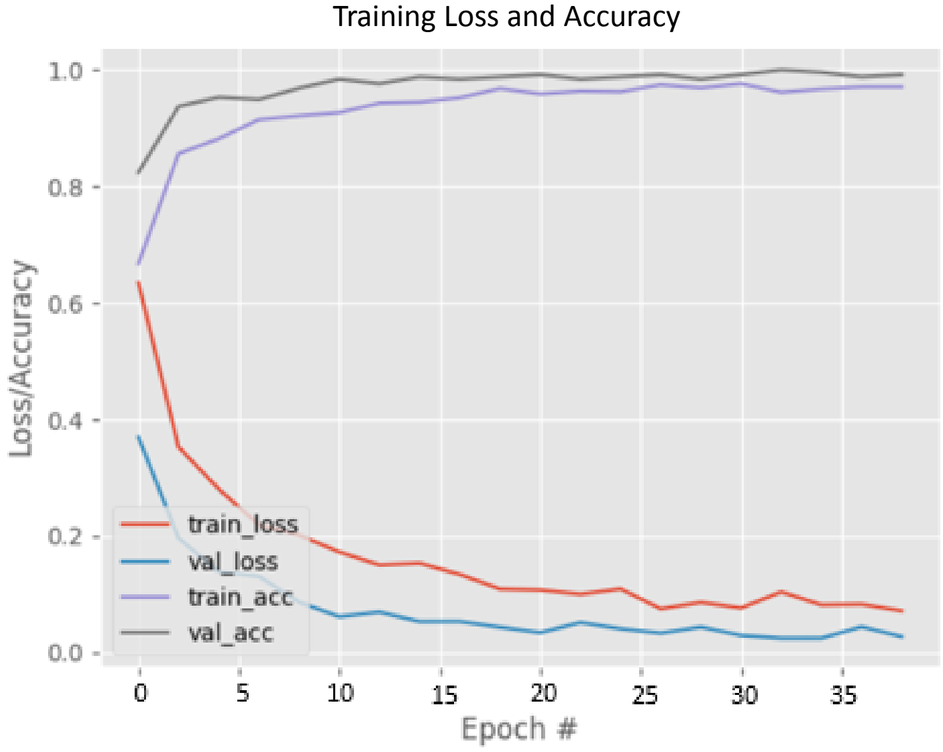
```
[INFO] evaluating network...
              precision    recall  f1-score   support

   with_mask       0.99      1.00      0.99       383
without_mask       1.00      0.99      0.99       384

    accuracy                           0.99       767
   macro avg       0.99      0.99      0.99       767
weighted avg       0.99      0.99      0.99       767

[INFO] saving mask detector model...
```

TRAINING

Gráficas

1

Optimizador: SGD
Lr: 1e-2
Epochs: 20
BS: 32
Loss: binary_crossentropy
Metric: Accuracy



2

Optimizador: ADAM
Lr: 1e-4
Epochs: 40
BS: 32
Loss: binary_crossentropy
Metric: Accuracy

② ▼

Faster R CNN

CóMO FUNCIONA

In a nutshell

1

Leer frame del video

Lectura de frame en el que se analizara la presencia de rostros con mascarilla

2

Detectar objeto

Face detector

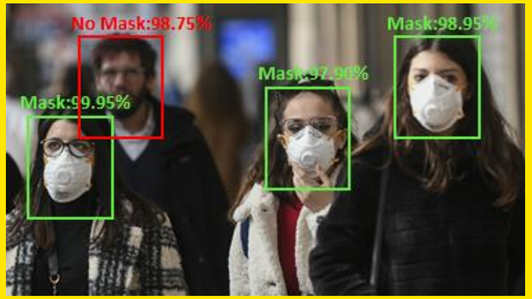
Modelo : RPN

Se encarga de proponer las ROI donde haya un objeto presente, las cuales servirán de input para la siguiente capa, para este caso rosotros.

Output: Tensor con los BBS de las ROI con rostros

3

Clasificar



Mask detector

Modelo :

- Clasificador SoftMax
- Regressor con regularización para los BBs

Output: Tensor

-Score (0-1)

-Coordenadas de los BBs

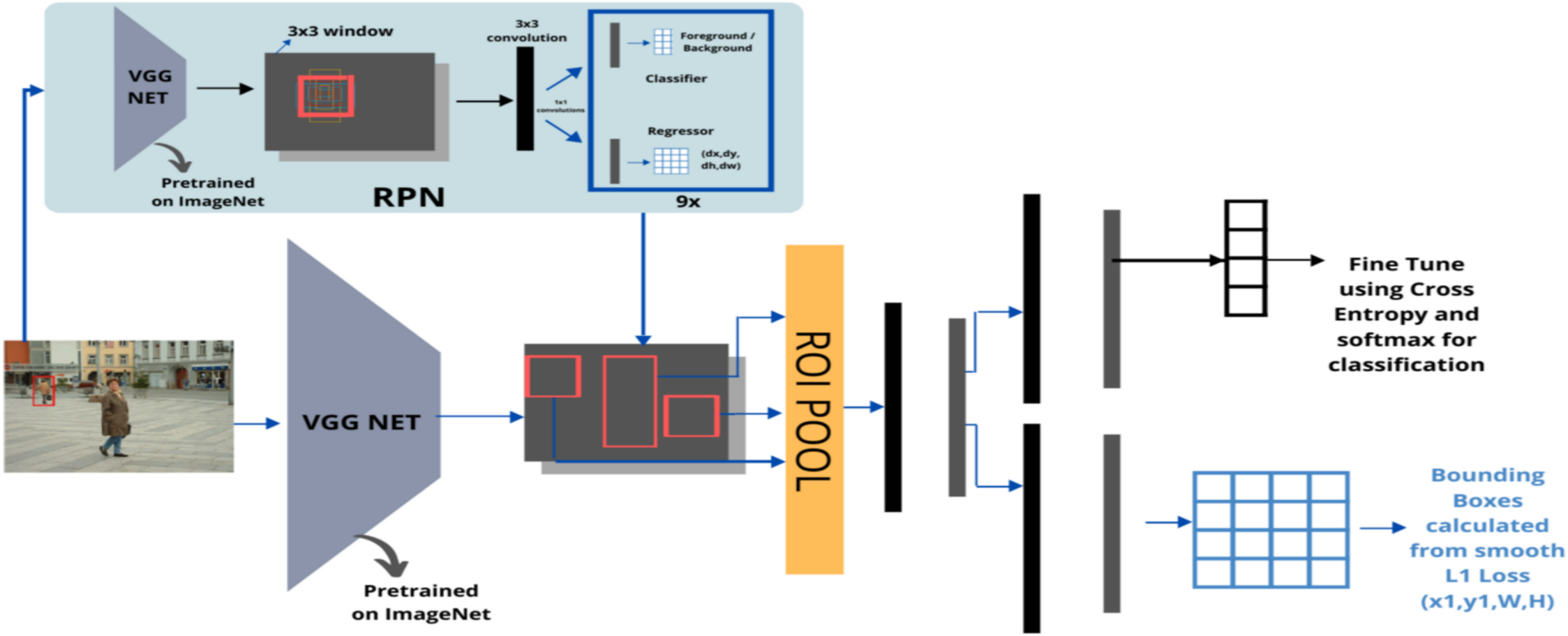
- Categoría del objeto detectado del BB

4

Repetir el proceso para cada frame

Repetir el proceso en cada frame al detectar al menos un rostro hasta el break

Arquitectura



Fuente : https://towardsdatascience.com/understanding-fast-r-cnn-and-faster-r-cnn-for-object-detection-adbb55653d97

TRAINING

▶ Classification report

TRAINING

- Dataset
- Fine tuning set up
    - SGD : Lr = 0.005 , momentum = 0.9
      weight decay = 0.0005

- Num. Epochs = 5 , 10 ,15, 25, los mejores resultados se obtuvieron con 10 epocas
- Batch Size : 5

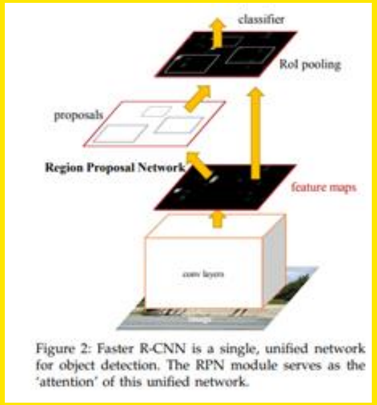
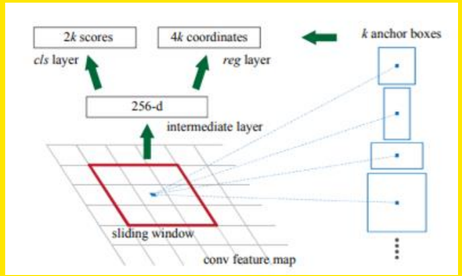


Figure 2: Faster R-CNN is a single, unified network for object detection. The RPN module serves as the 'attention' of this unified network.



COMPILAR:

```
Epoch: [4]  [170/171] lr: 0.000500
loss: 0.1688 (0.1854)
loss_classifier: 0.0632 (0.0613)
loss_box_reg: 0.0955 (0.1173)
loss_objectness: 0.0009 (0.0017)
loss_rpn_box_reg: 0.0032 (0.0051)
```

```
Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.475
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.806
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.495
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = 0.326
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.480
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.528
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.302
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.574
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.617
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = 0.519
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.615
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.661
```

```
Epoch: [9]  [170/171]
 loss: 0.1491 (0.1686)
loss_classifier: 0.0567 (0.0555)
loss_box_reg: 0.0912 (0.1066)
loss_objectness: 0.0004 (0.0016)
loss_rpn_box_reg: 0.0028 (0.0048)
```

Resultados

Referencias

Tutorial de Transfer learning

•https://pytorch.org/tutorials/intermediate/torchvision_tutorial.html

Notebook Inicial de Kaggle:

•https://www.kaggle.com/daniel601/pytorch-fasterrcnn

•

Archivos utilitarios para cálculo de Loss en test:

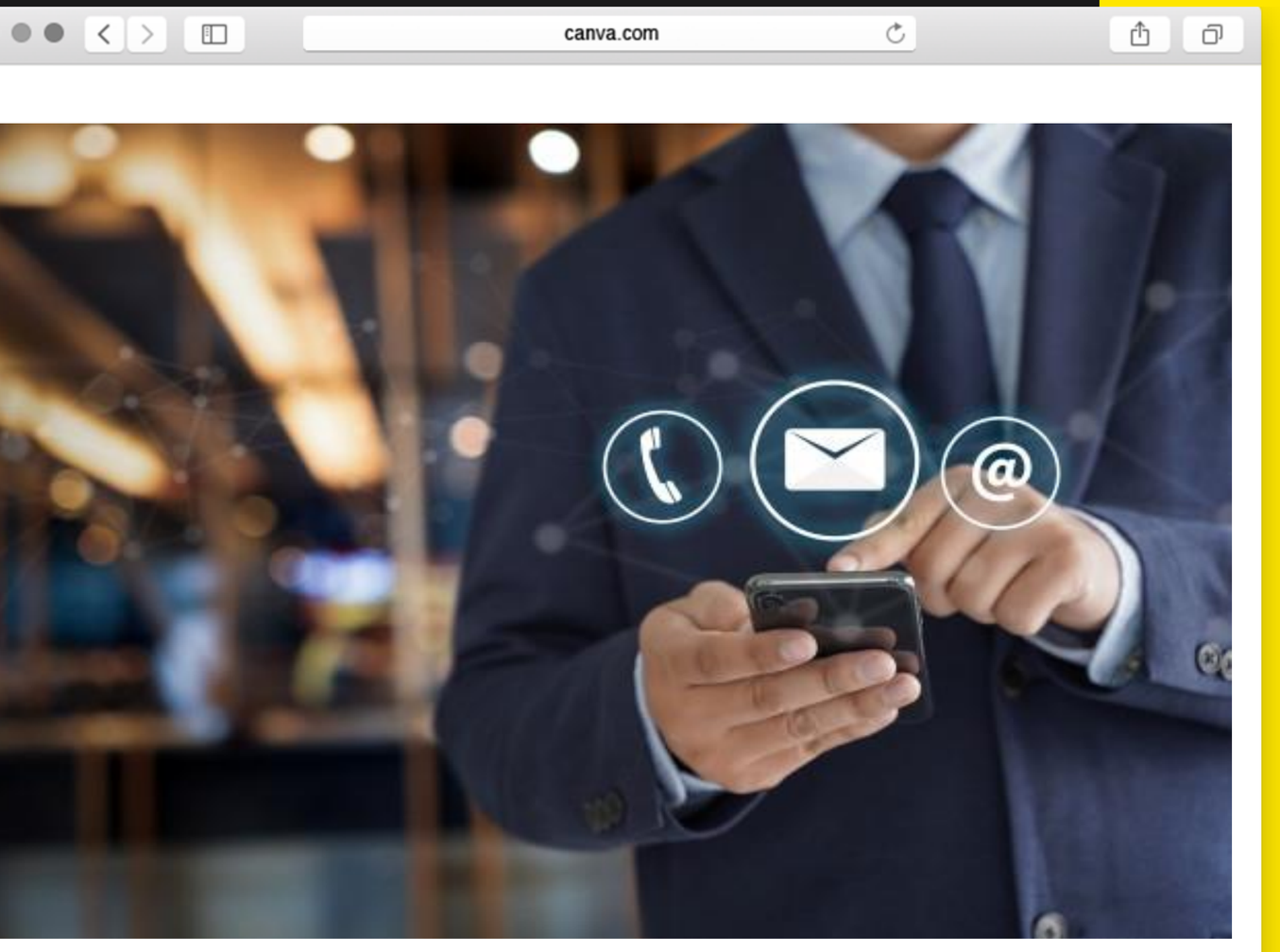
•https://github.com/pytorch/vision

Tutoriales Y Explicaciones Youtube:

•https://www.youtube.com/playlist?list=PLqnslRFeH2UrcDBWF5mfPGpqQDSta6VK4 *(Canal muy recomendado de tutoriales referentes a ml y dl)*

•https://www.youtube.com/watch?v=v5bFVbQvFRk&ab_channel=ArdianUmam

Contáctanos



TAMARA REATEGUI
tam.reategui@gmail.com

ALEXIS CORONADO
itz_@hotmail.es
https://www.linkedin.com/in/alexis-Coronado-Ortiz-ab5b08117