

<> Code

Issues

Pull requests

Actions

Projects

Security

Insights



0 stars

0 forks

2 watching

Branches

Tags

Activity

Public repository

1 Branch

0 Tags

Go to file

Go to file

+

Add file

Code

...

AlmaOlmos

Update README.md

2b88243 · 4 years ago

Documentacion	Cambiar estructura de proyecto	4 years ago
Imagenes	Cambiar estructura de proyecto	4 years ago
Presentacion	Cambiar estructura de proyecto	4 years ago
README.md	Update README.md	4 years ago
db.py	primer version	4 years ago
fingerprint_db.py	Update fingerprint_db.py	4 years ago
train_fingerprint.py	primer version	4 years ago

## README



# CNN para reconocimiento de huellas dactilares

Este proyecto fue desarrollado durante la 2da edición de Saturdays.AI Guadalajara

Proyecto para el reconocimiento de huellas de cuerpos postmortem y validación de huella dentro de base de datos mysql.

## Instalación

Usar entorno virtual con python 3.7, Linux Ubuntu.

- numpy
- matplotlib
- keras
- sklearn
- imgaug
- pymysql
- paramiko
- sshtunnel
- PIL

## Como usar

1. Agregar variables de entorno para conexión con instancia de base de datos mysql:

```
export SQL_HOSTNAME=<>
export SQL_USERNAME=<>
export SQL_PASSWORD=<>
export DB=<>
export DB_PORT=<>
export SSH_HOST=<>
export SSH_USER=<>
export SSH_PORT=<>
```



2. Se establece el path cambiando la línea 14 de train\_fingerprint.py
3. En fingerprint\_db se cambia línea 31 y 38 con la tabla correspondiente.
4. Cambiar el nombre de las columnas en las queries
5. Conexión:

```
python fingerprint_db.py
```



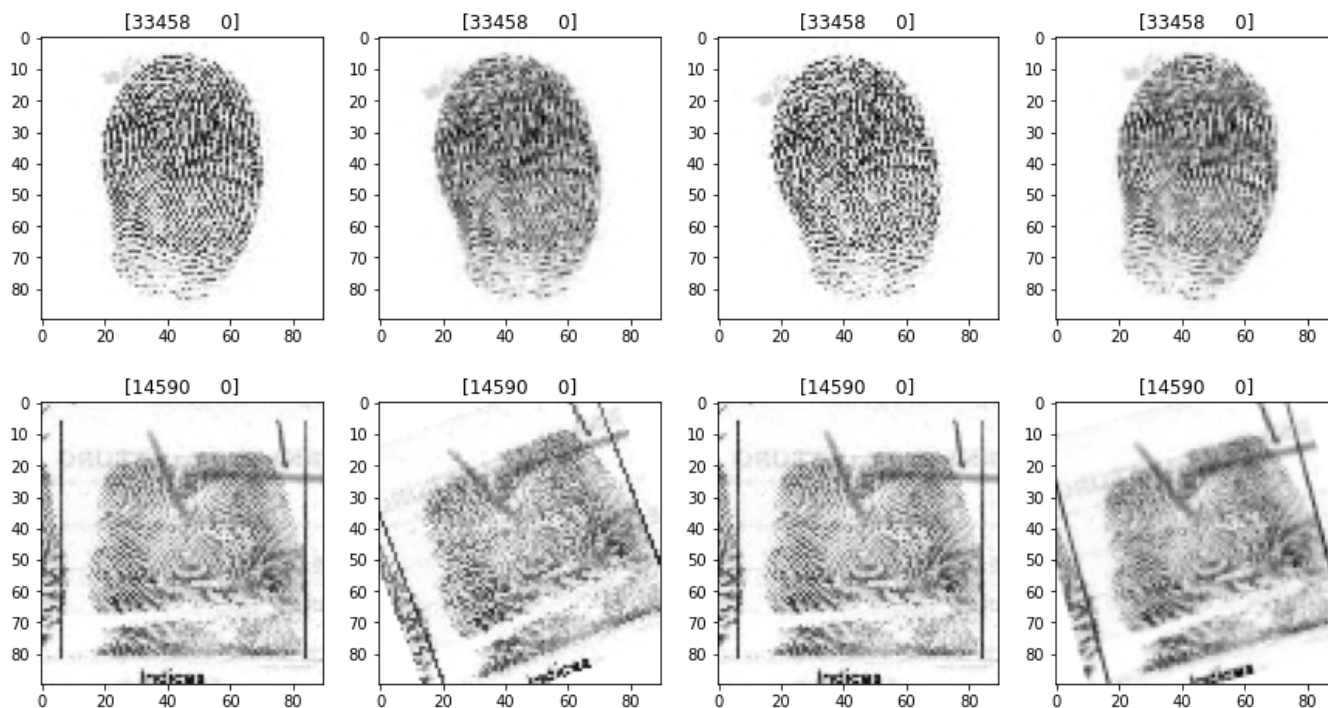
6. Entrenamiento:

```
python train_fingerprint.py
```




## Descripción

El modelo es una red neuronal convolucional que se entrenó con 1k imágenes reales y 3k generadas de forma aumentada. La base de datos no es de uso público y es necesario que el dataset que se utilice tenga el formato wsq.



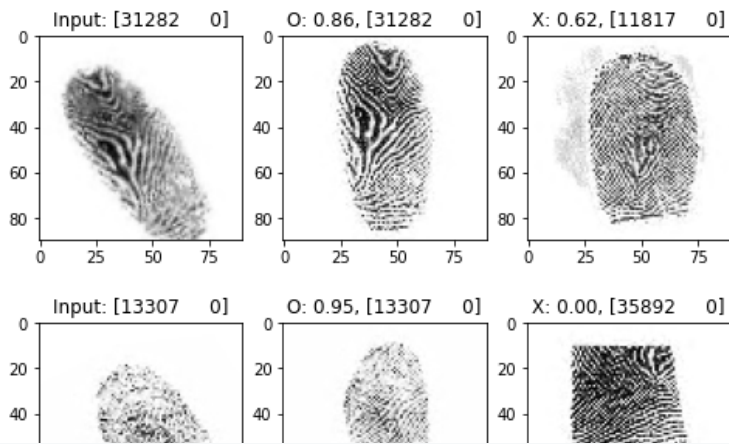
## La arquitectura de la red



Layer (type)	Output Shape	Param #	Connected to
input_4 (InputLayer)	(None, 90, 90, 1)	0	
input_5 (InputLayer)	(None, 90, 90, 1)	0	
model_3 (Model)	(None, 22, 22, 32)	9568	input_4[0][0] input_5[0][0]
subtract_2 (Subtract)	(None, 22, 22, 32)	0	model_3[1][0] model_3[2][0]
conv2d_6 (Conv2D)	(None, 22, 22, 32)	9248	subtract_2[0][0]
max_pooling2d_6 (MaxPooling2D)	(None, 11, 11, 32)	0	conv2d_6[0][0]
flatten_2 (Flatten)	(None, 3872)	0	max_pooling2d_6[0][0]
dense_3 (Dense)	(None, 64)	247872	flatten_2[0][0]
dense_4 (Dense)	(None, 1)	65	dense_3[0][0]
Total params: 266,753			
Trainable params: 266,753			
Non-trainable params: 0			

## Resultados

La salida del modelo es la imagen original con su etiqueta en el lado izquierdo, en el centro la imagen con mayor porcentaje de parentesco y en la derecha una imagen con menor parentesco.



## Releases

No releases published

## Packages

No packages published

## Contributors 2



**AlmaOlmos** Alma Olmos



**nachovazquez98** Ignacio David Vazquez Perez

## Languages

● Python 100.0%