

Práctica de reconocimiento de gestos

Sistemas de Interacción Persona-Computador

Curso 2020/2021

Rafael Arnay del Arco

Introducción

- El reconocimiento de gestos permite a los seres humanos interactuar naturalmente con una máquina sin usar dispositivos físicos de entrada.
- Se plantea un escenario habitual de interacción persona-computador en el que el usuario se encuentra sentado frente al ordenador y se dispone de una cámara con la que podemos capturar imágenes del mismo.

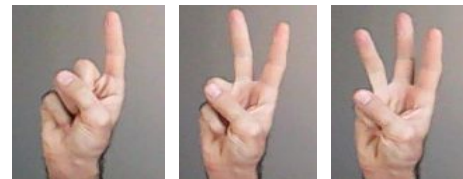


Objetivos

- Desarrollar un sistema de reconocimiento automático de gestos de la mano.

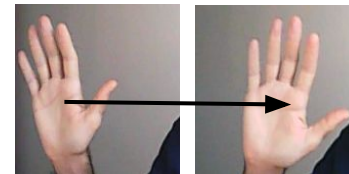
El sistema debe ser capaz de:

- Contar los dedos que el usuario tenga extendidos
- Reconocer gestos de la mano



- Objetivos opcionales:

- Pintar con el movimiento de la mano
- Reconocer desplazamientos de la mano
- Otras ideas propias...



Metodología

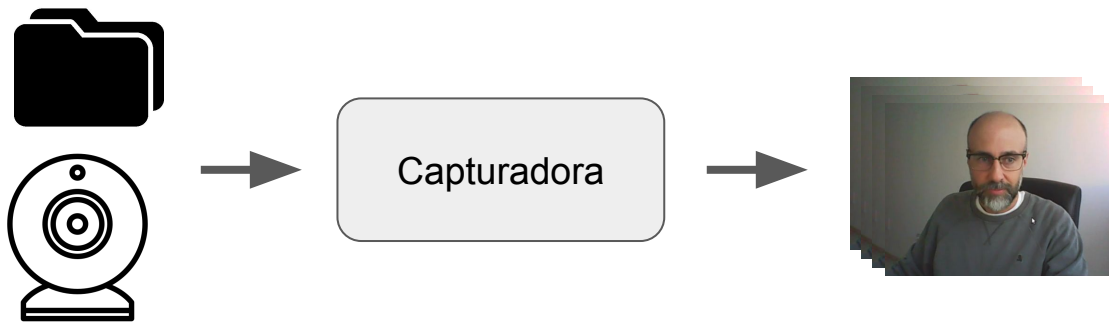
¿Qué es OpenCV?



- Es una librería de visión artificial y aprendizaje automático.
- Se diseñó para acelerar la implantación de sistemas automáticos de percepción en productos comerciales.
- Dispone de más de 2500 algoritmos (Reconocimiento de caras, identificación de objetos, clasificación de comportamiento humano en vídeos, seguimiento de objetos móviles, etc...)
- Funciona en C, C++, Python, Java y MATLAB y soporta Windows, Linux, Android y MAC OS

Metodología

1. Leer imágenes de la cámara o de archivo de vídeo



C++

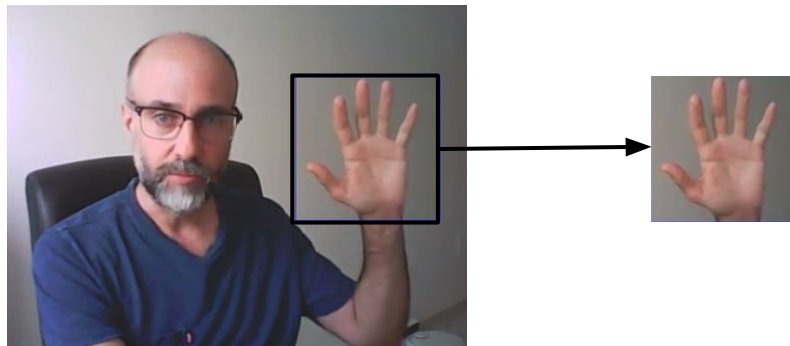
```
VideoCapture cap;  
cap.open(0);
```

Python

```
cv2.VideoCapture(0)
```

Metodología

2. Seleccionar una región de interés para su procesamiento



C++

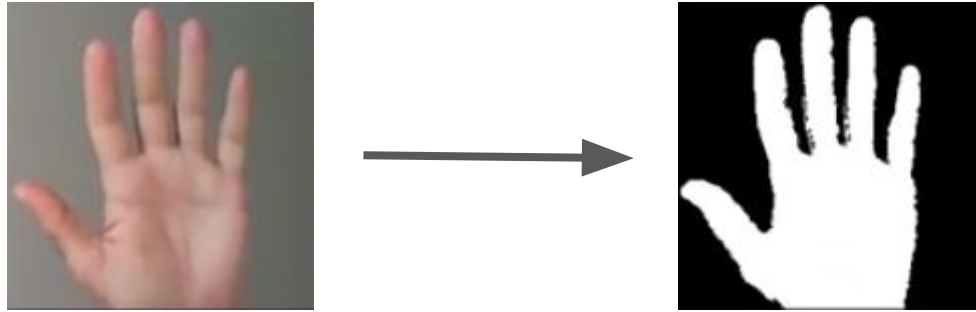
```
Rect rect(400,100,200,200);  
roi = img(rect)
```

python

```
roi = frame[y1:y2,x1:x2,:]
```

Metodología

3. Sustracción de fondo en la región de interés

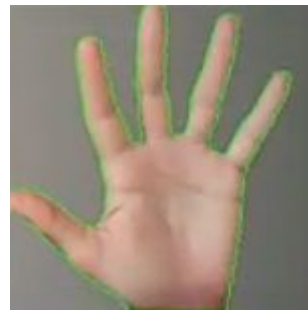


[C++/Python](#)

```
createBackgroundSubtractorMOG2()
```

Metodología

4. Detección del contorno de la mano

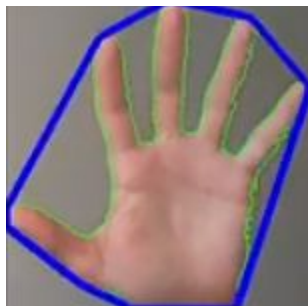


[C++/Python](#)

```
findContours(...)
```


Metodología

5. Detección de la malla convexa

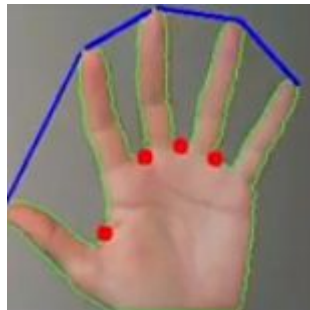
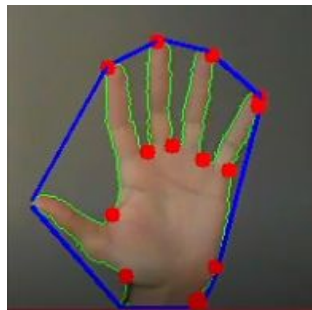


[C++/Python](#)

```
convexHull(...)
```

Metodología

6. Detección y filtrado de defectos de convexidad



[C++/Python](#)

```
convexityDefects(...)
```

Criterios de evaluación

- Práctica individual o en grupos de dos.
- Subir el código y un vídeo en el que se vea a los miembros del grupo y la funcionalidad implementada. También deben explicar el código en el vídeo.

Rúbrica:

- | | |
|--|-----------------|
| ● Hasta el filtrado correcto de los defectos de convexidad | 5 |
| ● Contar correctamente los dedos | +1 |
| ● Detectar gesto | +1 (cada gesto) |
| ● Pintar, detectar movimiento, otros... | +1 (cada uno) |
| ● Correcta estructura del código y comentarios suficientes | +1 |