# INVESTIGACIÓN OPERATIVA SUPERIOR

visión por computadora (para investigación operativa)

**Presencial** 

una introducción a ...

visión por computadora

visión por computadora clásica visión por computadora basada en redes neuronales



# objetos

# métodos









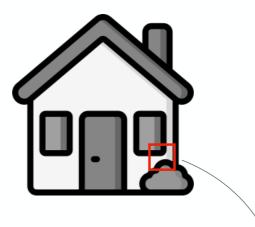
imágenes y videos

matemáticos y estadísticos

# imágenes

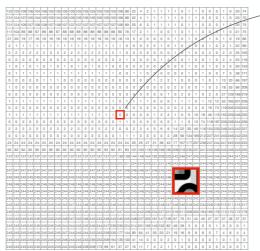


# imágenes



recorte de 35 x 35 px

# imágenes

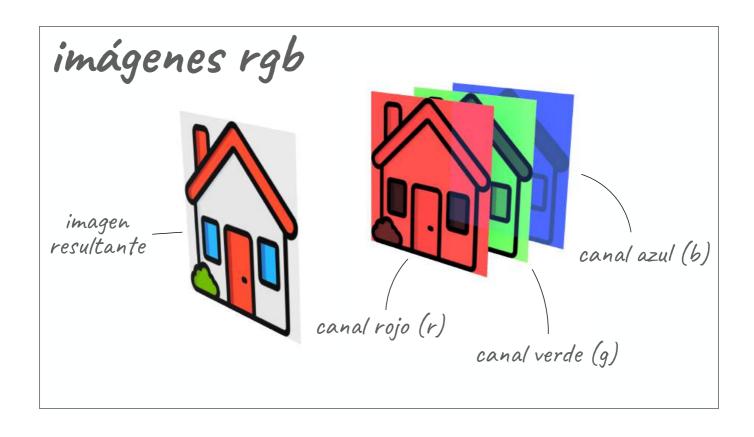


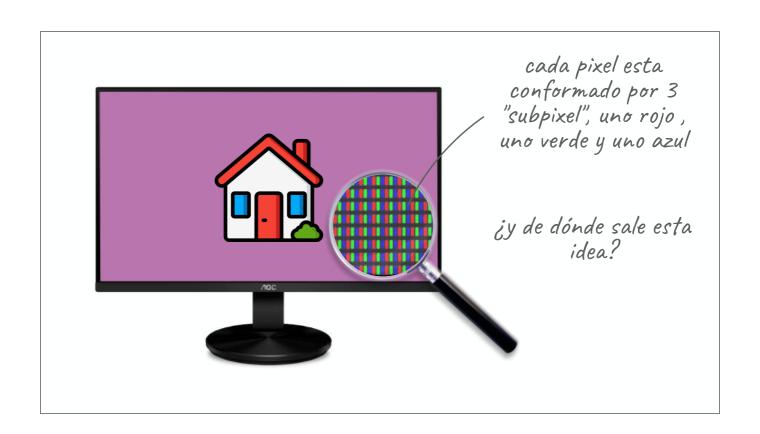
intensidad de color

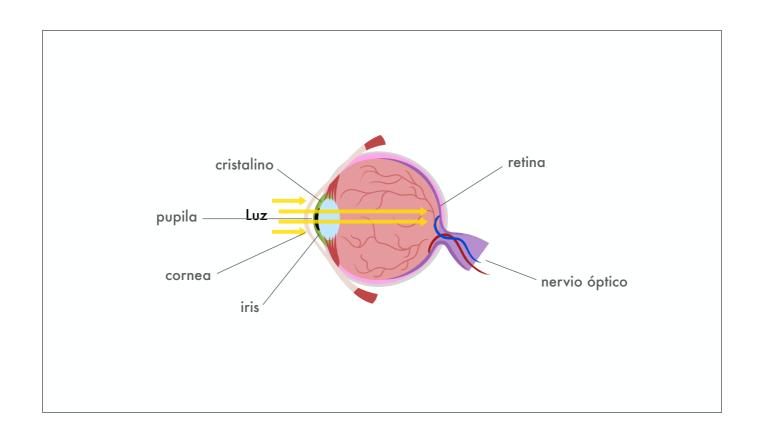


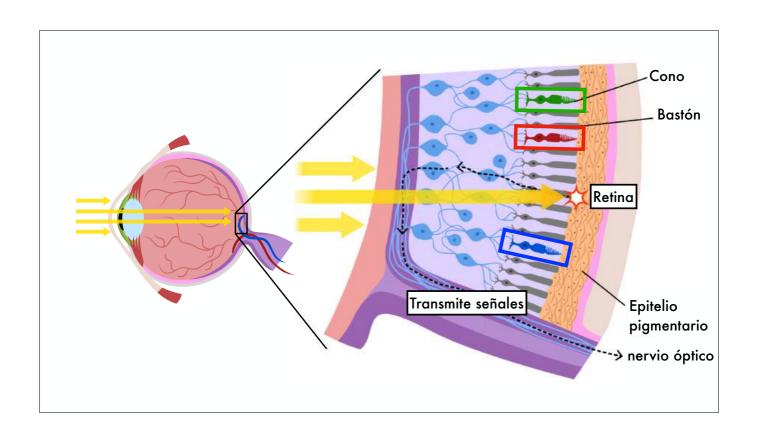












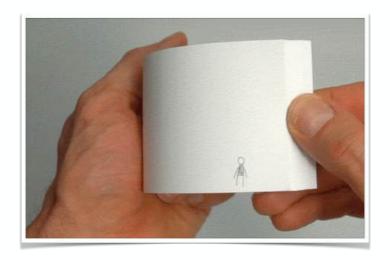
# vecchtevæmet eeltrateaja con imágemepinelpython...



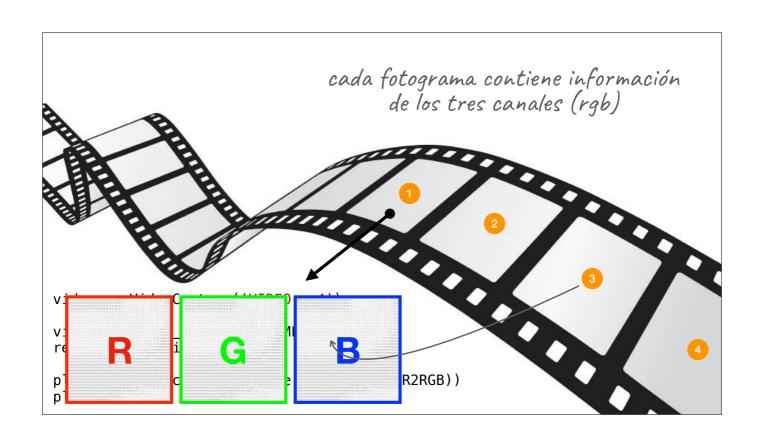
DEMO 1

jy videos?



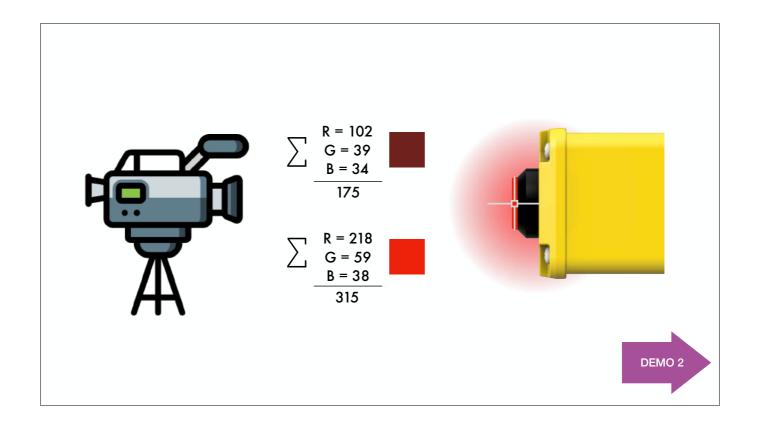


es lo mismo pero se forma con varias imágenes encadenadas







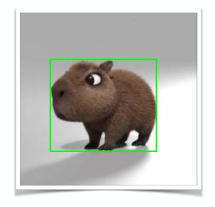


# algunos métodos de visión por computadora





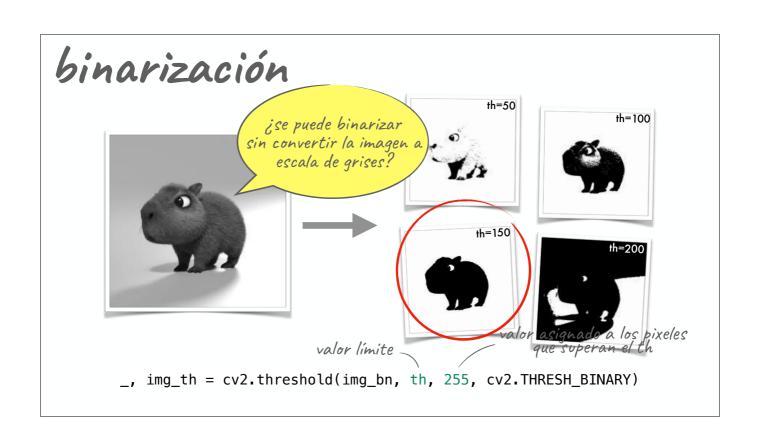
# objetivo



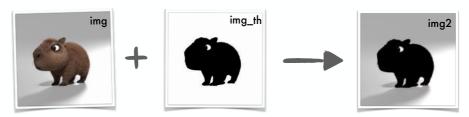
detectar la ubicación precisa del carpincho

# primer paso ... (trabajar con un solo canal)

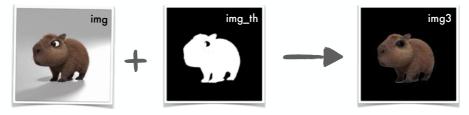
img\_bn = cv2.cvtColor(img, cv2.COLOR\_BGR2GRAY)



# máscaras

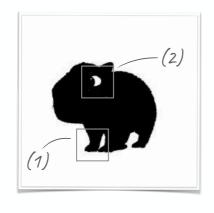


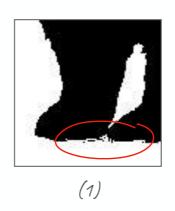
 $img2 = cv2.bitwise\_and(img, img, mask=img\_th)$ 

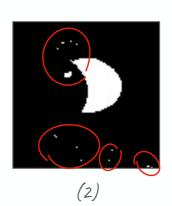


img3 = cv2.bitwise\_and(img, img, mask=cv2.bitwise\_not(img\_th))

# procesamiento morfológico





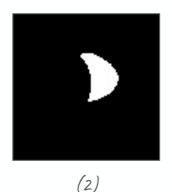




(erosión)







img = cv2.erode(img, np.ones((5, 5), np.uint8), iterations = 1)

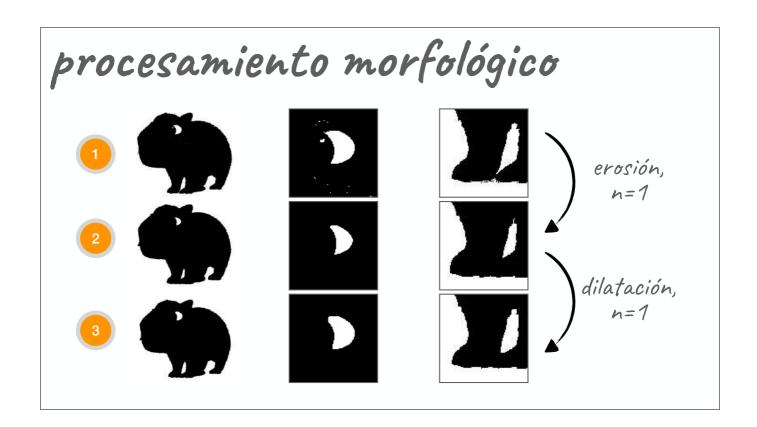
# procesamiento morfológico (dilatación)

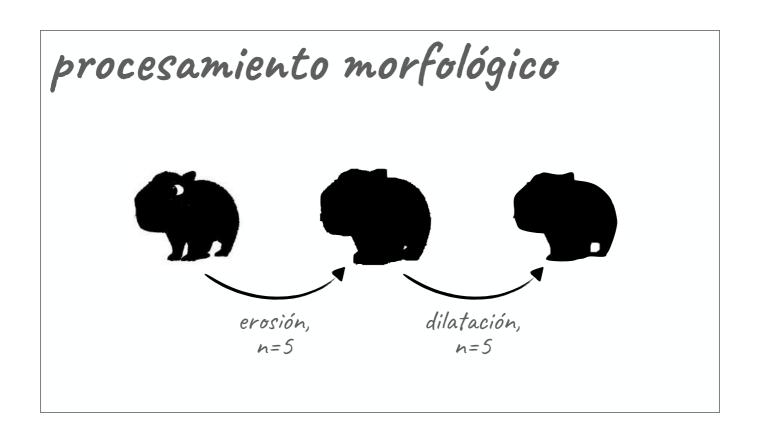






img = cv2.dilate(img, np.ones((5, 5), np.uint8), iterations = 1)





# detección de bordes

devuelve una lista de python con los contornos encontrados



contornos, \_ = cv2.findContours(imagen, modo, método)

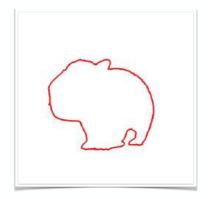
imagen binarizada

cv2.drawContours(imagen, contornos, id contornos, c

dibuja los contornos en la imagen

el objeto debe ser blanco y el fondo negro cv2.bitwise\_not(img)

detección de bordes



contornos sobre una contornos sobre la imagen en blanco

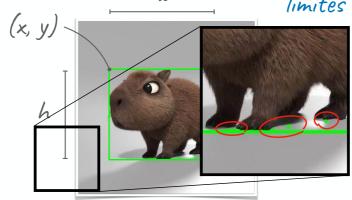


imagen original

# boundingboxes

x, y, w, h = cv2.boundingRect(contorno)

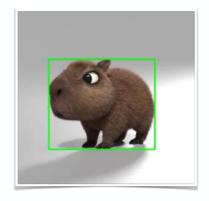
examina todos los puntos en el contorno para determinar los límites de la forma



# boundingboxes

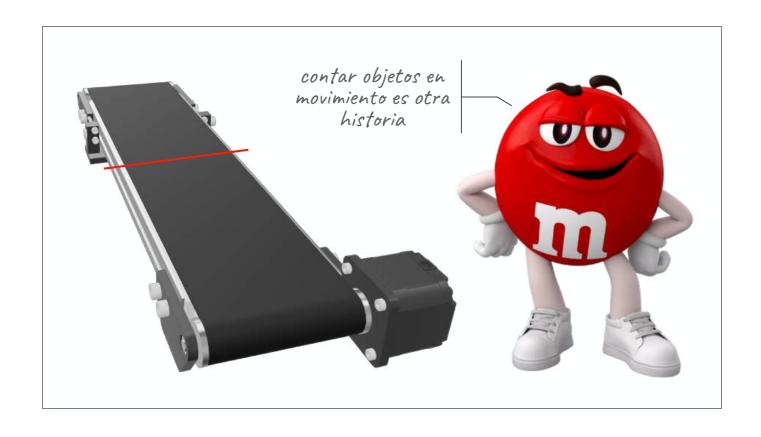
número de pixeles dentro de un contorno

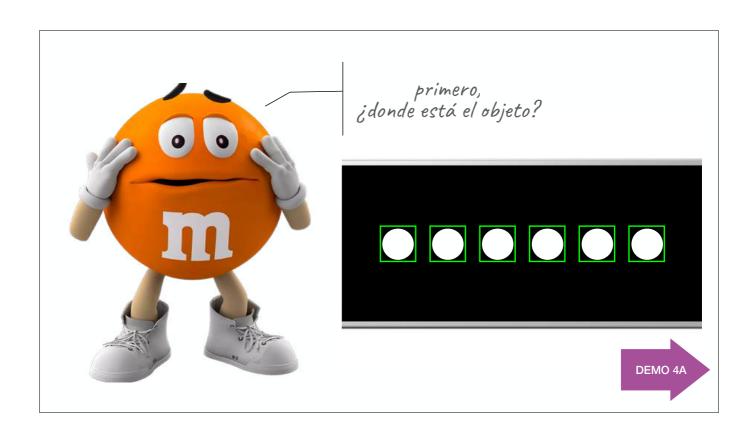
```
for c in contornos:
    if cv2.contourArea(c) > 50:
        x, y, w, h = cv2.boundingRect(contour)
        cv2.rectangle(img, (x, y), (x + w, y + h), (0, 255, 0), 3)
```



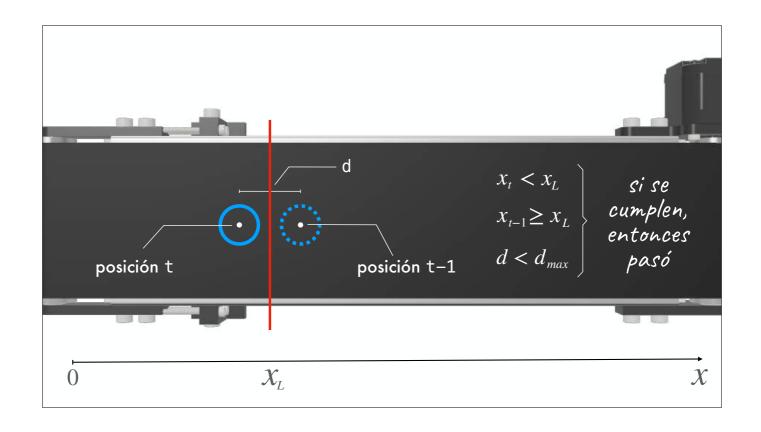
DEMO 3

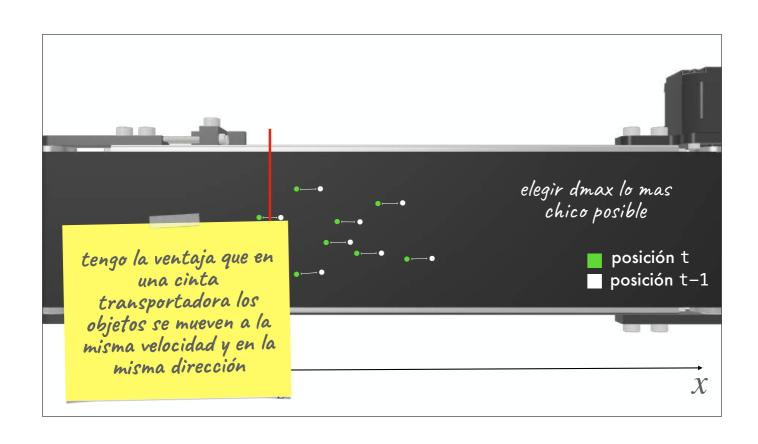
# otro ejemplo industrial ...





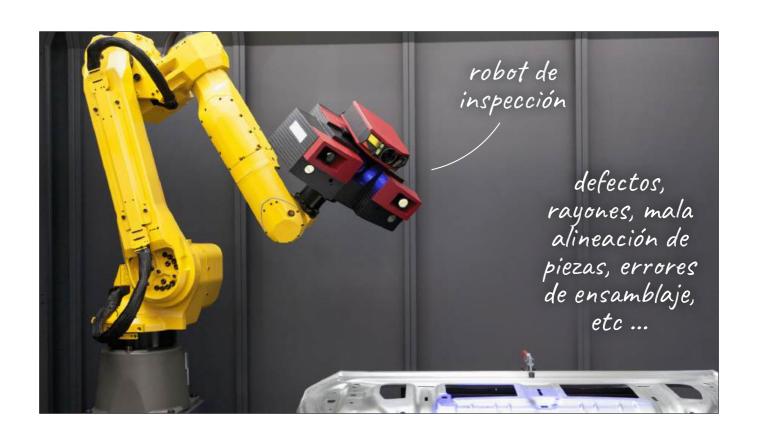




































# algunas practicas optativas ...

analizar los tres videos que se encuentran en el drive y obtener las distribuciones de la variable aleatoria





# bibliografía y otros ...

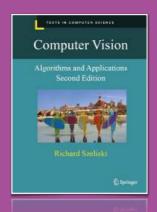
## [Python] OpenCV:

https://opencv.org/ https://docs.opencv.org/4.x/d9/df8/tutorial\_root.html

### [Videos]:

https://www.youtube.com/watch?v=oXlwWbU8l2o





## **Computer Vision**

Autor: Richard Szeliski Editorial: Springer



# INVESTIGACIÓN OPERATIVA SUPERIOR

jmuchas gracias!