

# **Tema 9: Detección, extracción de características, segmentación y reconocimiento de objetos**

## **Parte II**

# Índice

- Transformada de Hough
- Características SIFT
- Segmentación de imágenes
  - Algoritmo de las K-medias
  - Segmentación basada en regiones

# Índice

- Reconocimiento de objetos
  - Reconocimiento mediante características
  - Reconocimiento Machine Learning/Deep Learning
  - Reconocimiento de caras
- Reconocimiento de movimiento

# Introducción. Reconocimiento de objetos

- El reconocimiento de objetos consiste en, dado algún conocimiento (forma, apariencia, etc.) sobre uno o varios objetos y una imagen, encontrar qué objetos están en la imagen y dónde.
- El reconocimiento es un proceso difícil debido a:
  - Presencia de otros objetos no modelados
  - Cambio de iluminación
  - Cambio de punto de vista del objeto
  - Oclusión
  - Escala

# Introducción. Reconocimiento de objetos

- Ejemplos de uso:
  - Vehículos sin conductor
  - Visión o navegación robótica
  - Identificación de enfermedades en imágenes médicas
  - Reconocimiento facial

# Ejemplos: reconocimiento facial

- Se suele reconocer una posible posición en la imagen para una cara (mediante color de la piel, identificación de los ojos, etc.) y luego se reconoce la persona (con técnicas de aprendizaje)



# Ejemplos: reconocimiento facial

- Ejemplos de uso:
  - Desbloquear dispositivos móviles
  - Fuerzas de seguridad
  - Identificación de personas en redes sociales
  - Pagos
  - Agilizar procesos en aeropuertos, hospitales, etc.

# Introducción. Reconocimiento de objetos

- Clasificación vs Detección vs Localización
  - Técnicas para identificar objetos.
  - Clasificación: asignar una etiqueta
  - Localización: marcar con una caja un objeto en la imagen
  - Detección: combinación de las anteriores



# Introducción. Reconocimiento de objetos

- Técnicas:
  - Coincidencia de plantillas
  - Segmentación de imágenes y análisis de blobs
  - Reconocimiento de características
  - Machine Learning / Deep Learning

# Introducción. Reconocimiento de objetos

- Técnicas:
  - Coincidencia de plantillas
  - Segmentación de imágenes y análisis de blobs
  - Reconocimiento de características
  - Machine Learning / Deep Learning

# Reconocimiento mediante características

- Imaginemos que tenemos una imagen de un objeto a reconocer (modelo)
- Extraemos, por ejemplo, las características SIFT de dicha imagen. El objeto ahora es representado por sus características SIFT
- Ahora tenemos una nueva imagen (escena) donde queremos “buscar” ese objeto
- Extraemos los SIFT de esta nueva imagen
- Encontramos las correspondencias entre las características del modelo y de la imagen

# Reconocimiento mediante características

- Esto se puede hacer calculando la distancia euclídea del descriptor
- Para cada característica del modelo:
  - Encontramos la característica de la escena cuya distancia euclídea esté por debajo de un cierto umbral
  - Ahora tenemos una correspondencia entre los descriptores del modelo y de la escena



# Reconocimiento mediante características

- Ahora debemos encontrar la transformación entre el modelo y la escena
- Para simplificar, vamos a ver cómo se puede obtener la transformación 2D-2D afín:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

donde las variables  $m$  son los parámetros de rotación y escala y los  $t$  son los de traslación

# Reconocimiento mediante características

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

- Tenemos un sistema de varias ecuaciones con varias incógnitas, reescribimos la ecuación de arriba

$$\begin{bmatrix} x & y & 0 & 0 & 1 & 0 \\ 0 & 0 & x & y & 0 & 1 \\ & & \dots & & & \\ & & \dots & & & \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \\ t_x \\ t_y \end{bmatrix} = \begin{bmatrix} u \\ v \\ \vdots \end{bmatrix}$$

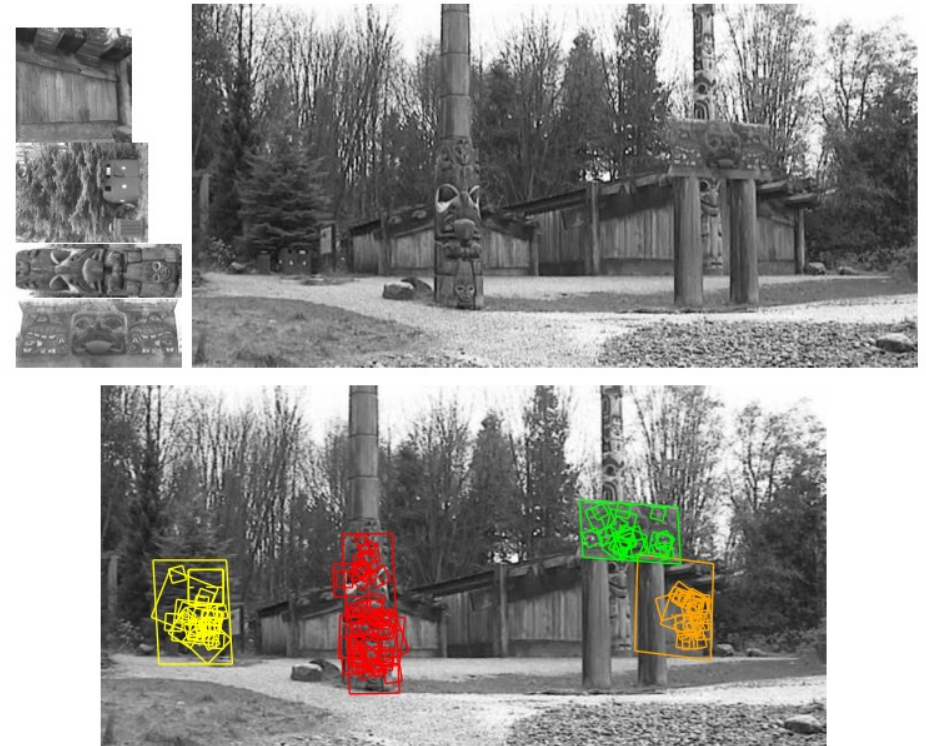
- Cada emparejamiento introduce dos nuevas filas a la primera y última matriz. Nombramos las matrices como:

$$\mathbf{Ax} = \mathbf{b}$$

# Reconocimiento características: transformación

- Resolvemos el sistema anterior mediante mínimos cuadrados
- Consiste en encontrar la matriz  $x$  que minimiza el error cuadrático medio entre todos los emparejamientos
- Se resuelve este sistema:

$$\mathbf{x} = [\mathbf{A}^T \mathbf{A}]^{-1} \mathbf{A}^T \mathbf{b},$$

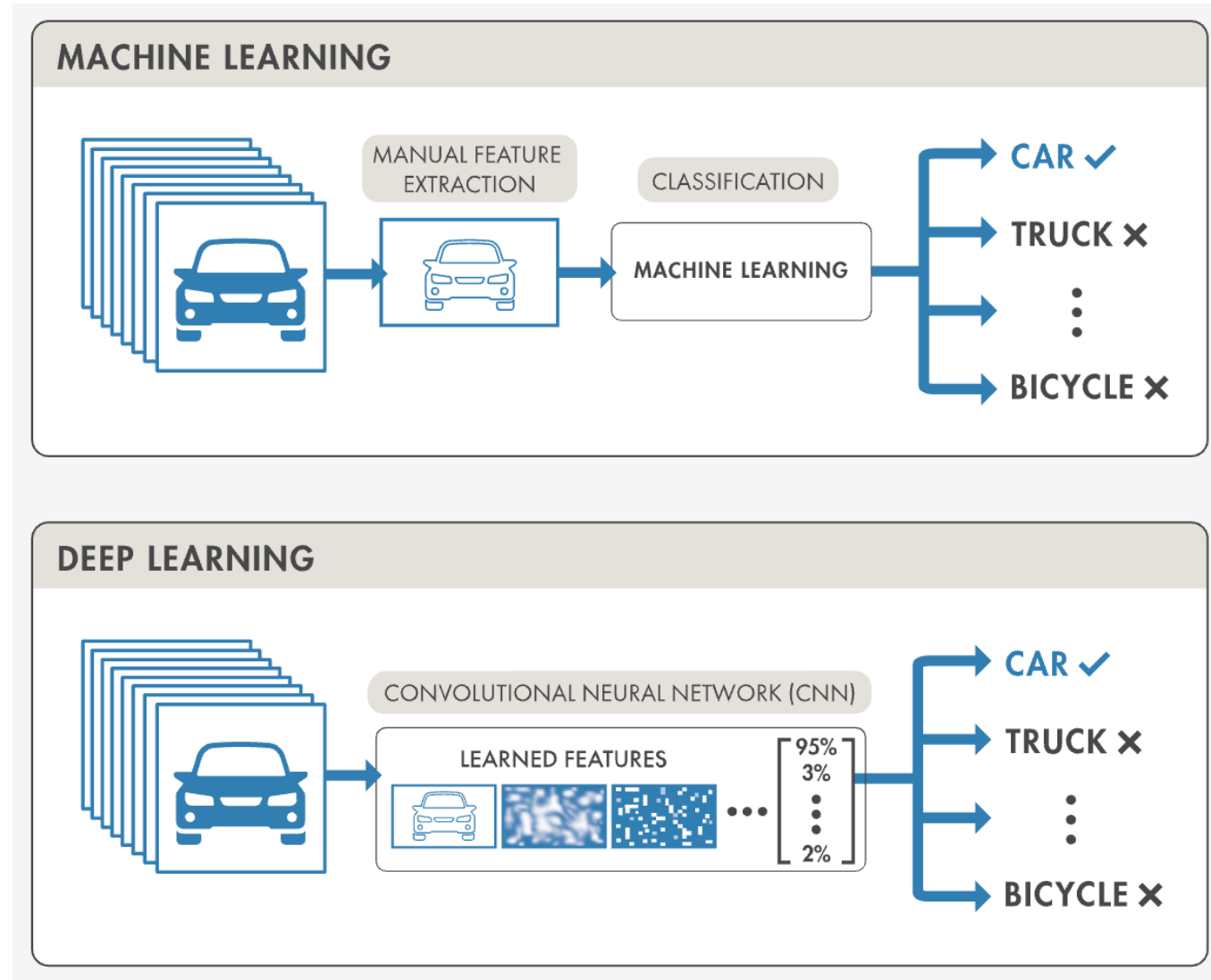


# Introducción. Reconocimiento de objetos

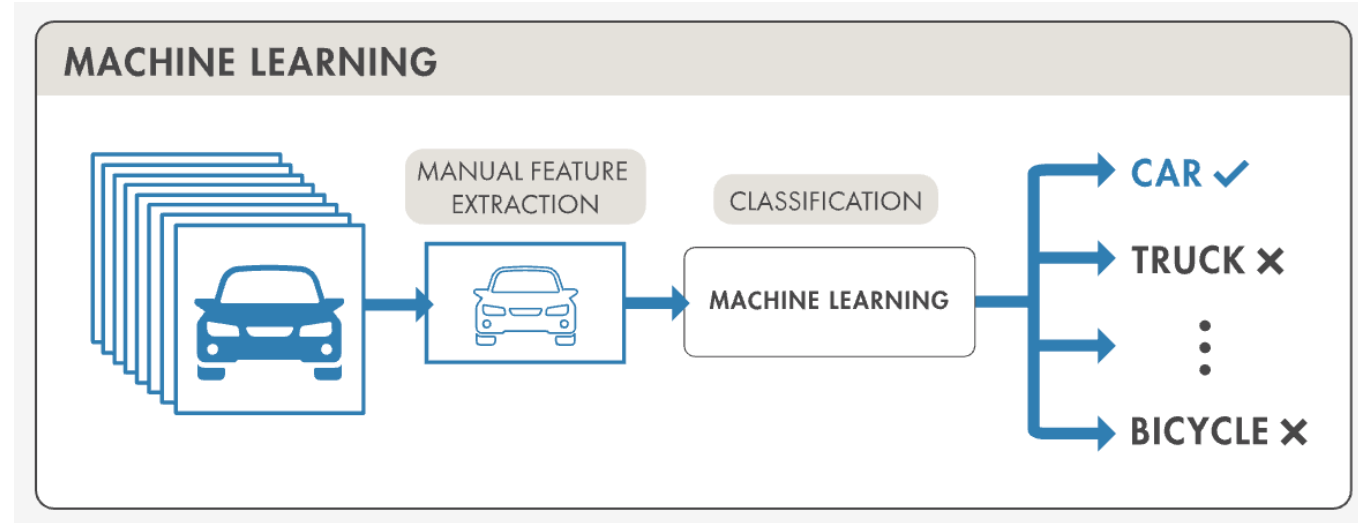
- Técnicas:
  - Coincidencia de plantillas
  - Segmentación de imágenes y análisis de blobs
  - Reconocimiento de características
  - Machine Learning / Deep Learning



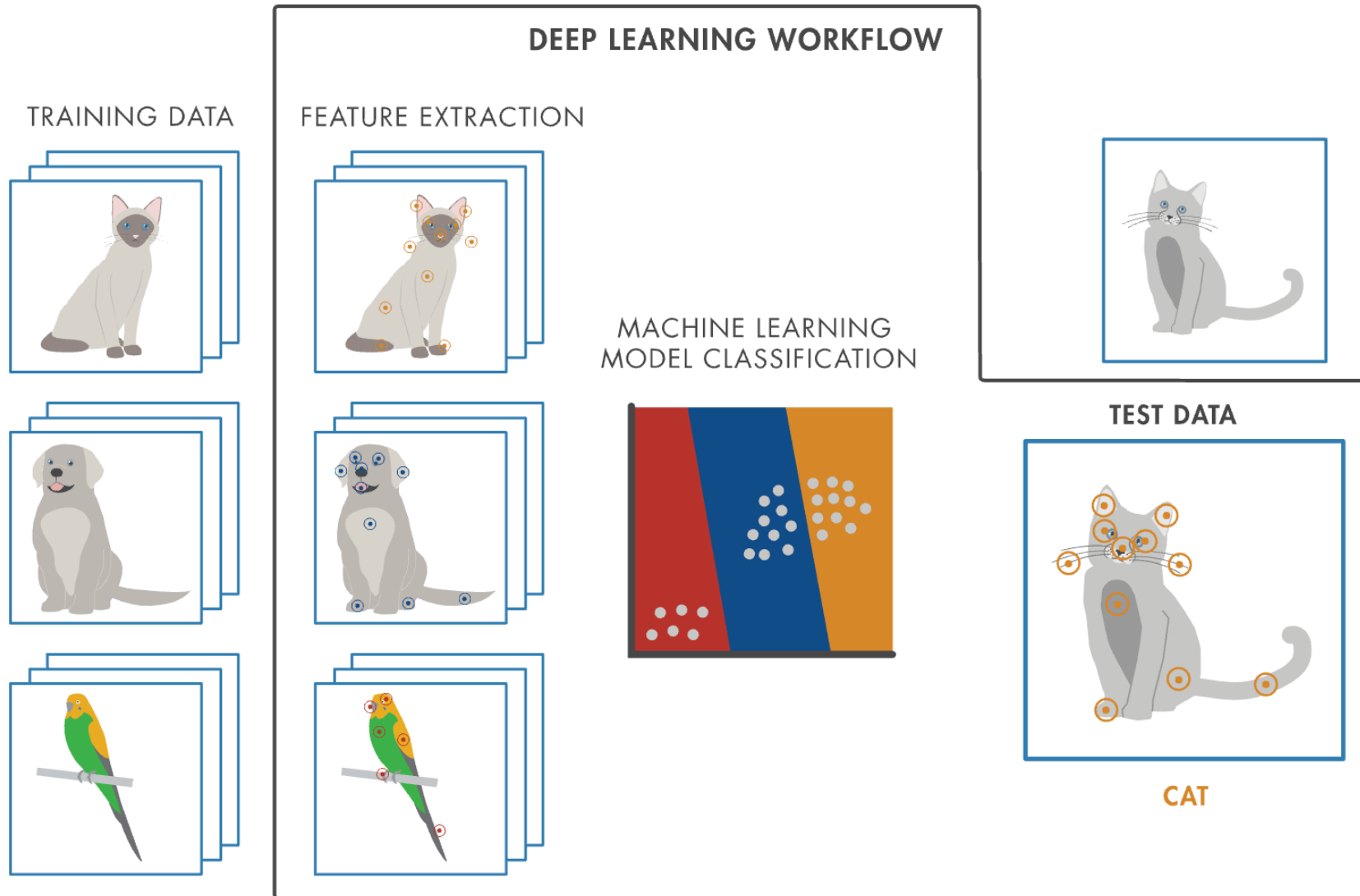
# Reconocimiento mediante técnicas ML/DL



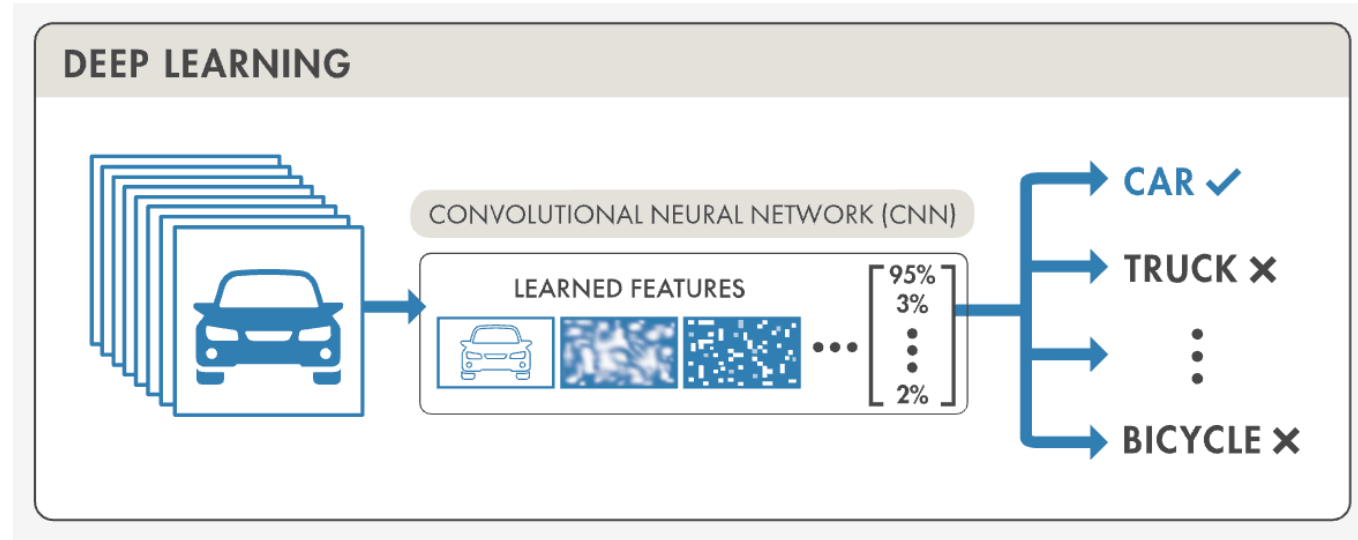
# Reconocimiento mediante técnicas ML/DL



# Reconocimiento mediante técnicas ML/DL

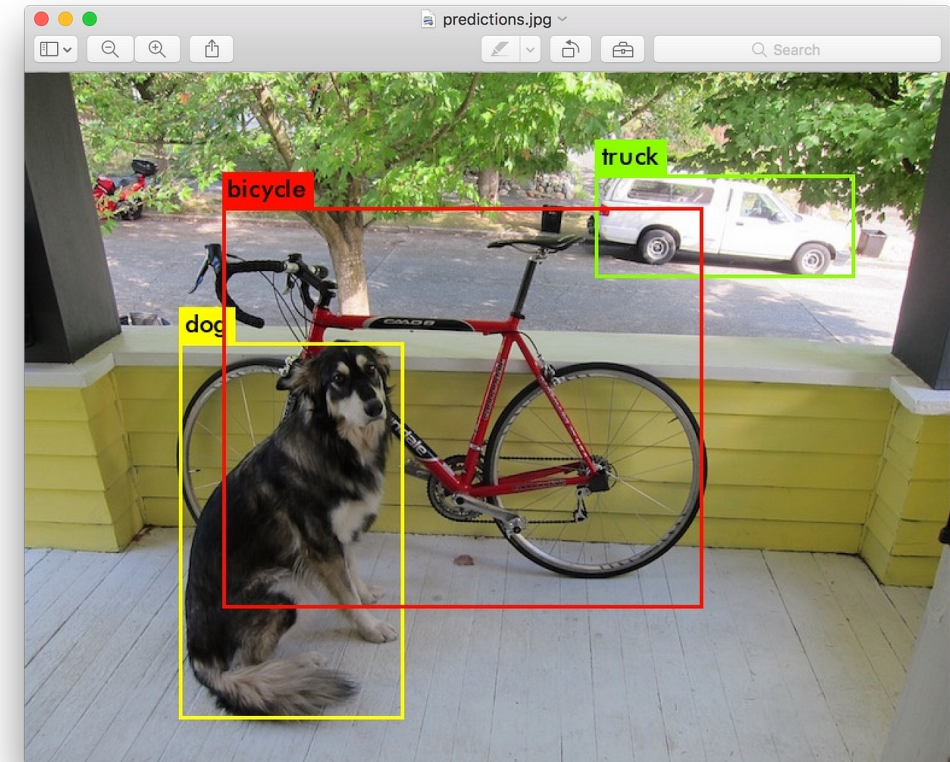
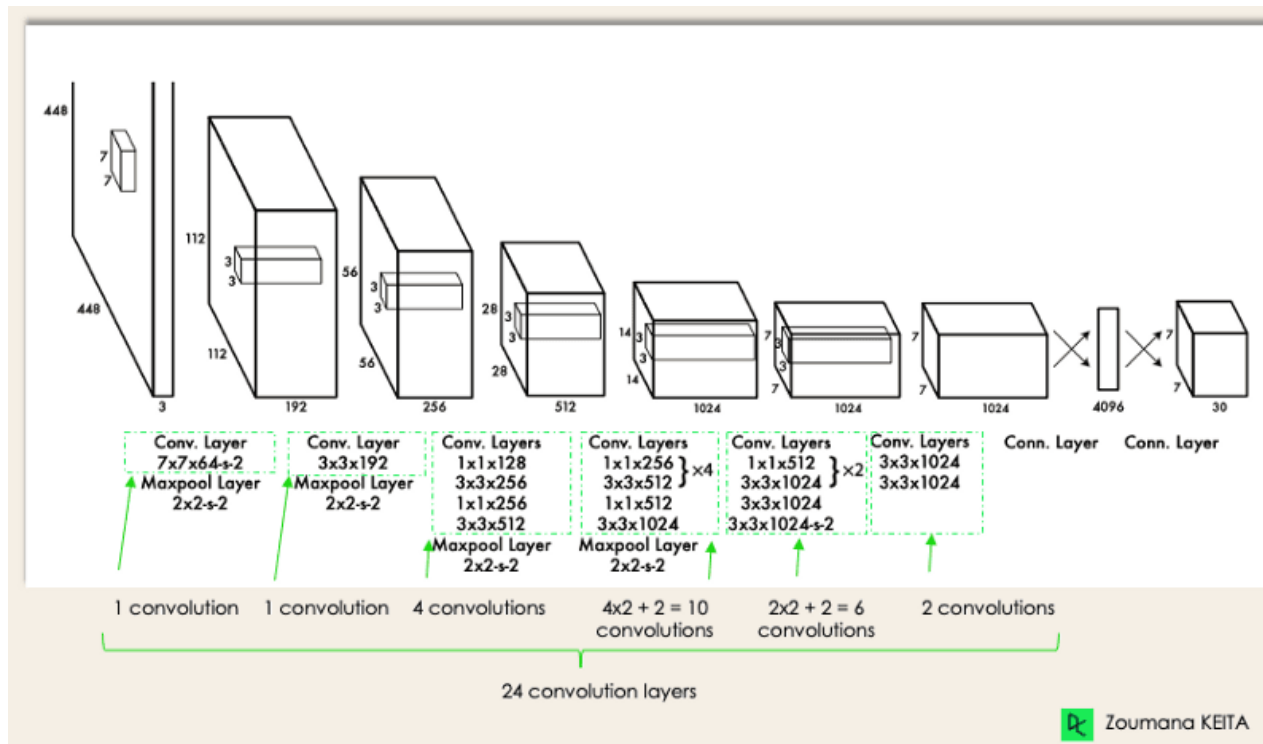


# Reconocimiento mediante técnicas ML/DL

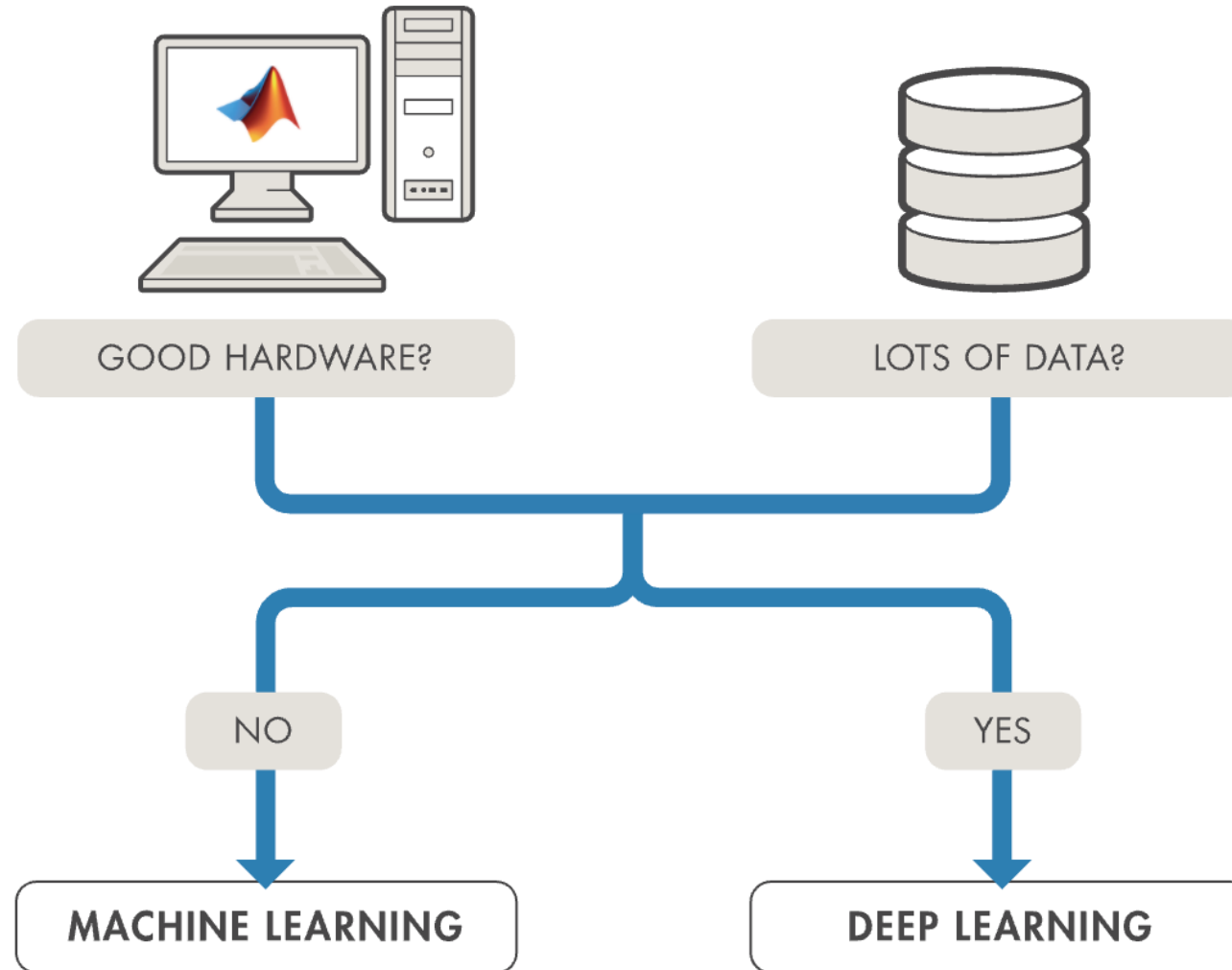


# Reconocimiento mediante técnicas ML/DL

- YOLO (You Only Look Once) 2015
  - CNN



# Machine Learning vs Deep Learning



# Curiosidad. Teachable Machine

- Permite crear modelos de aprendizaje automático.



# Índice

- Reconocimiento de objetos
  - Reconocimiento mediante características
  - Reconocimiento Machine Learning/Deep Learning
  - Reconocimiento de caras
- Reconocimiento de movimiento



# Reconocimiento de caras

- Algoritmo de Viola&Jones: Robust Real-time Object Detection. International Journal of Computer Vision. 2001
- El objetivo es detectar caras, no reconocerlas
- Otro objetivo es que sea muy rápido: este método tarda pocos milisegundos en procesar una imagen
- El método tiene pocos falsos positivos y un alto porcentaje de detección correcta
- Sólo sirve para caras frontales o con poco giro
- Le afecta el cambio de luminosidad

# Estructura general del método

- Extracción de características
  - Uso de imagen integral
  - Extracción de muchísimas características
- Selección de características
  - Algoritmo AdaBoost de entrenamiento
- Cascada de clasificadores
  - Conseguir más velocidad

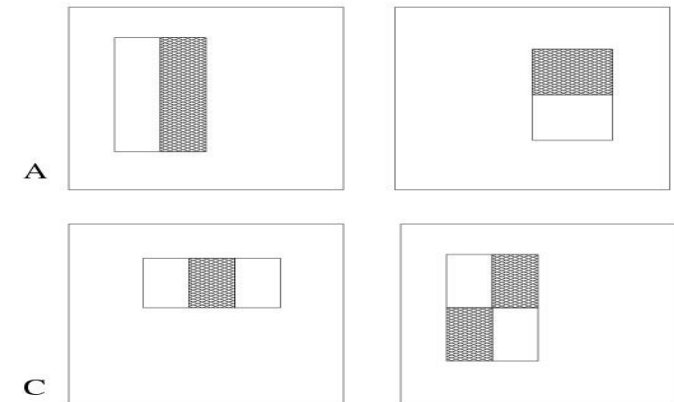
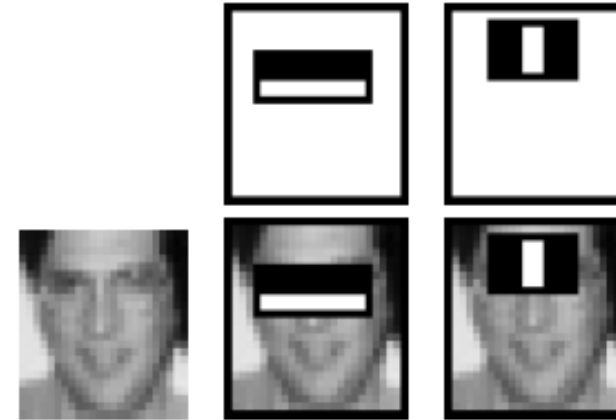
# Extracción de características

- Las caras comparten algunas características comunes:
  - La región del puente de la nariz es más clara que la de los ojos
  - La región de los pómulos es también más clara que los ojos
- Características a buscar:
  - Localizar nariz-ojos
  - Valores: claro-oscuro



# Características en rectángulo

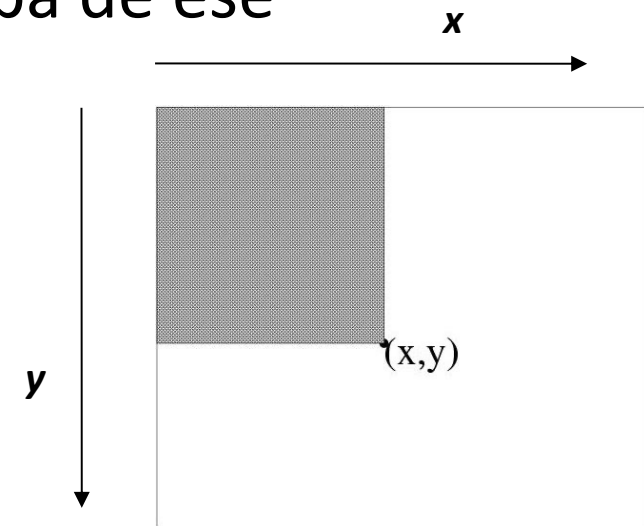
- Valor de la característica
  - $\frac{\sum(\text{píxeles en área negra})}{\sum(\text{píxeles en área blanca})}$
- Tres tipos: dos, tres, cuatro-rectángulos
- Cada característica está relacionada con una localización especial en la sub-ventana
- Cada característica puede tener cualquier tamaño y orientación



# Imagen Integral

- Dada una resolución de ventana de 24x24 existen 180000 posibles características!
- Por ello, es necesario un cálculo rápido
- Introducen el uso de la imagen integral: cada punto en la imagen integral es la suma de todos los píxeles de la imagen original a la izquierda y arriba de ese punto

$$\begin{aligned} \text{imagen original} &= I(x, y) \\ \text{imagen integral} &= II(x, y) = \sum_{x' \leq x, y' \leq y} I(x', y') \end{aligned}$$



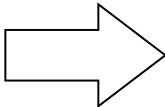
# Cálculo de la imagen Integral

- Se puede calcular en un solo paso:

$$s(x, y) = s(x, y-1) + i(x, y)$$

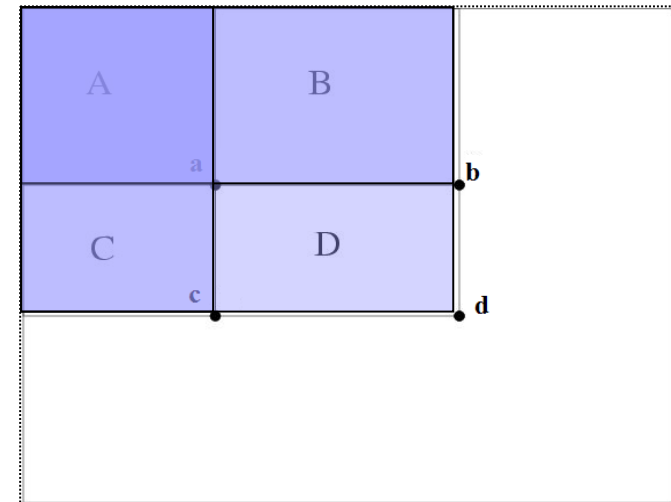
$$ii(x, y) = ii(x-1, y) + s(x, y)$$

donde  $s(x, y)$  es la suma acumulada de la columna. Hay que tener en cuenta las primeras fila y columna

| Imagen |   |   |   |   | Imagen integral |    |    |    |
|--------|---|---|---|---|-----------------|----|----|----|
| 0      | 1 | 1 | 1 |  | 0               | 1  | 2  | 3  |
| 1      | 2 | 2 | 3 |   | 1               | 4  | 7  | 11 |
| 1      | 2 | 1 | 1 |   | 2               | 7  | 11 | 16 |
| 1      | 3 | 1 | 0 |   | 3               | 11 | 16 | 21 |

# ¿Qué ventaja tiene la imagen integral?

- Como lo que queremos calcular es la suma de los píxeles dentro de un rectángulo, la imagen integral permite hacer este cálculo con tres operaciones básicas
- Para calcular características con dos, tres o cuatro rectángulos es necesario realizar 6, 8 y 9 referencias a la imagen, respectivamente (ejemplo con 2)



$$ii(a) = A$$

$$ii(b) = A+B$$

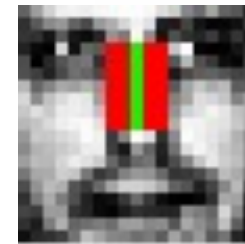
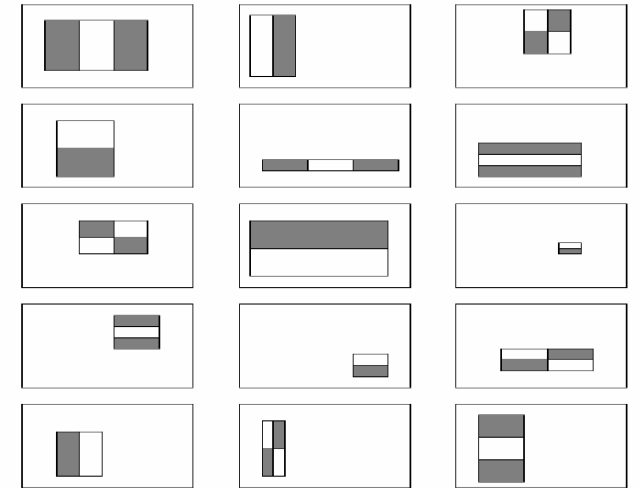
$$ii(c) = A+C$$

$$ii(d) = A+B+C+D$$

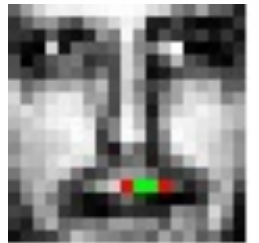
$$D = ii(d) + ii(a) - ii(b) - ii(c)$$

# Selección de características

- Demasiadas características: en una ventana 24x24 hay 180000 posibles combinaciones de tamaño y orientación
- Hay que seleccionar aquellas que sean relevantes para detectar una cara
- Para ello, podemos usar el algoritmo AdaBoost



Carac. relevante



Carac. irrelevante



# Índice

- Reconocimiento de objetos
  - Reconocimiento mediante características
  - Reconocimiento Machine Learning/Deep Learning
  - Reconocimiento de caras
- Reconocimiento de movimiento

# Reconocimiento de movimiento

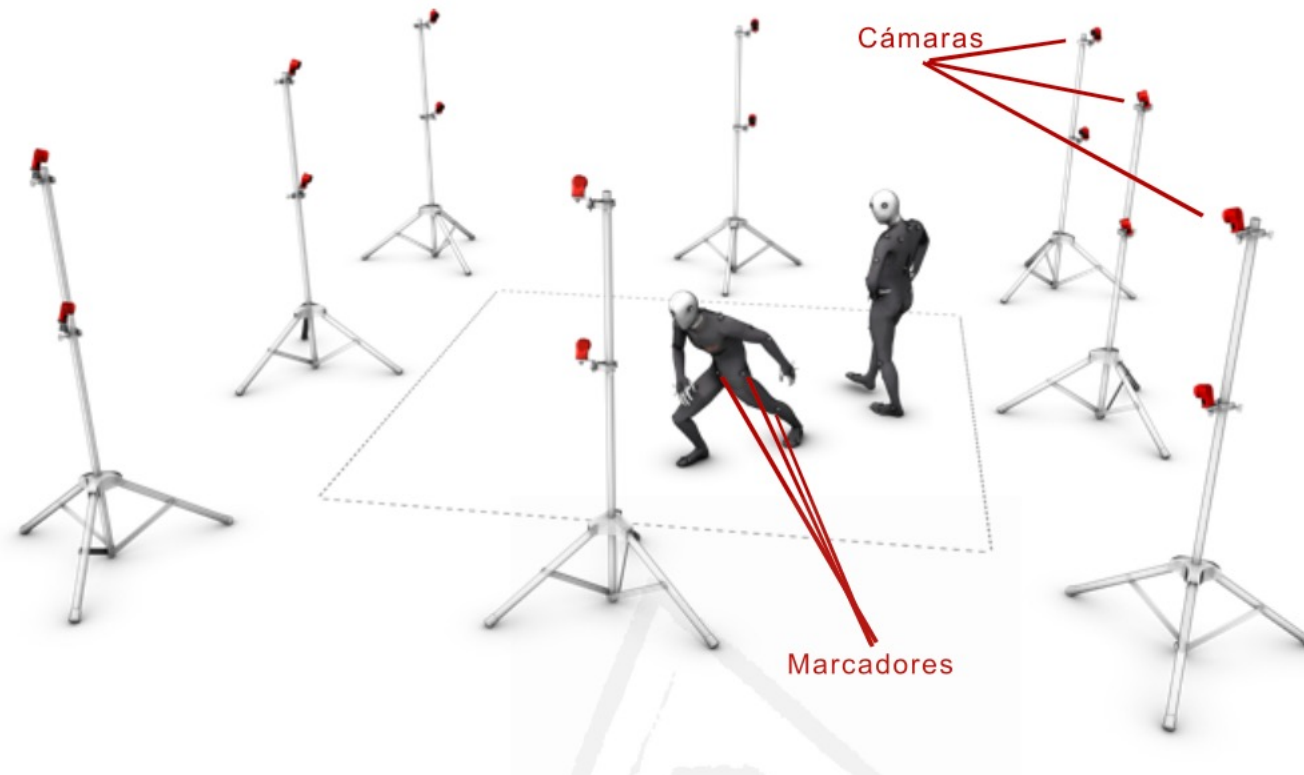
- Ejemplos de uso
  - Desarrollo de interfaces
  - Ámbito médico
  - Videojuegos y realidad virtual
- Estimar posición del esqueleto humano.
  - Captura del movimiento
  - Análisis del movimiento

# Reconocimiento de movimiento

- Tecnologías
  - Vídeo
  - Sistemas de captura de movimiento basados en sensores o marcadores
  - Sistemas capaces de generar mapas de profundidad o nubes de puntos (z-cam, kinect).

# Reconocimiento de movimiento

- Sistemas de captura de movimiento basados en sensores o marcadores



# Reconocimiento de movimiento

- Sistemas de captura de movimiento basados en sensores o marcadores



# Reconocimiento de movimiento

- Sistemas de captura de movimiento basados en sensores o marcadores



# Reconocimiento de movimiento

- Tecnologías
  - Vídeo
  - Sistemas de captura de movimiento basados en sensores o marcadores
  - Sistemas capaces de generar mapas de profundidad o nubes de puntos (z-cam, kinect).

# Reconocimiento de movimiento

- Sistemas capaces de generar mapas de profundidad o nubes de puntos (z-cam, kinect).



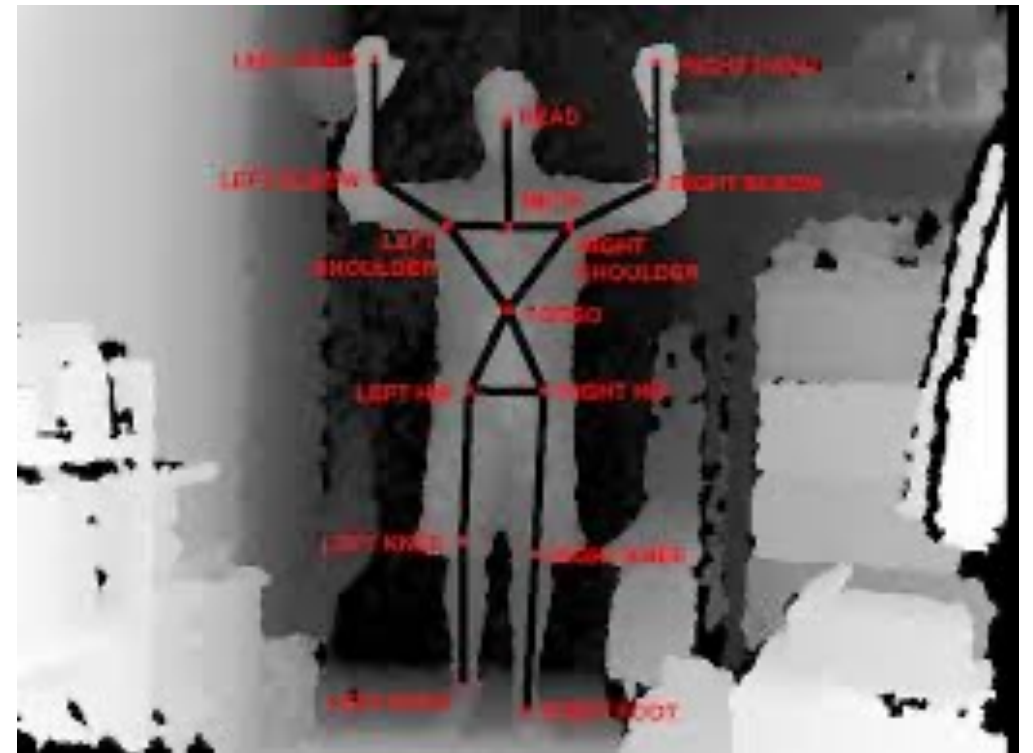


# Reconocimiento de movimiento

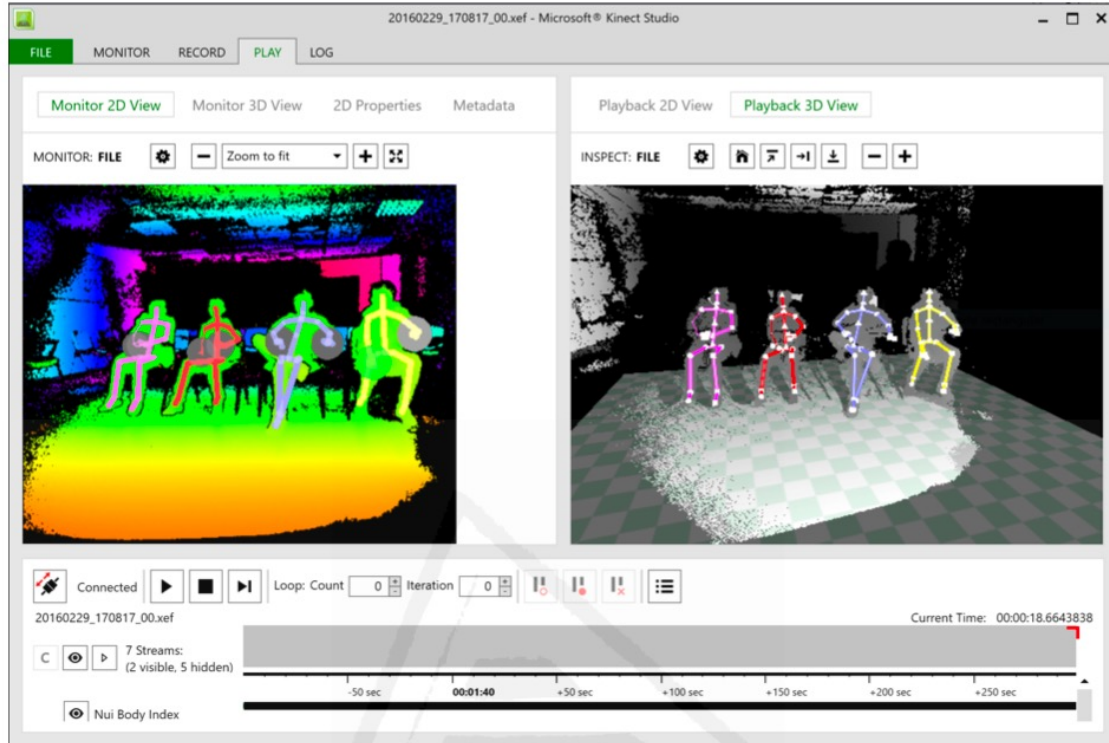
- Kinect
  - Xbox360
  - Cámara a color
  - Emisor de infrarrojos
  - Sensor de profundidad de infrarrojos
  - Motor de inclinación (v1)
  - Conjunto de micrófonos
- Uso no limitado a videojuegos
- Variedad de librerías

# Reconocimiento de movimiento

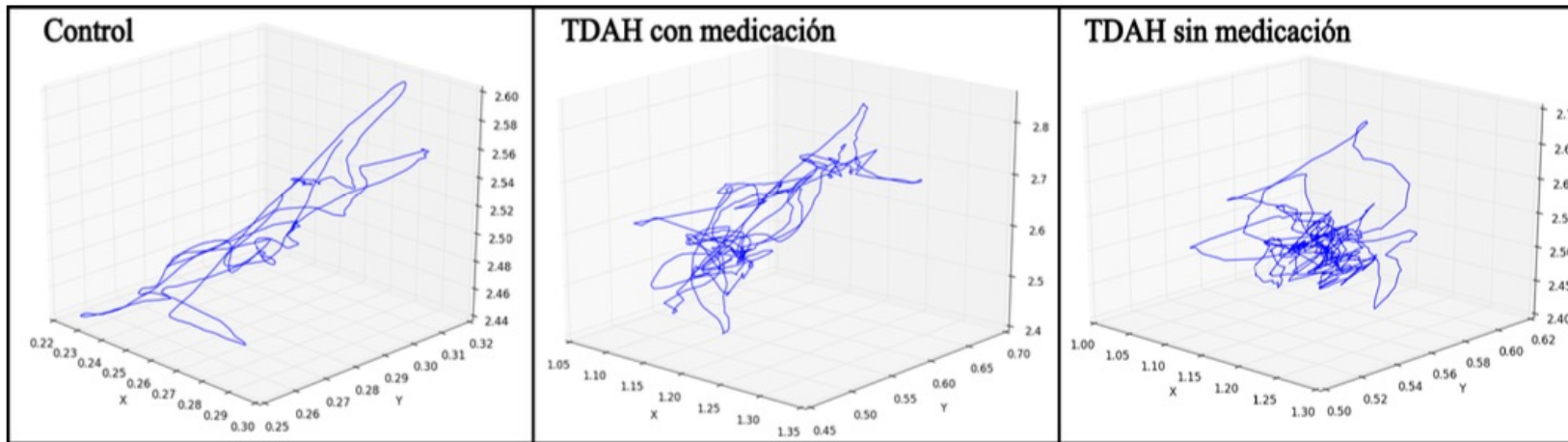
- Kinect
  - Herramientas de aprendizaje automático para detectar cuerpo humano.
- Analizar movimiento
- Detectar gestos
- Posturas



# Reconocimiento de movimiento



# Reconocimiento de movimiento

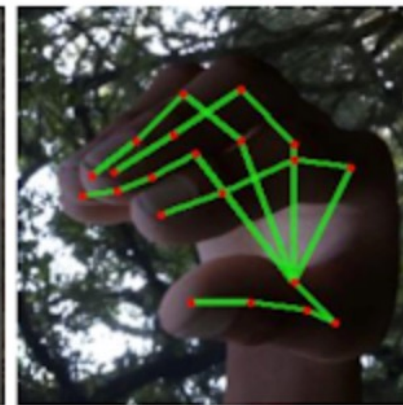
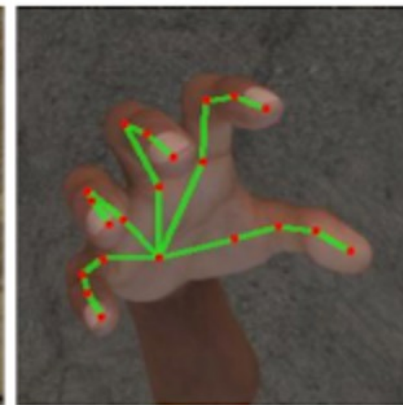
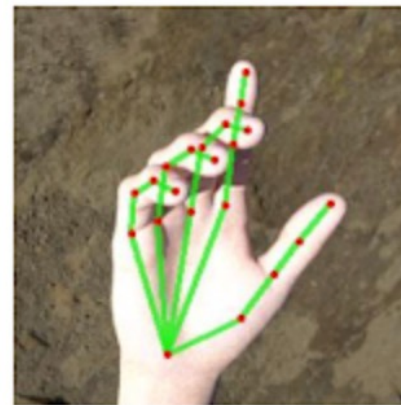
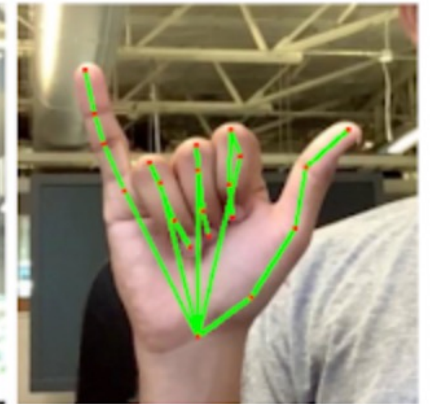
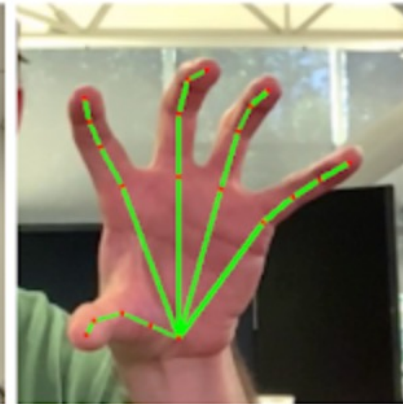
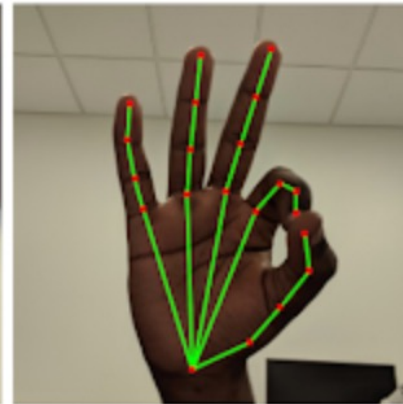
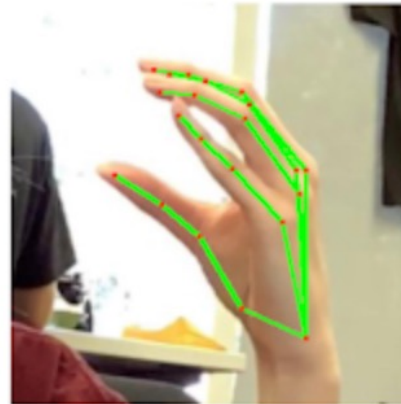
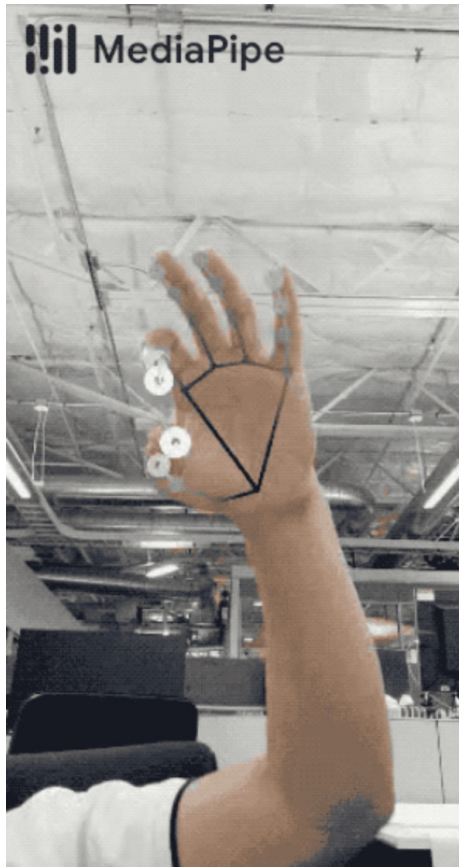


# Reconocimiento de movimiento

- Tecnologías
  - Vídeo
  - Sistemas de captura de movimiento basados en sensores o marcadores
  - Sistemas capaces de generar mapas de profundidad o nubes de puntos (z-cam, kinect).

# Reconocimiento de movimiento

- MediaPipe



# Referencias

- Método de Viola&Jones para reconocimiento de caras:  
[http://research.microsoft.com/~viola/Pubs/Detect/violaJones\\_IJCV.pdf](http://research.microsoft.com/~viola/Pubs/Detect/violaJones_IJCV.pdf)
- <http://www.pigeon.psy.tufts.edu/avc/kirkpatrick/default.htm>
- <https://es.mathworks.com/solutions/image-video-processing/object-recognition.html>