

# OpenCV

Open Source Computer Vision Library



Aliendo Marcos - Baranovsky Felipe - Rigonatto Eduardo Valverde

# ¿Qué es OpenCV?

Es una biblioteca de software de código abierto ampliamente utilizada para aplicaciones de visión por computadora y machine learning.

Fue desarrollada originalmente por Intel y es mantenida por una comunidad activa de desarrolladores.

Cuenta con más de 2500 algoritmos optimizados para análisis de imágenes y videos.





## Historia

Desarrollado inicialmente por Intel en 1999 como un proyecto de investigación. Finalmente liberada bajo licencia BSD en el 2000. Actualmente mantenida por Willow Garage e Itseez.

Desde entonces, se convirtió en una de las librerías de visión por computadoras más usadas en el mundo, con más de 18 millones de descargas a la fecha.



## Características

Algunas de sus tantas características incluyen el procesamiento de imágenes y videos, detección y reconocimiento de objetos, machine learning y deep learning.

Soportado por una gran cantidad de lenguajes de programación, incluyendo C++, python, java, javascript, etc.

Altamente optimizado para aprovechar procesadores multinúcleo y GPU.

# Visión por Computadora

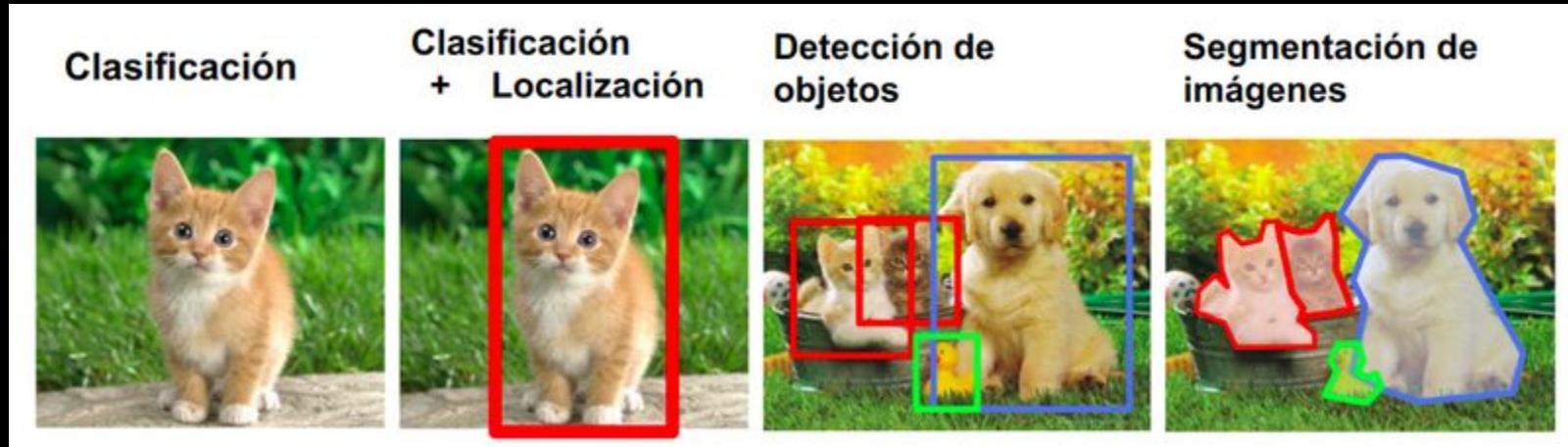
La visión por computadora es un campo de estudio que se enfoca en permitir que las computadoras interpreten, analicen y comprendan imágenes y videos digitales.

Implica el desarrollo de algoritmos y técnicas que permiten a las máquinas extraer información significativa de datos visuales.



# Tareas importantes

- Clasificación de imágenes.
- Localización de objetos.
- Detección de objetos.
- Segmentación de imágenes.



## Aplicación

Se puede utilizar para automatizar tareas que serían difíciles o imposibles de realizar para los humanos, como identificar objetos en una imagen o rastrear el movimiento de personas o vehículos.



## **Automatización**

Se utiliza para automatizar tareas que de otro modo requerirían intervención humana, como el control de calidad en la fabricación.

## **Eficiencia**

Puede analizar grandes cantidades de datos de manera rápida y precisa, lo que lleva a una mayor eficiencia en diversas industrias.

## **Seguridad**

Se puede utilizar para monitorear y detectar posibles riesgos de seguridad, como la detección de objetos en vehículos autónomos.

# Procesamiento de Imágenes

- Subconjunto de la visión por computadora.
- Mejorar/Corregir sus características.
- Aplicar algoritmos.



# Aplicaciones

Una librería con un universo de aplicaciones



# Aplicación



- Mejora de Calidad.



- Segmentación de imagen.



- Detección de bordes.



- Filtrado y transformaciones.



- Reconocimiento de patrones y clasificación.



- Reconstrucción 3D.



- Análisis de movimiento.



- Proyectar realidad.

# Principios de Imagen



## Carga y Visualización

Amplia variedad de soporte



## Acceso y Manipulación de Píxeles

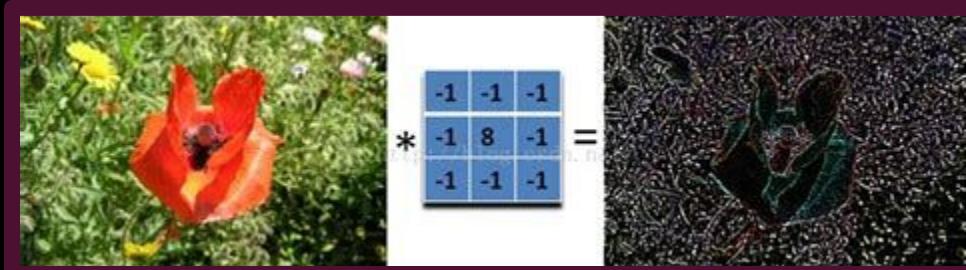
Imágenes representadas como matrices de píxeles.

Píxeles accesibles mediante indexación de matrices.

Píxeles manipulables mediante operaciones aritméticas básicas.

# Filtrado de Imágenes

- Técnica de transformación de valor.
- Mejorar/Modificar una imagen.
- Aplicar operaciones matemáticas a cada pixel.



- **Smoothing:** Difuminar para reducir el ruido. Incluye el desenfoque gaussiano y el desenfoque mediano.
- **Sharpening:** Intensificar los bordes para aumentar la nitidez. Incluye los algoritmos Laplaciano y Sobel.
- **Edge detection:** Identificar y resaltar los bordes. Incluye los algoritmos Canny y Scharr.



Original



Smoothing



Original



Edge Detection  
(Canny)



Original



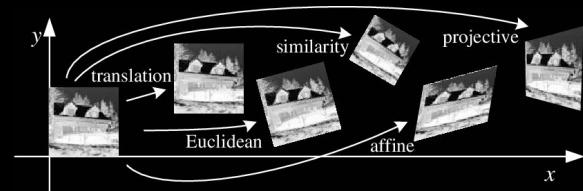
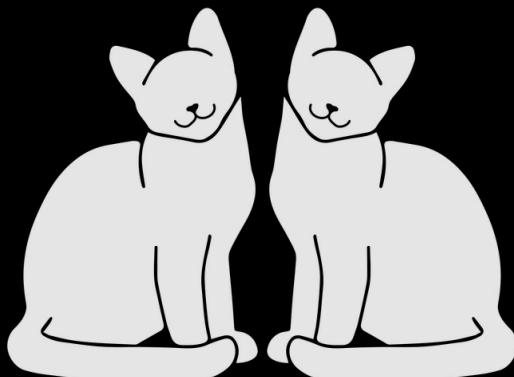
Smoothing



Edge Detection  
(Canny)

# Transformación de Imágenes

- Operaciones para modificar la apariencia de una imagen.
- Mejorar/Corregir una imagen.
- Aplicar operaciones matemáticas a cada pixel





- **Scaling:** Cambia el tamaño de una imagen. Se utiliza la función `resize`.



- **Rotation:** Gira una imagen en un ángulo determinado. Se utilizan las funciones `getRotationMatrix2D()` y `warpAffine()`.



- **Translation:** Desplaza una imagen en las direcciones x e y. Se utiliza la función `warpAffine()`.



- **Affine Transformation:** Conserva líneas paralelas y proporciones de distancia del mundo real. Se utilizan las funciones `getAffineTransform()` y `warpAffine()`.



- **Perspective Transformation:** Corrige la proyección de una imagen. Se utilizan las funciones `getPerspectiveTransform()` y `warpPerspective()`.

# Detección de Contornos y Formas

En visión por computadora, los contornos se definen como los límites de un objeto con intensidad similar. Los contornos se pueden utilizar para el análisis de formas y la detección de objetos.

- **Finding Contours:** Para encontrar contornos en una imagen, primero debemos convertirla a una imagen binaria.
- **Drawing Contours:** Después de encontrar los contornos, podemos dibujarlos en la imagen original usando la función drawContours() en OpenCV.
- **Shape Detection:** OpenCV proporciona varias funciones para la detección de formas, como approxPolyDP() y minEnclosingCircle().

# Detección y Descripción de Características

Técnicas que se utilizan para identificar puntos o regiones interesantes en una imagen y describirlos de una manera que puedan compararse fácilmente con otras imágenes.

Estas características se pueden describir utilizando varios métodos, como SIFT y SURF.

- **SIFT:** (Transformación de características invariantes de escala). Es un método para detectar y describir características locales en imágenes.
- **SURF:** (Características robustas aceleradas). Es un método para detectar y describir características locales en imágenes. Es similar a SIFT, pero es más rápido.

# Coincidencia de Características

La coincidencia de características es el proceso de encontrar correspondencias entre dos o más imágenes.

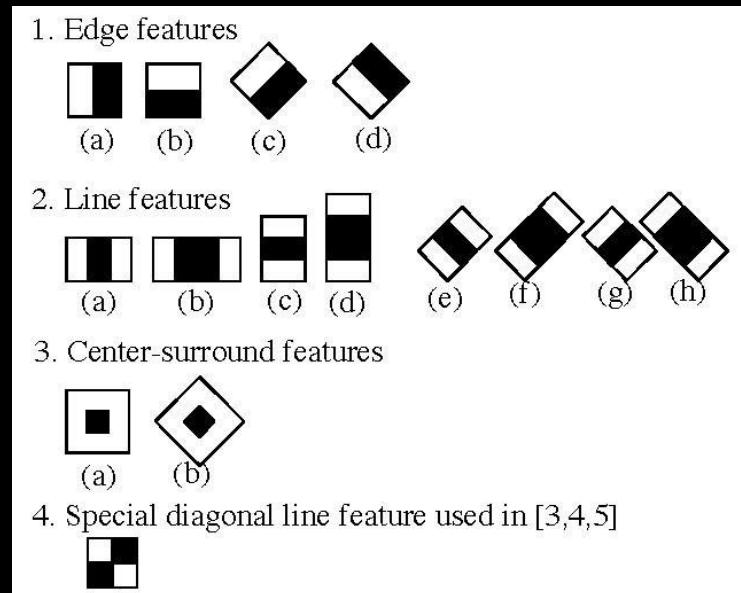
Es una tarea fundamental en visión por computadora y se utiliza en una variedad de aplicaciones como reconocimiento de objetos, unión de imágenes y reconstrucción 3D.

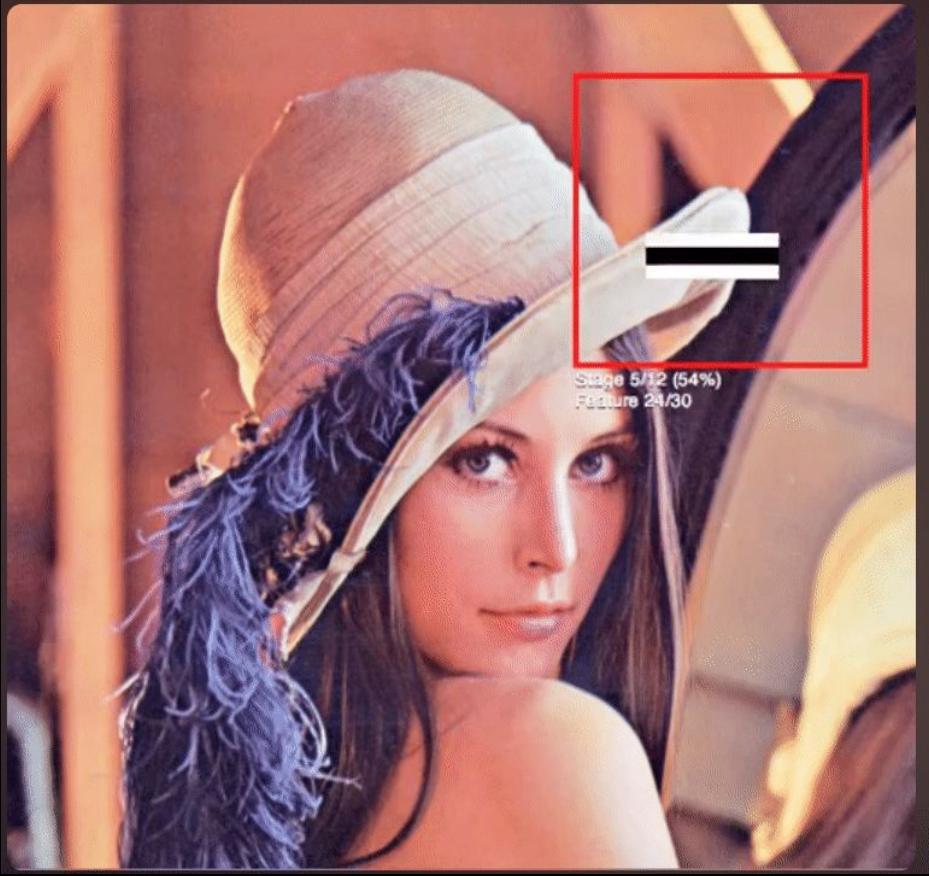


# Detección de Objetos

La detección de objetos es una técnica de visión por computadora que puede identificar y ubicar objetos dentro de una imagen o video. Es un componente clave de muchas aplicaciones, como vehículos autónomos, sistemas de seguridad y motores de búsqueda de imágenes.

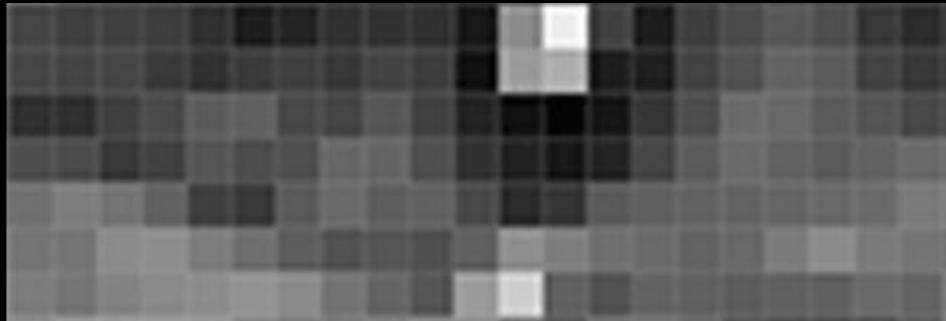
- **Haar Cascades:** Las cascadas de Haar son un método popular para la detección de objetos en OpenCV. Utilizan una serie de clasificadores entrenados en muestras de imágenes positivas y negativas para identificar objetos según sus características.





Stage 5/12 (54%)  
Feature 24/30

- **HOG:** El histograma de gradientes orientados (HOG) es otro método para la detección de objetos en OpenCV. Funciona calculando los gradientes de una imagen y luego creando un histograma de las orientaciones de los gradientes.



# Detección de Rostros



## ¿Qué es?

La detección de rostros es el proceso de detectar rostros en una imagen o frame y localizarlos. Es una tarea fundamental en visión por computadora y tiene muchas aplicaciones, como en sistemas de seguridad, redes sociales y entretenimiento.



## ¿Cómo lo realiza?

OpenCV proporciona varios métodos para la detección de rostros, incluidas cascadas de Haar, patrones binarios locales (LBP) y enfoques basados en aprendizaje profundo que utilizan redes neuronales convolucionales (CNN).

# Visión en Estéreo



## ¿Qué es?

Extraer información de profundidad de dos o más imágenes tomadas desde diferentes ángulos.



## ¿Cómo lo realiza?

Mediante funciones de calibración, rectificación y cálculo de mapas de disparidad. Se pueden utilizar para generar una nube de puntos 3D a partir de un par de imágenes estéreo.

# Flujo Óptico

Técnica para rastrear el movimiento de objetos analizando el movimiento de píxeles para determinar la dirección y la velocidad del movimiento.



## Object Tracking

Algoritmos MeanShift y CamShift.  
Rastrear objetos en videos o  
transmisiones de cámaras en vivo.

- Seguimiento de objetos en sistemas de vigilancia.
- Reconocimiento de gestos en la interacción persona-computadora.
- Odometría visual en robótica y drones.



## Camera Motion Tracking

Seguimiento del movimiento de cámara.  
Puede estimar la posición y orientación de  
la cámara en el espacio 3D.

# Machine Learning

OpenCV proporciona una variedad de algoritmos de aprendizaje automático que se pueden utilizar para clasificación de imágenes, detección de objetos y más.

## **Clasificación de Imágenes**

Los algoritmos de aprendizaje automático de OpenCV se pueden utilizar para clasificar imágenes en diferentes categorías, como identificar objetos o reconocer caras.

## **Detección de Objetos**

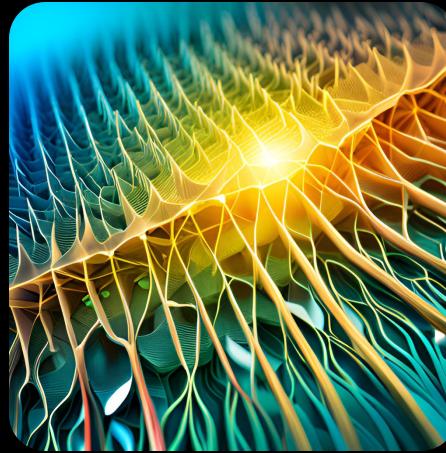
Los algoritmos de aprendizaje automático de OpenCV también se pueden utilizar para la detección de objetos, como la detección de vehículos o peatones en una transmisión de video.

# Deep Learning



## ¿Qué es?

El aprendizaje profundo es un subconjunto del aprendizaje automático que implica el entrenamiento de redes neuronales artificiales para realizar tareas como clasificación de imágenes, detección de objetos y reconocimiento de voz.



## Usos en OpenCV

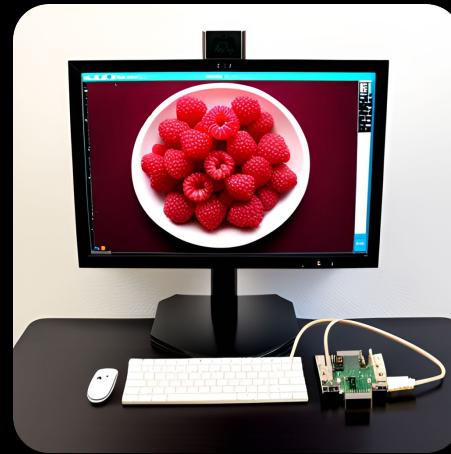
OpenCV proporciona un módulo de red neuronal profunda que se puede utilizar para cargar modelos previamente entrenados y realizar inferencias en imágenes o videos.

# OpenCV + Raspberry Pi



## Instalación

Un método popular es usar paquetes pre-buldeados disponibles en el sistema operativo Raspbian. Otra opción es compilar OpenCV desde la fuente,

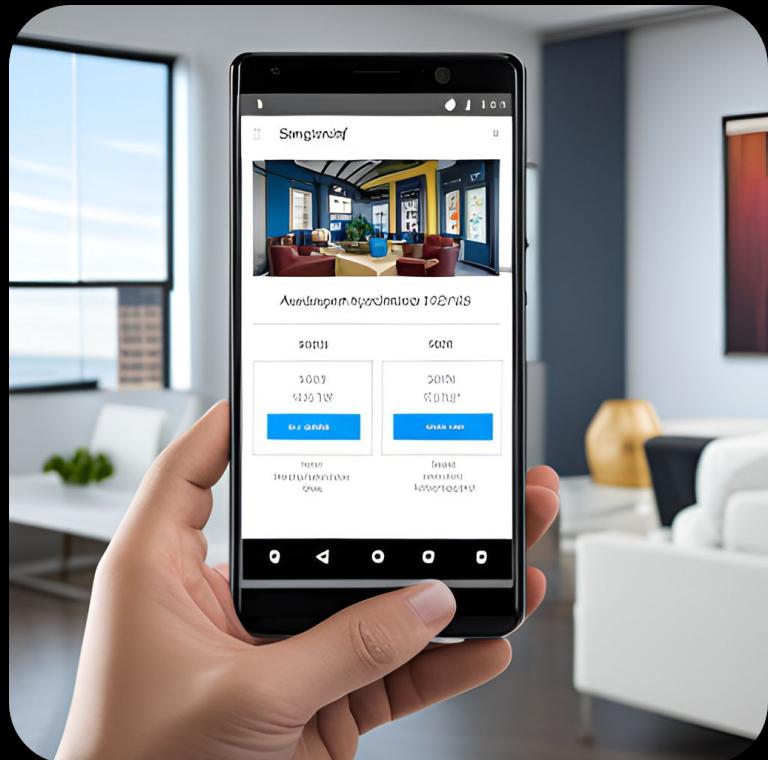


## Proyectos

Incluyen la creación de una cámara de seguridad que detecta movimiento y envía alertas, la construcción de un robot que puede navegar mediante visión por computadora y el desarrollo de un espejo inteligente que muestra información y reconoce rostros. ¡Las posibilidades son infinitas!

# OpenCV + Android

Para usar OpenCV en una aplicación de Android, es necesario agregar la biblioteca OpenCV como una dependencia.



# OpenCV + iOS

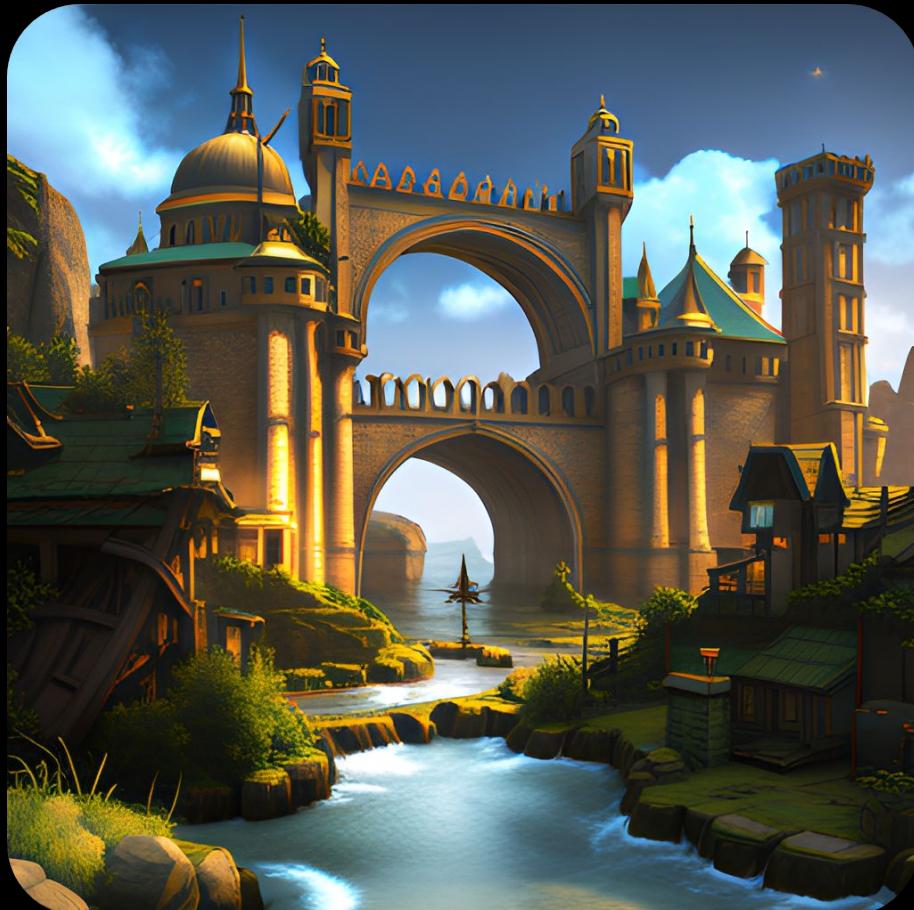
OpenCV está disponible como marco para el desarrollo de iOS, lo que permite una fácil integración de capacidades de visión por computadora en aplicaciones de iOS.



# OpenCV + Unity

OpenCV se puede integrar con Unity para agregar capacidades de visión por computadora a estos juegos y simulaciones.

Para utilizar OpenCV con Unity, deberá instalar el complemento OpenCV para Unity. Este complemento proporciona un conjunto de scripts y sombreadores de C# que le permiten utilizar funciones OpenCV dentro de Unity. Una vez instalado el complemento, puede importar su código OpenCV y usarlo en sus proyectos de Unity.



# OpenCV + AR



## ¿Qué es?

AR (Realidad Aumentada) es una tecnología que superpone objetos virtuales en el entorno del mundo real.  
Se utiliza en diversas aplicaciones, como juegos, educación y publicidad.

# OpenCV + VR

La tecnología de realidad virtual (VR) es una tecnología que crea entornos virtuales tridimensionales e interactivos que pueden ser explorados y experimentados por una persona a través de dispositivos como visores o gafas de VR.



# OpenCV + Robótica



Detección de Objetos



Navegación



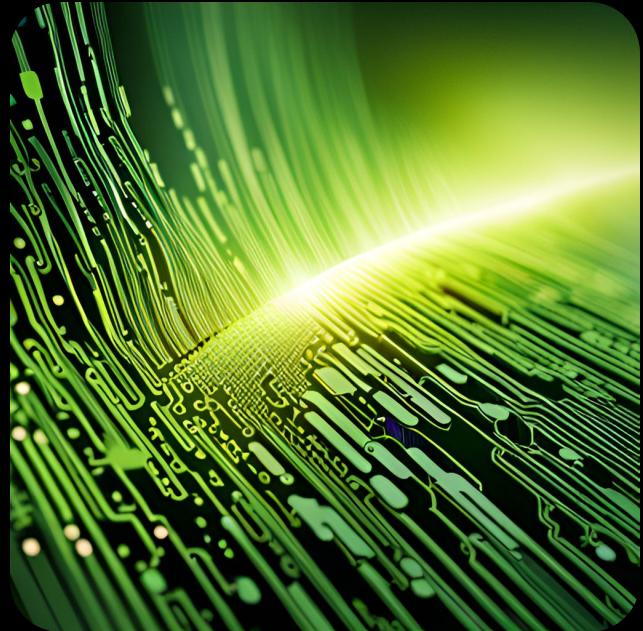
# OpenCV + Autos Automáticos

Permite que perciban su entorno y tomen decisiones basadas en esa percepción.

Incluye la detección de carriles, la detección de peatones, el reconocimiento de señales de tráfico y el seguimiento de objetos.



# Ejemplos



# ¡Gracias!

