**What is Computer Vision?**

Computer vision is a field of study within artificial intelligence (AI) that focuses on enabling computers to Intercept and extract information from images and videos, in a manner similar to human vision. It involves developing algorithms and techniques to extract meaningful information from visual inputs and make sense of the visual world.

**Computer Vision Examples:**

Here are some examples of computer vision:

Facial recognition: Identifying individuals through visual analysis.

Self-driving cars: Using computer vision to navigate and avoid obstacles.

Robotic automation: Enabling robots to perform tasks and make decisions based on visual input.

Medical anomaly detection: Detecting abnormalities in medical images for improved diagnosis.

Sports performance analysis: Tracking athlete movements to analyze and enhance performance.

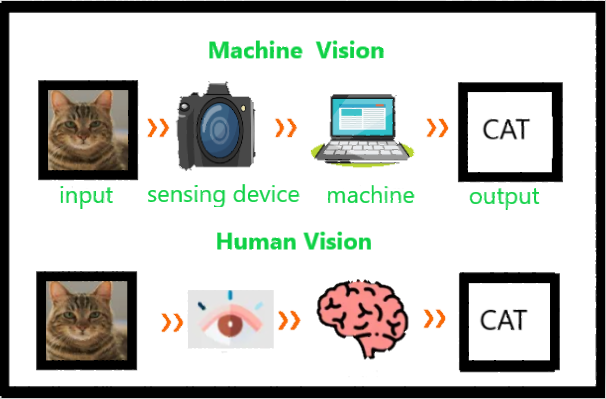
Manufacturing fault detection: Identifying defects in products during the manufacturing process.

Agricultural monitoring: Monitoring crop growth, livestock health, and weather conditions through visual data.

These are just a few examples of the many ways that computer vision is used today. As the technology continues to develop, we can expect to see even more applications for computer vision in the future.

**Overview to Computer Vision**

Computer vision means the extraction of information from images, text, videos, etc. Sometimes computer vision tries to mimic human vision. It’s a subset of computer-based intelligence or Artificial intelligence which collects information from digital images or videos and analyze them to define the attributes.



The entire process involves image acquiring, screening, analyzing, identifying, and extracting information. This extensive processing helps computers to understand any visual content and act on it accordingly. Computer vision projects translate digital visual content into precise descriptions to gather multi-dimensional data. This data is then turned into a computer-readable language to aid the decision-making process. The main objective of this branch of Artificial intelligence is to teach machines to collect information from images.

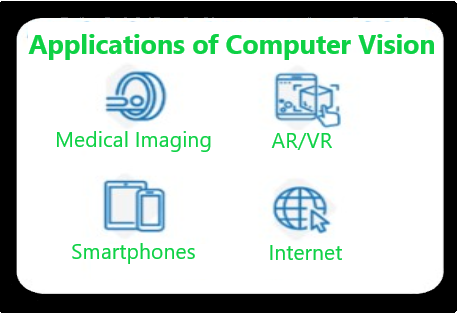
**Applications of Computer Vision**

Medical Imaging: Computer vision helps in MRI reconstruction, automatic pathology, diagnosis, and computer aided surgeries and more.

AR/VR: Object occlusion, outside-in tracking, and inside-out tracking for virtual and augmented reality.

Smartphones: All the photo filters (including animation filters on social media), QR code scanners, panorama construction, Computational photography, face detectors, image detectors like (Google Lens, Night Sight) that we use are computer vision applications.

Internet: Image search, Mapping, photo captioning, Ariel imaging for maps, video categorization and more.



**Detect an object with OpenCV-Python**

OpenCV is the huge open-source library for computer vision, machine learning, and image processing and now it plays a major role in real-time operation which is very important in today’s systems. By using it, one can process images and videos to identify objects, faces, or even the handwriting of a human. This article focuses on detecting objects.

**Object Detection**

Object Detection is a computer technology related to computer vision, image processing, and deep learning that deals with detecting instances of objects in images and videos. We will do object detection in this article using something known as haar cascades.

**Optical character recognition (OCR)** is a technology that recognizes text in images, such as scanned documents and photos. Perhaps you’ve taken a photo of a text just because you didn’t want to take notes or because taking a photo is faster than typing it. Fortunately, thanks to smartphones today, we can apply OCR so that we can copy the picture of text we took before without having to retype it.

**Segmentation**

Image segmentation is a computer vision technique that partitions a digital image into discrete groups of pixels—image segments—to inform object detection and related tasks. By parsing an image's complex visual data into specifically shaped segments, image segmentation enables faster, more advanced image processing.

**Image segmentation vs. object detection vs. image classification**

Image segmentation represents an advanced evolution of both image classification and object detection, as well as a distinct set of unique computer vision capabilities.

Image classification applies a class label to an entire image. For example, a simple image classification model might be trained to categorize vehicle images as “car” or “truck”. Conventional image classification systems are limited in sophistication, as they do not process individual image features separately.

Object detection combines image classification with object localization, generating rectangular regions, called “bounding boxes”, in which objects are located: rather than merely labeling a vehicle image as “car” or “truck”, an object detection model could indicate where in the image the car(s) or truck(s) can be found. While object detection can classify multiple elements within an image and approximate each element’s width and height, it cannot discern precise boundaries or shapes. This limits the ability of conventional object detection models to delineate closely bunched objects with overlapping bounding boxes.

Image segmentation processes visual data at the pixel level, using various techniques to annotate individual pixels as belonging to a specific class or instance. “Classic” image segmentation techniques determine annotations by analyzing inherent qualities of each pixel (called “heuristics”) like color and intensity, while deep learning models employ complex neural networks for sophisticated pattern recognition. The outputs of this annotation are segmentation masks, representing the specific pixel-by-pixel boundary and shape of each class—typically corresponding to different objects, features or regions—in the image.

**Common traditional (or "classic") image segmentation techniques include:**

**Thresholding:** Thresholding methods create binary images, classifying pixels based on whether their intensity is above or below a given “threshold value”. Otsu’s method is often used to determine the threshold value that minimizes intra-class variation.

**Histograms:** Histograms, which plot the frequency of certain pixel values in an image, are often used to define thresholds. For example, histograms can infer the values of background pixels, helping isolate object pixels.

**Edge detection:** Edge detection methods identify the boundaries of objects or classes by detecting discontinuities in brightness or contrast.

**Watersheds:** Watershed algorithms transform images into grayscale, then generate a topographical map in which each pixel’s “elevation” is determined by its brightness. Regions, boundaries and objects can be inferred from where “valleys”, “ridges” and “catchment basins” form.

**Region-based segmentation:** Starting with one or more “seed pixels”, region-growing algorithms group together neighboring pixels with similar characteristics. Algorithms can be agglomerative or divisive.

**Clustering-based segmentation**: An unsupervised learning method, clustering algorithms divide visual data into clusters of pixels with similar values. A common variant is K-means clustering, in which k is the number of clusters: pixel values are plotted as data points, and k random points are selected as center of a cluster (“centroid”). Each pixel is assigned to a cluster based on the nearest—that is, most similar—centroid. Centroids are then relocated to the mean of each cluster and the process is repeated, relocating centroids with each iteration until clusters have stabilized. The process is visualized here.