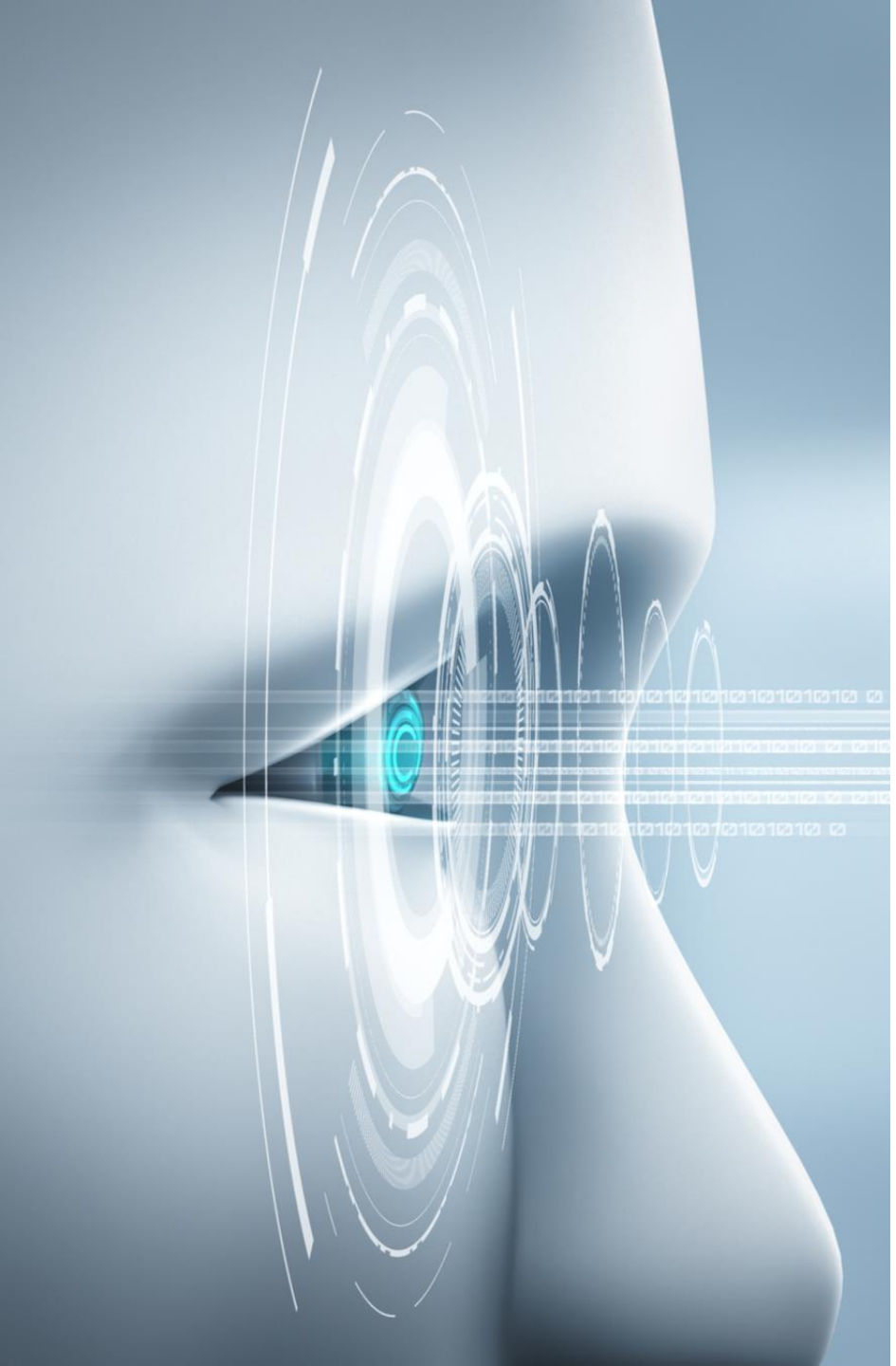


Computer Vision Applications

BY QUADEER SHAIKH

About me

Work Experience

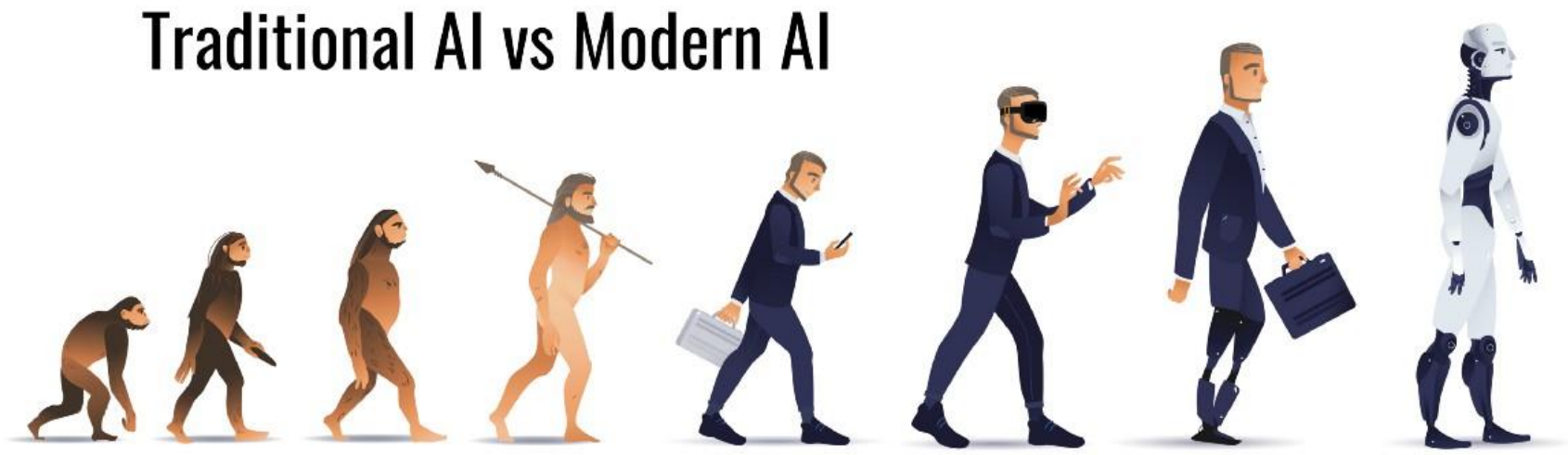
- Risk Analyst
  - Morgan Stanley (Jan 2023 – Present)
- Data Science Intern
  - AkzoNobel Coatings International B.V. Netherlands (Feb 2022 – Dec 2022)
- Data Science Intern
  - EzeRx Health Tech Pvt. Ltd. (Jan 2022 – July 2022)
- Associate Engineer
  - Tata Communications Ltd. (July 2019 – Aug 2020)
- Network Automation and Analysis Engineer Intern
  - Cisco (June 2018 – July 2018)

Education

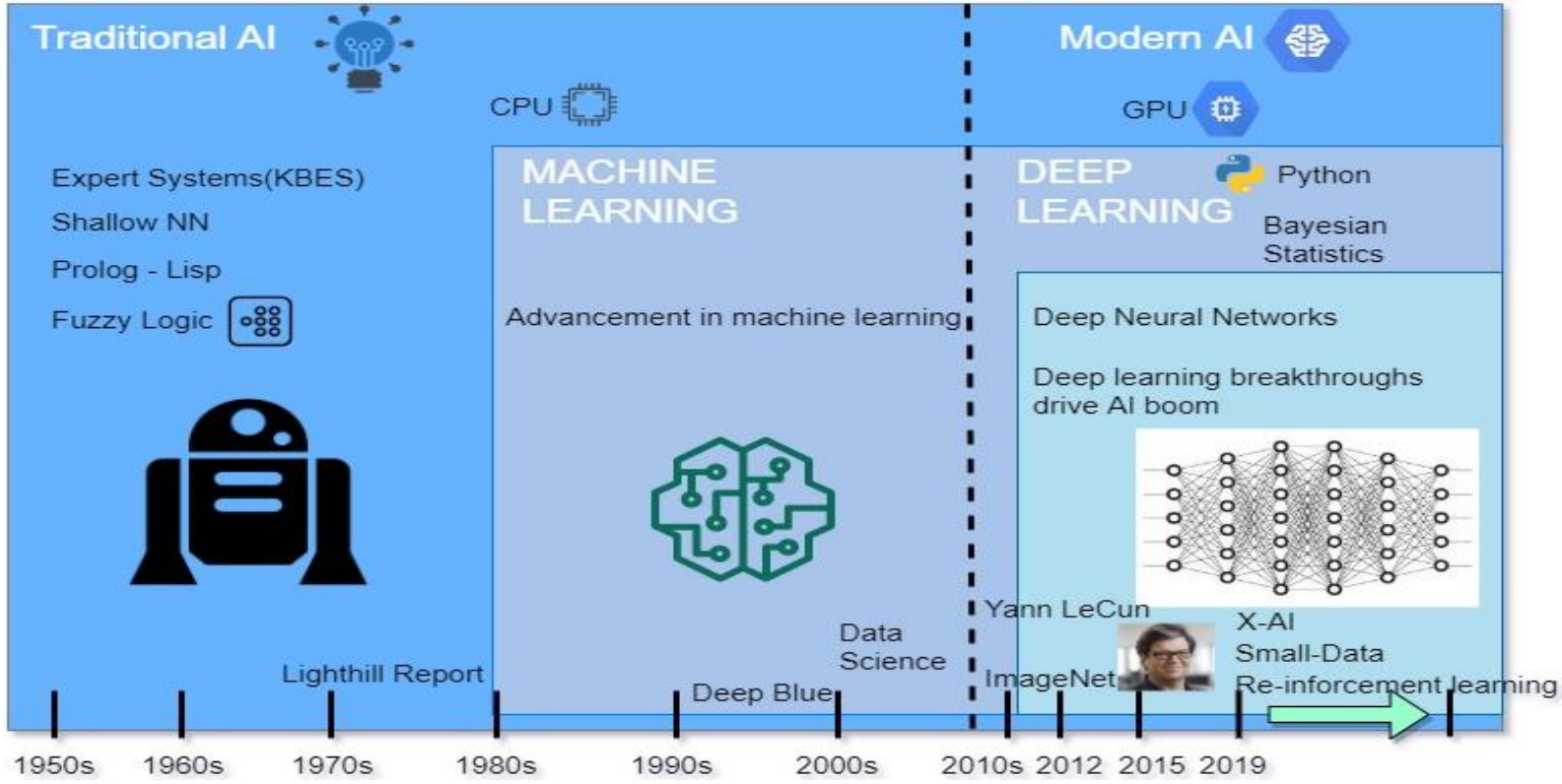
- M.Tech – Artificial Intelligence
  - NMIMS (2021 - 2023, currently pursuing)
- B.E. – Computer Engineering
  - Mumbai University (2015 - 2019)

Introduction

- Classical/Traditional AI
- Modern Day AI



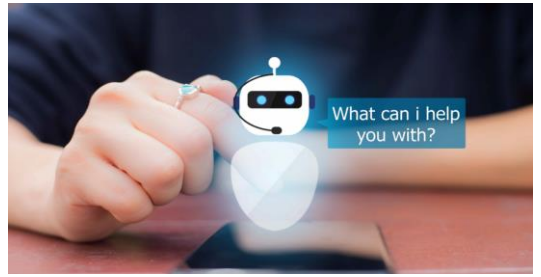
Traditional/Classical vs Modern AI

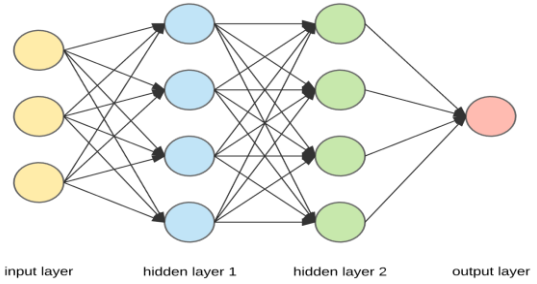


Source: Awais Bajwa, Traditional vs Modern AI, TowardsDataScience

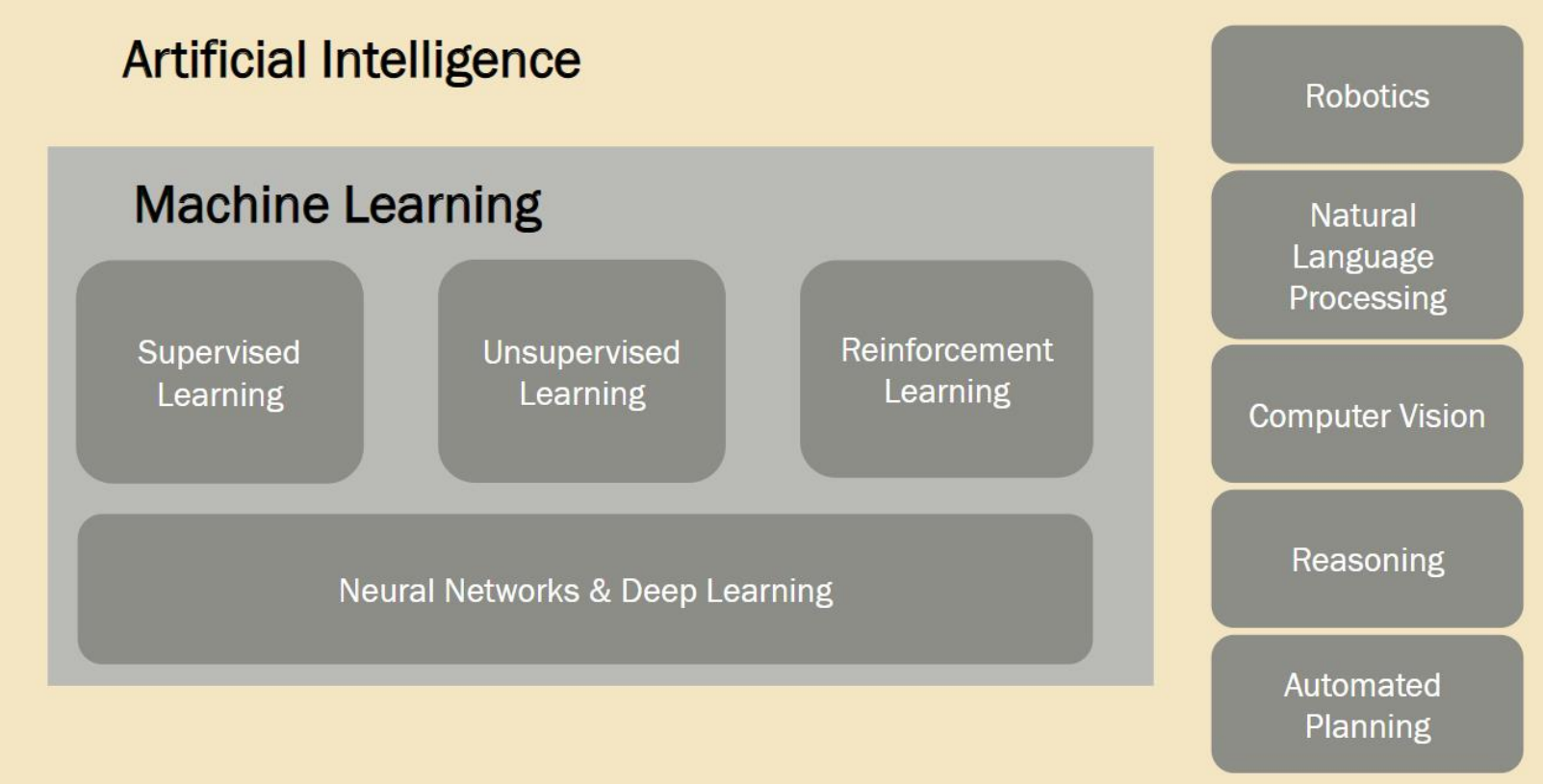
Capabilities of AI

- Natural Language Processing
- Knowledge Representation
- Automated Reasoning
- Machine Learning
- **Computer Vision**
- Robotics



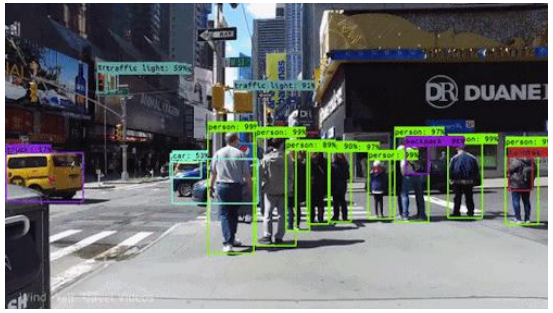


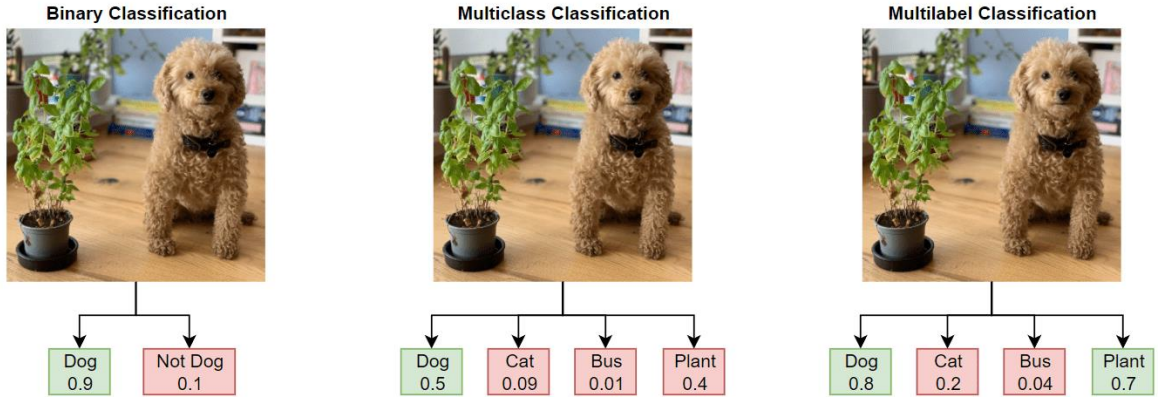
AI in a Nutshell



Source: Dr. Prasad Ramanathan

Computer Vision





Computer Vision

Traditional Computer Vision

1. Relies on basic image processing techniques
2. Handcrafted feature extraction/Mathematical feature extraction
3. Classic Machine Learning techniques e.g. Image classification using SVM, Naïve Bayes, Logistic Regression, etc. Image Segmentation using K-Means

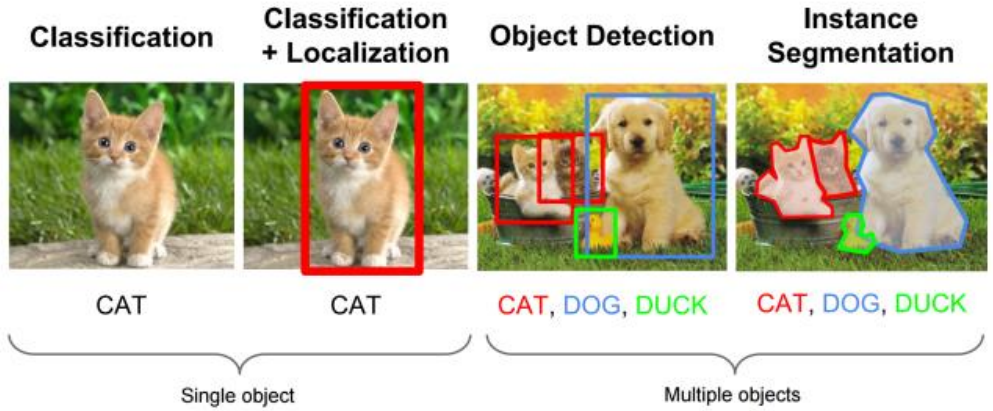
Deep Learning Based Computer Vision

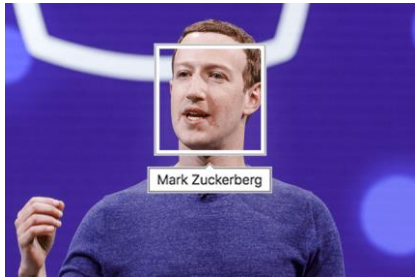
1. Data Driven approach (requires image data)
2. Feature extraction and pattern recognition is **learnt** during the training process
3. Training neural networks on a dataset of images for the desired task (classification, segmentation, object detection, etc)

Typical Tasks in Computer Vision

Recognition

1. Object Classification/Image Classification
2. Object Identification
3. Object Detection
4. Content Based Image Retrieval (Reverse Image Search)
5. Pose estimation
6. OCR – Optical Character Recognition
7. Face Recognition

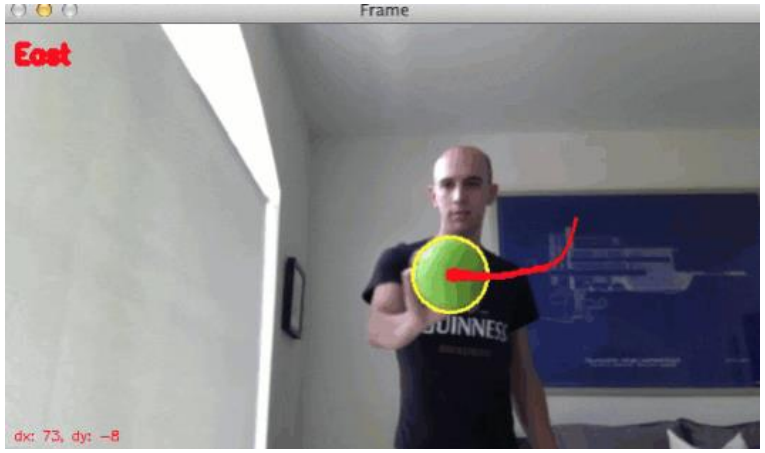




Typical Tasks in Computer Vision

Motion Analysis

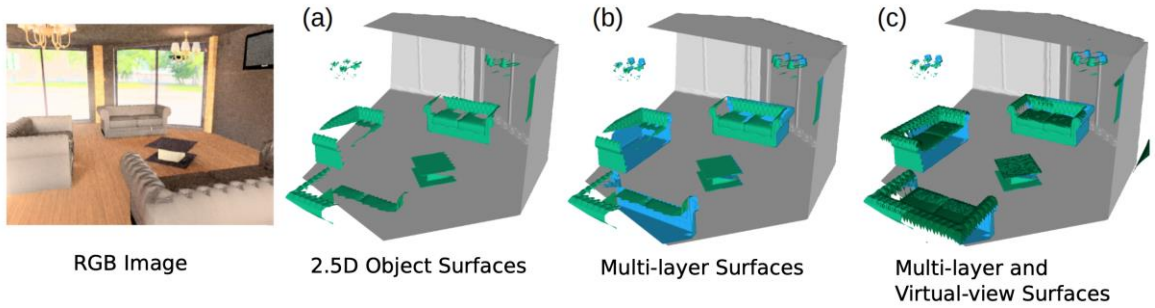
1. Tracking
2. Optical Flow

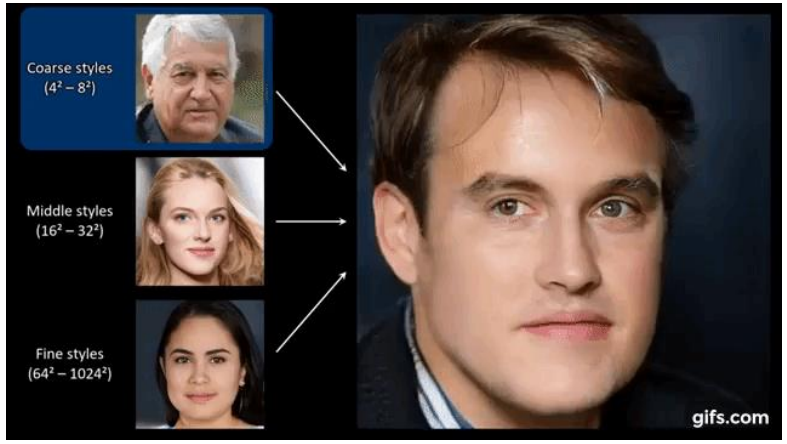


Typical Tasks in Computer Vision

Scene Reconstruction and Image Generation

1. Given one or (typically) more images of a scene, or a video, scene reconstruction aims at computing the 3D model of the scene
2. Generation of new synthetic instances of images that can pass for real images

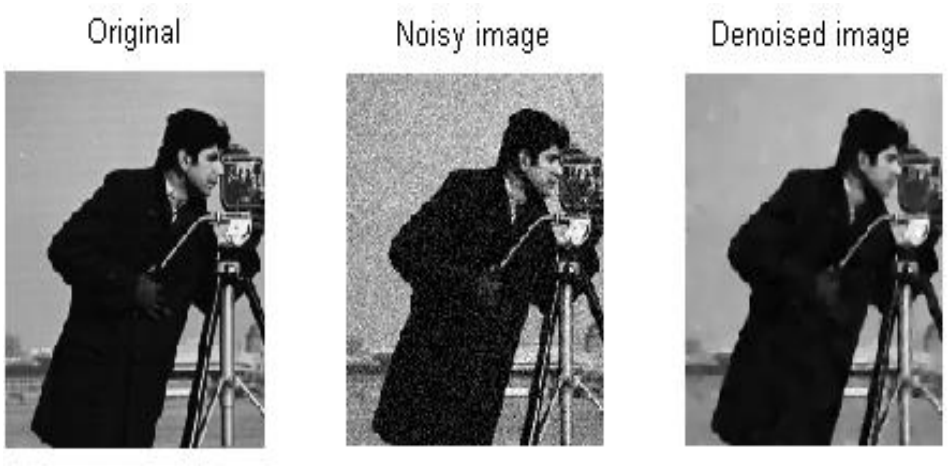


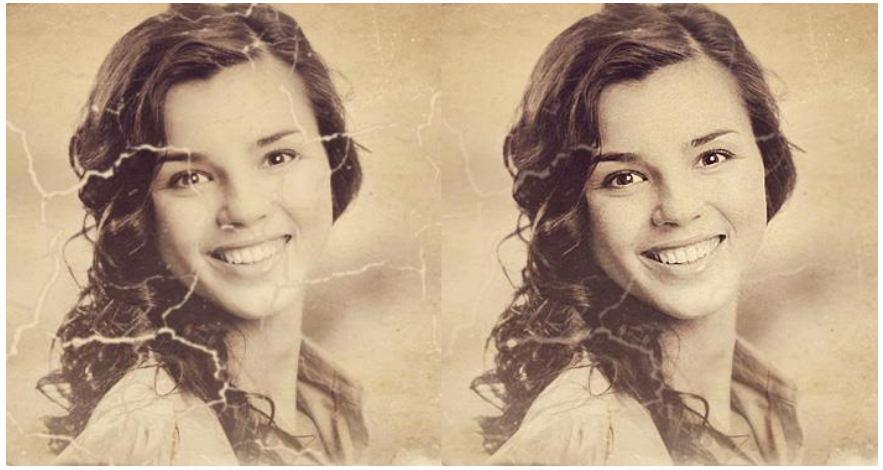


Typical Tasks in Computer Vision

Image Restoration

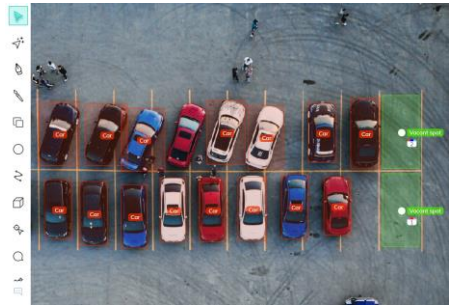
1. Aim is to remove noise from images
2. Restoring the erroneous parts of the images





Applications of Computer Vision

1. Transportation
   - Self Driving Cars, Pedestrian Detection, Traffic Flow Analysis
2. Healthcare
   - Xray Analysis, Cancer Detection, Movement Analysis, CT and MRI
3. Manufacturing
   - Reading Text and Barcodes, Defect Inspection
4. Agriculture
   - Aerial survey and imaging, livestock health monitoring, plant disease detection



For more applications visit this link: https://www.v7labs.com/blog/computer-vision-applications

Computer Vision Pipeline/System

A pipeline is **a series of processes that migrate data from a source to a destination.**

1. Image acquisition/collection
   - Cameras, radar, ultra sonic cameras, etc
2. Preprocessing
   - Resampling to assure the image coordinate system, noise reduction, contrast enhancement, augmentation
3. Feature Extraction
   - Lines, edges, ridges, corners, blobs, etc
4. Detection/Segmentation/Different Operation
5. Decision Making
   - E.g. generating report of a medical diagnosis, automated vehicle changing its direction, etc.

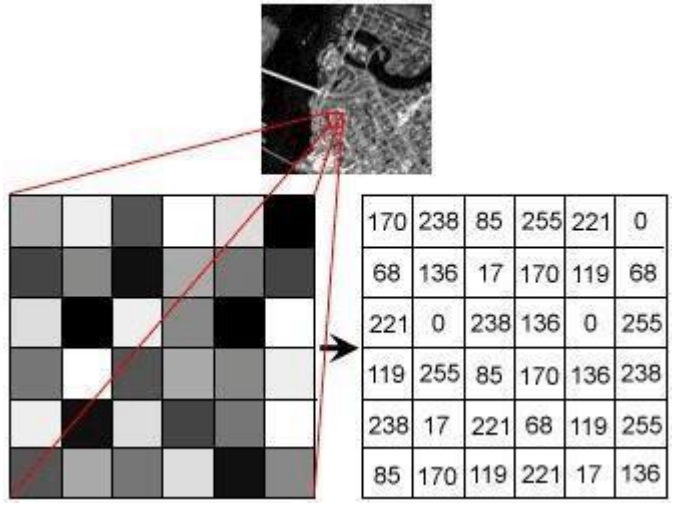
All about Images

1. Digital Image as a 2D function
2. Image Representation
3. Color Spaces in Image

Digital Image as a 2D Function

An image may be defined as a two-dimensional function, f(x, y), where x and y are spatial (plane) coordinates, and the amplitude of f at any pair of coordinates (x, y) is called the intensity or gray level of the image at that point.

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & f(0,2) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & f(1,2) & \dots & f(1,N-1) \\ \vdots & \vdots & \vdots & & \vdots \\ f(M-1,0) & f(M-1,1) & f(M-1,2) & \dots & f(M-1,N-1) \end{bmatrix}$$



170	238	85	255	221	0
68	136	17	170	119	68
221	0	238	136	0	255
119	255	85	170	136	238
238	17	221	68	119	255
85	170	119	221	17	136

Image Representation

In an 8-bit image **each pixel occupies exactly one byte**. This means each pixel has 256 possible numerical values, from 0 to 255. Therefore, the color palette for an 8-bit image normally contains 256 entries, defining color 0 through color 255.

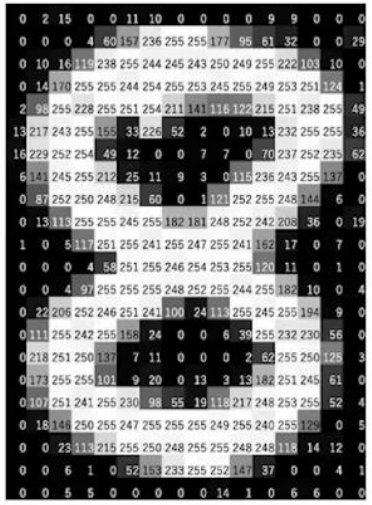
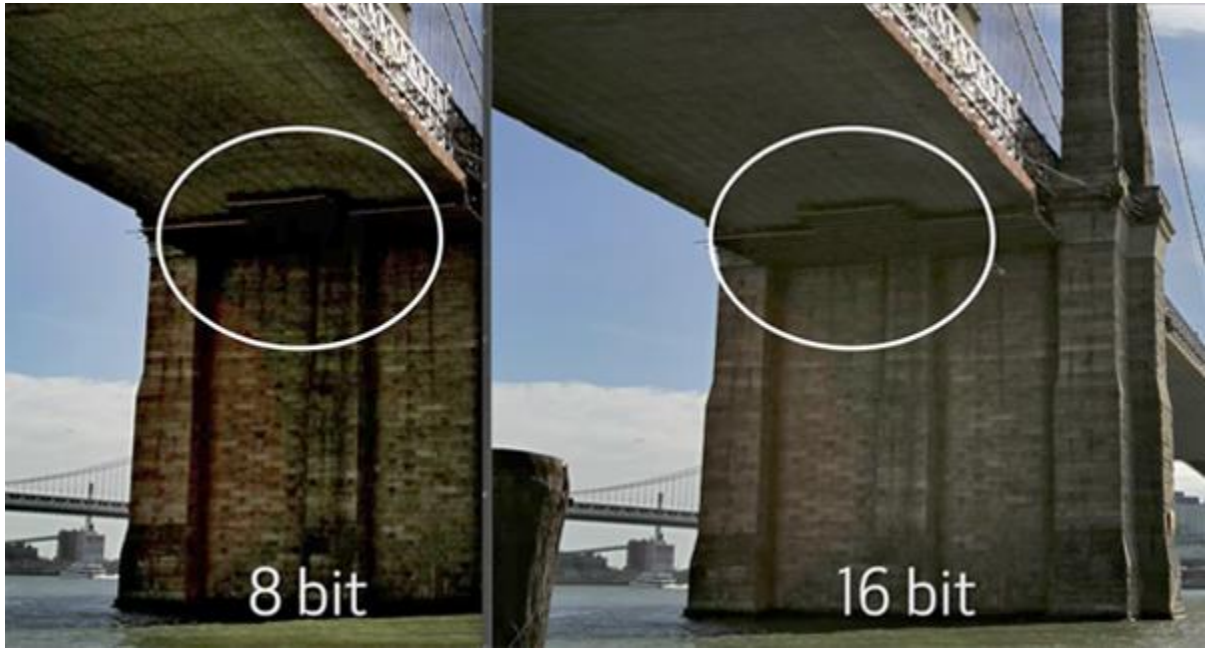


Image Representation



Images with higher bit representations have more range to display the pixel intensity. Therefore, they also tend to be larger in file size.

Color Space

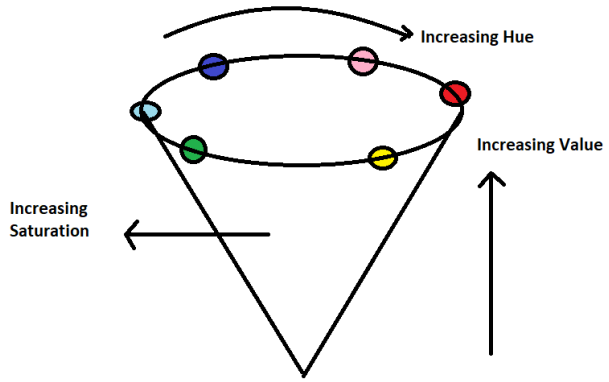
Commonly used color spaces

1. Grayscale – 1 channel, Dimensions: (height x width)
2. RGB – 3 channels, Dimensions: (height x width x channels)
3. HSV – 3 channels, Dimensions: (height x width x channels)

HSV color space: It stores color information in a cylindrical representation of RGB color points. It attempts to depict the colors as perceived by the human eye. Hue value varies from 0-179, Saturation value varies from 0-255 and Value (brightness value) varies from 0-255. It is mostly used for color segmentation purpose

HSV Contd.

- Hue is determined by the dominant wavelength of the visible spectrum. It is the attribute that permits colors to be classified as red, yellow, green, blue, or an intermediate color.
- Saturation pertains the amount of white light mixed with a hue. High saturation colors contain little or no white light
- Value (Brightness Value) refers to intensity



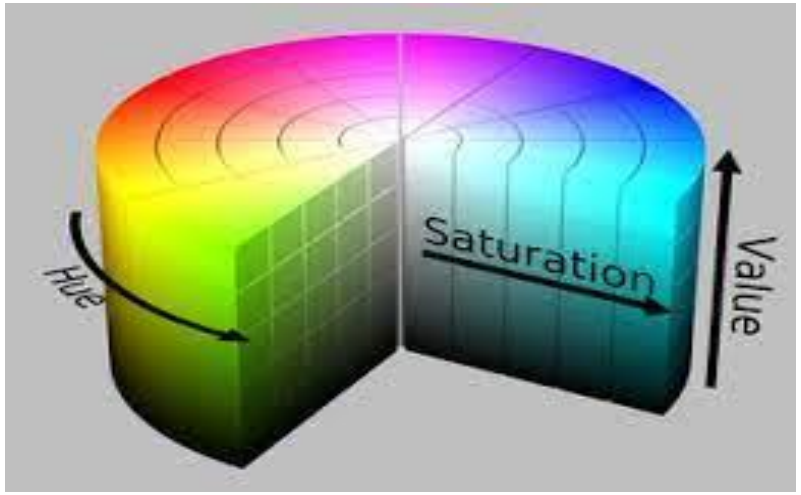


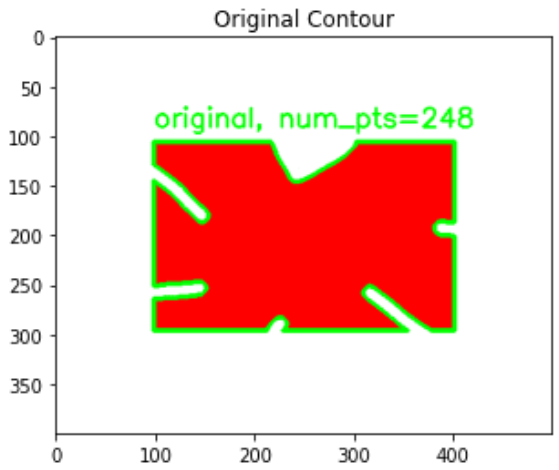
Image Operations

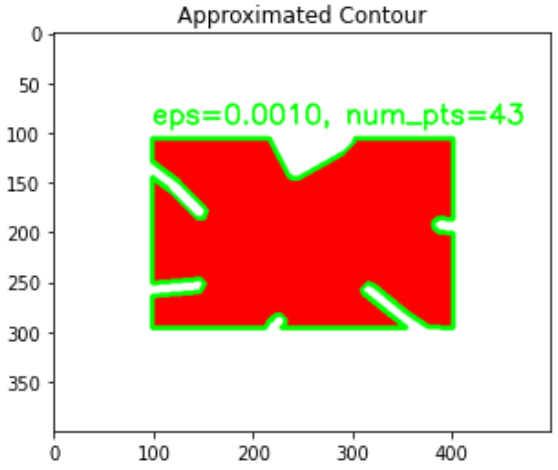
1. Pixel based operations
    1. Contrast Stretching
    2. Thresholding
    3. Inverting/Negative Images
    4. Erosion and Dilation
    5. Bitwise operations, etc.
2. Regions based operations
    1. Contour Detection
    2. Edge Detection
    3. Line Detection, etc.

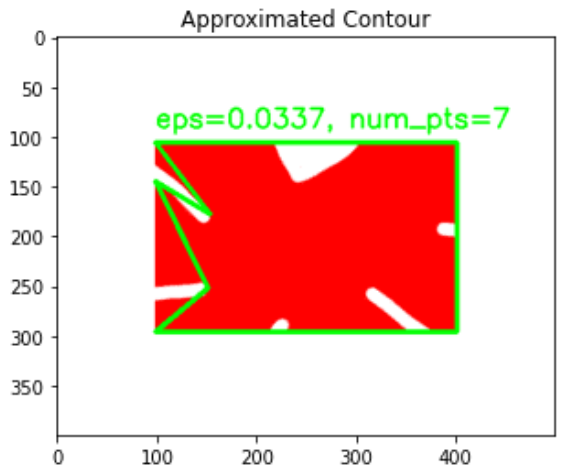
Contour Detection

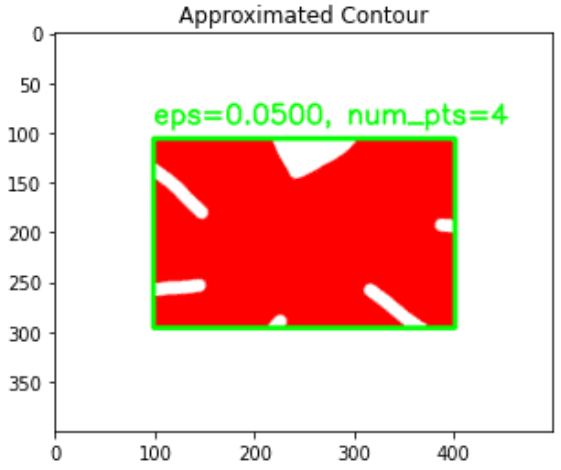
Contours are defined as the line joining all the points along the boundary of an image that are having the same intensity. Contours come handy in shape analysis, finding the size of the object of interest, and object detection.

Simplifying Contours by Approximation









https://pyimagesearch.com/2021/10/06/opencv-contour-approximation/

https://learnopencv.com/contour-detection-using-opencv-python-c#Steps-for-Finding-and-Drawing-Contours-in-OpenCV

Thank you

For any queries drop an email at: quadeershaikh15.8@gmail.com