

# 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

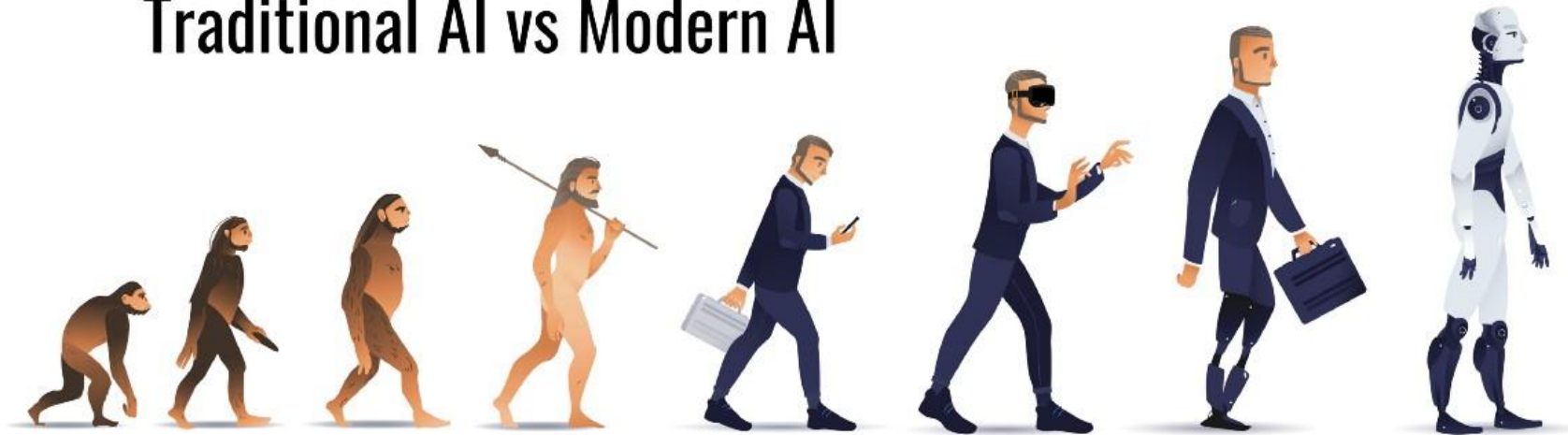
- M.Tech – Artificial Intelligence
  - NMIMS (2021 - 2023, currently pursuing)
- B.E. – Computer Engineering
  - Mumbai University (2015 - 2019)

# Introduction

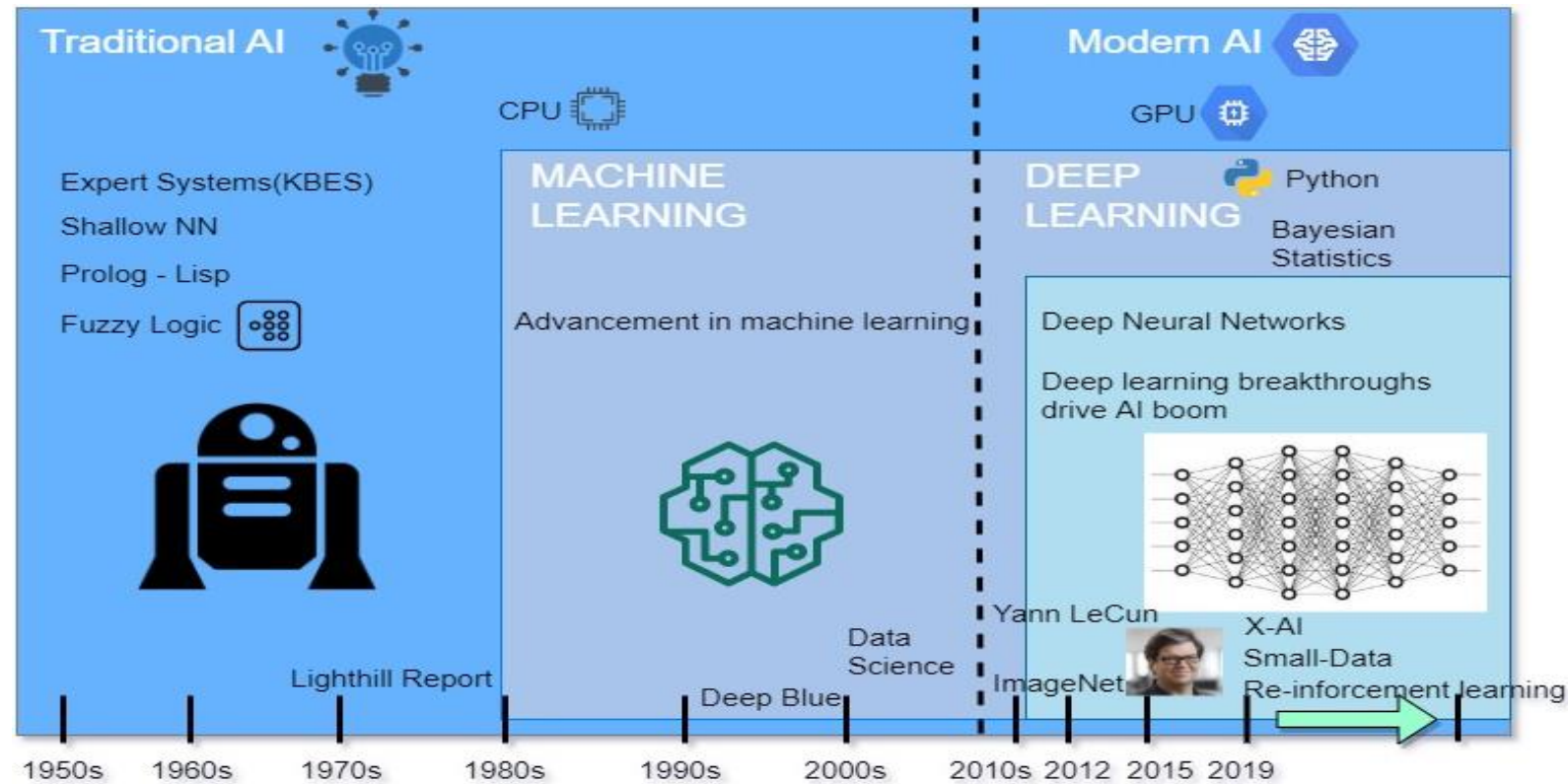
---

- Classical/Traditional AI
- Modern Day AI

## Traditional AI vs Modern AI



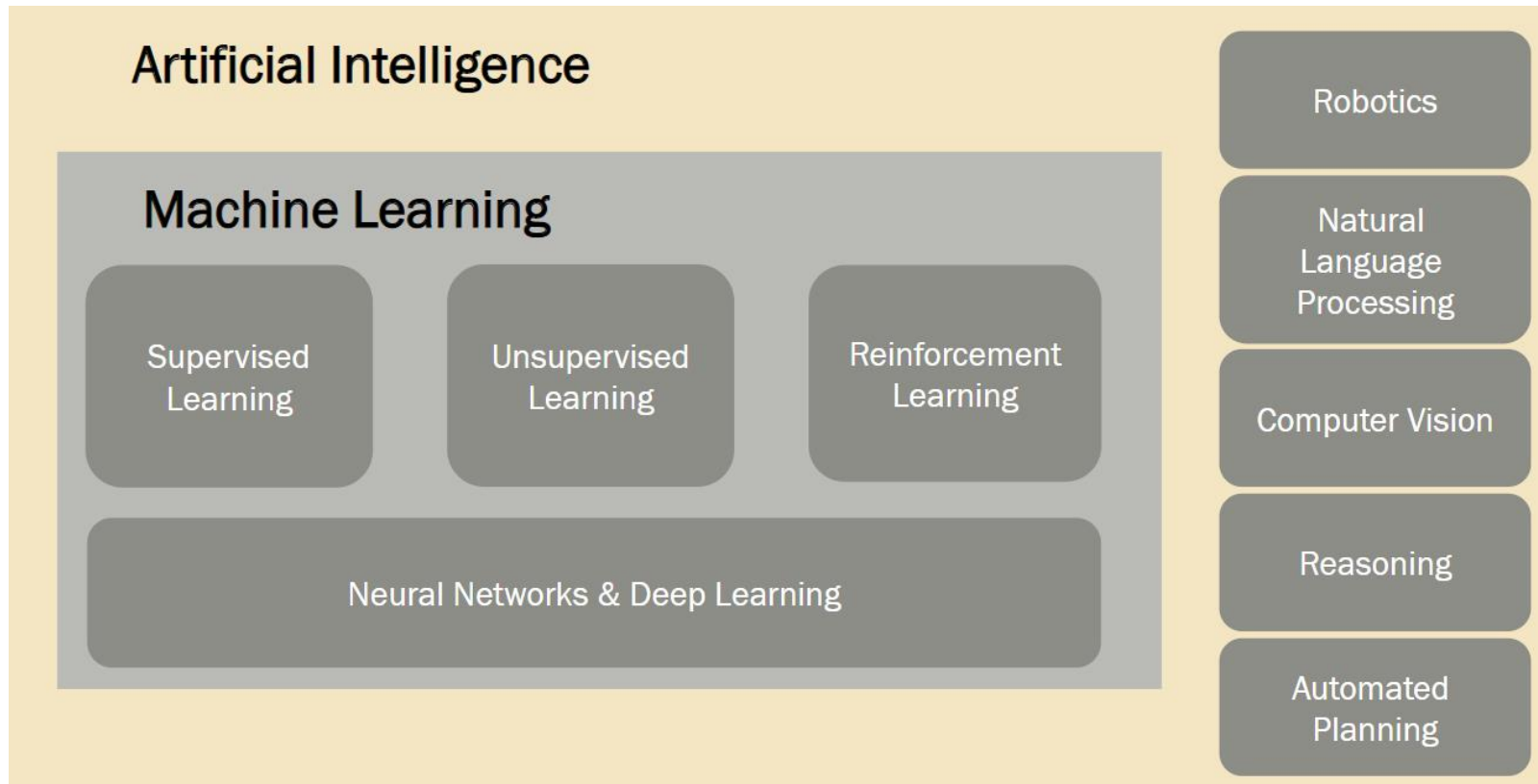
# Traditional/Classical vs Modern AI



Source: Awais Bajwa, Traditional vs Modern AI, TowardsDataScience

# AI in a Nutshell

---

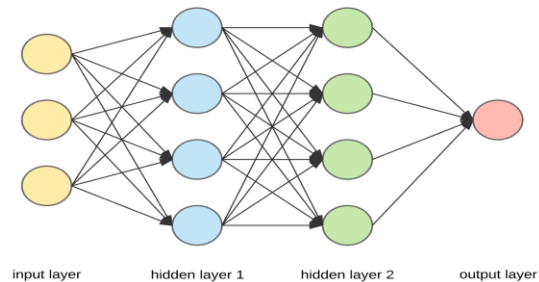


Source: Dr. Prasad Ramanathan

# Capabilities of AI

---

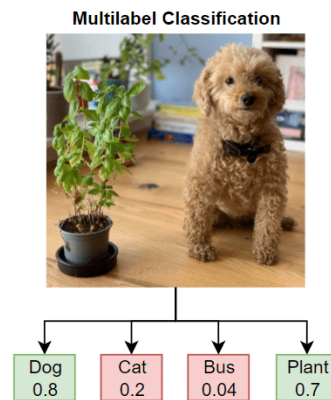
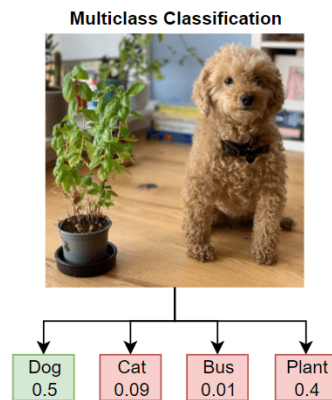
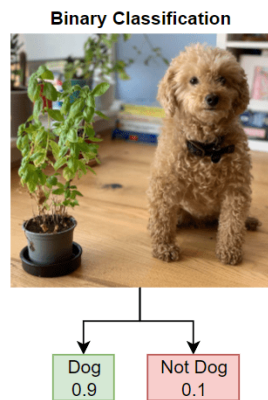
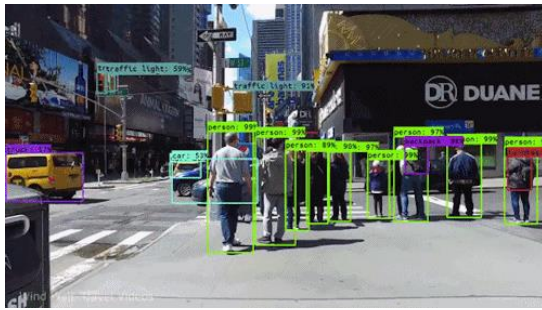
- Natural Language Processing
- Knowledge Representation
- Automated Reasoning
- Machine Learning
- **Computer Vision**
- Robotics





# 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

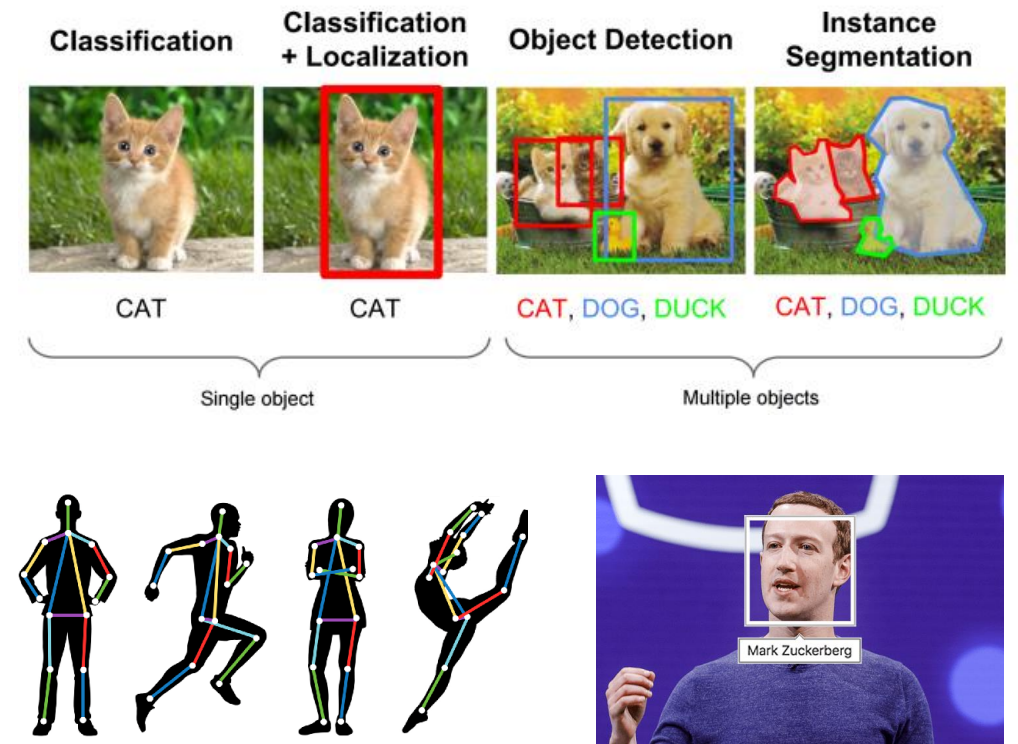
1. Data Driven approach (requires image data)
2. Feature extraction and pattern recognition is **learnt** during the training process
3. Training neural networks on a dataset of images for the desired task (classification, segmentation, object detection, etc)



# 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

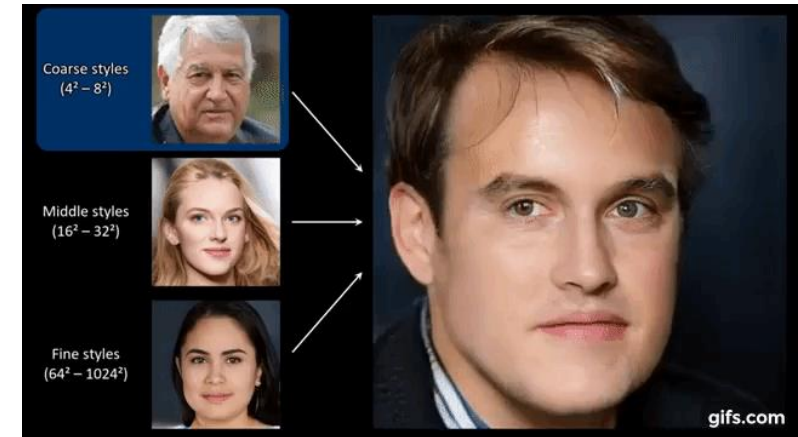
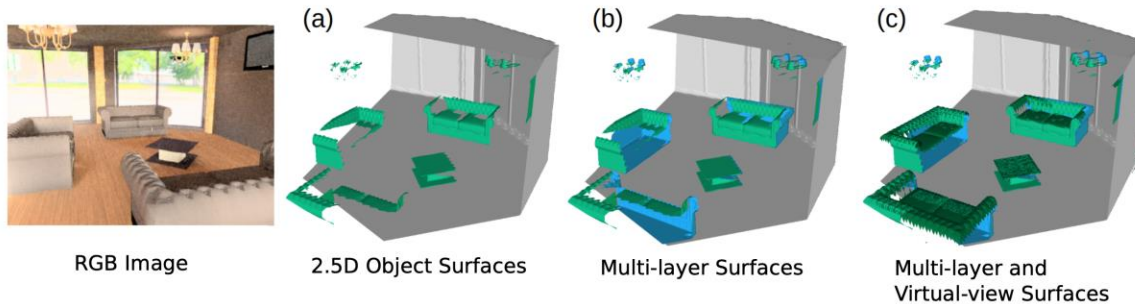


# Typical Tasks in Computer Vision

---

## 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

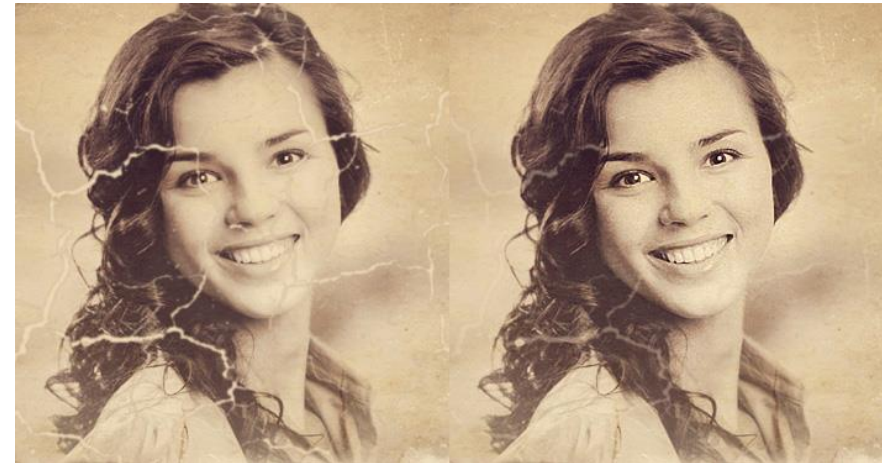
Original



Noisy image



Denoised image



# Applications of Computer Vision

---

## 1. Transportation

- Self Driving Cars, Pedestrian Detection, Traffic Flow Analysis

## 2. Healthcare

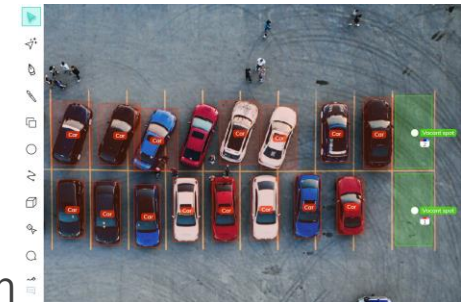
- Xray Analysis, Cancer Detection, Movement Analysis, CT and MRI

## 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

---

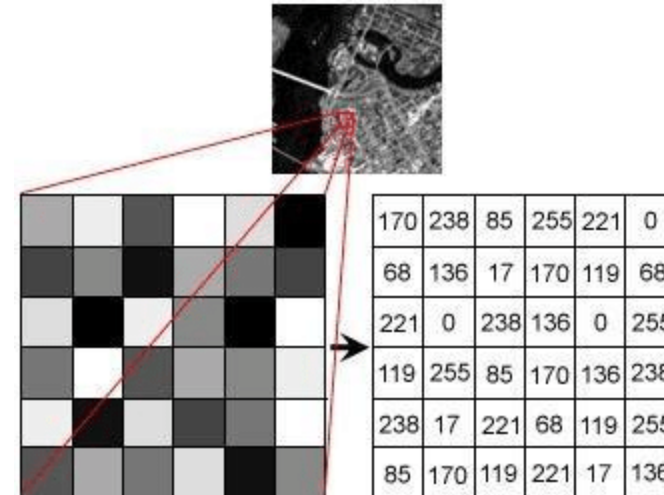
1. Digital Image as a 2D function
2. Image Representation
3. Color Spaces in Image

# 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.

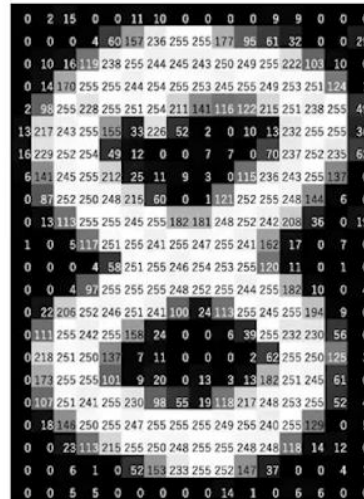
$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & f(0,2) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & f(1,2) & \dots & f(1,N-1) \\ \vdots & \vdots & \vdots & & \vdots \\ \vdots & \vdots & \vdots & & \vdots \\ f(M-1,0) & f(M-1,1) & f(M-1,2) & \dots & f(M-1,N-1) \end{bmatrix}$$



# 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.



0	2	15	0	0	11	10	0	0	0	0	9	9	0	0	0
0	0	0	4	60	157	236	255	255	177	95	61	32	0	0	29
0	10	16	115	238	255	244	245	243	250	249	255	222	103	10	0
0	14	170	255	255	244	254	255	253	245	255	249	253	251	124	1
2	96	255	228	255	251	254	211	141	116	122	215	251	238	255	49
13	217	243	255	155	33	226	52	2	0	10	13	232	255	255	36
16	229	252	254	49	12	0	0	7	7	0	70	237	252	235	62
6	141	245	255	212	25	11	9	3	0	115	236	243	255	137	0
0	87	252	250	248	215	60	0	1	121	252	255	248	144	6	0
0	13	118	255	255	245	255	182	181	248	252	242	208	36	0	19
1	0	5	117	251	255	241	255	247	255	241	162	17	0	7	0
0	0	0	4	58	251	255	246	254	253	255	120	11	0	1	0
0	0	4	97	255	255	255	248	252	255	244	255	182	10	0	4
0	22	206	252	246	251	241	100	24	113	255	245	255	194	9	0
0	111	255	242	255	155	24	0	0	6	39	255	232	230	56	0
0	218	251	250	137	7	11	0	0	0	2	62	255	250	128	3
0	173	255	255	101	9	20	0	13	3	13	182	251	245	61	0
0	107	251	241	255	230	98	55	19	118	217	248	253	255	52	4
0	18	148	250	255	247	255	255	255	249	255	240	255	129	0	5
0	0	23	113	215	255	250	248	255	255	248	248	118	14	12	0
0	0	6	1	0	52	153	233	255	252	147	37	0	0	4	1
0	0	5	5	0	0	0	0	0	14	1	0	6	6	0	0

# 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

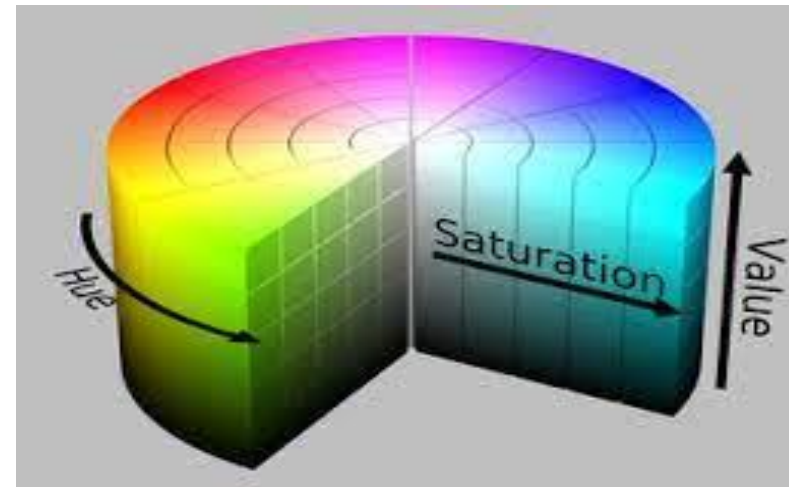
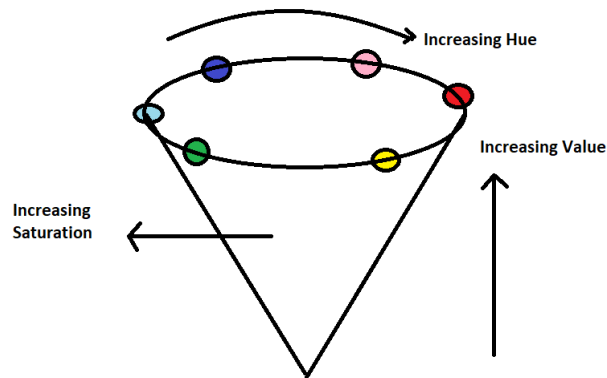
1. Grayscale – 1 channel, Dimensions: (height x width)
2. RGB – 3 channels, Dimensions: (height x width x channels)
3. HSV – 3 channels, Dimensions: (height x width x channels)

**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

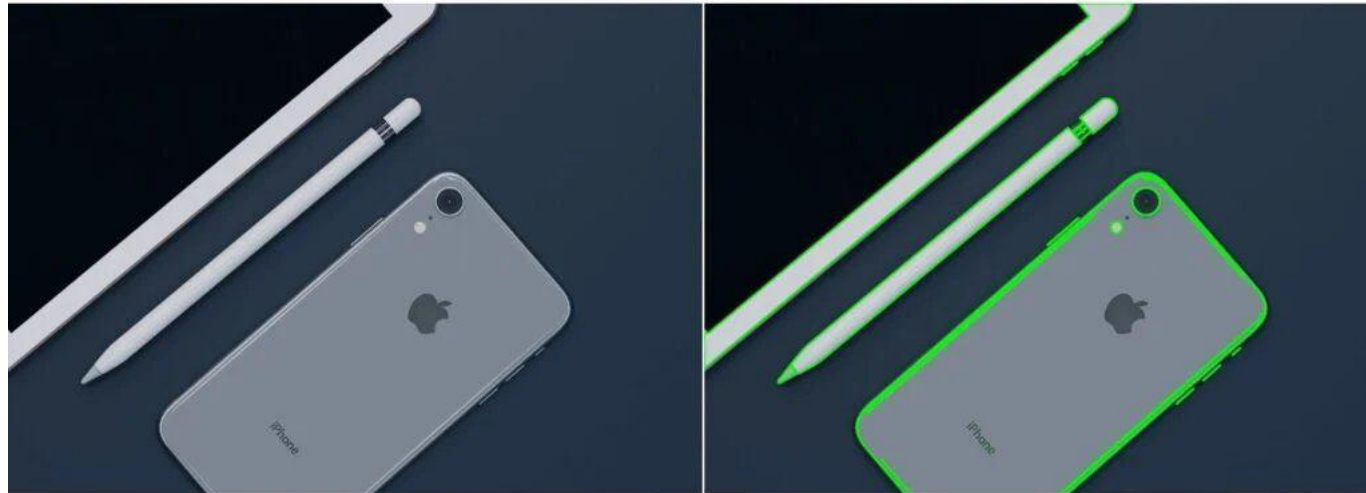
---

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

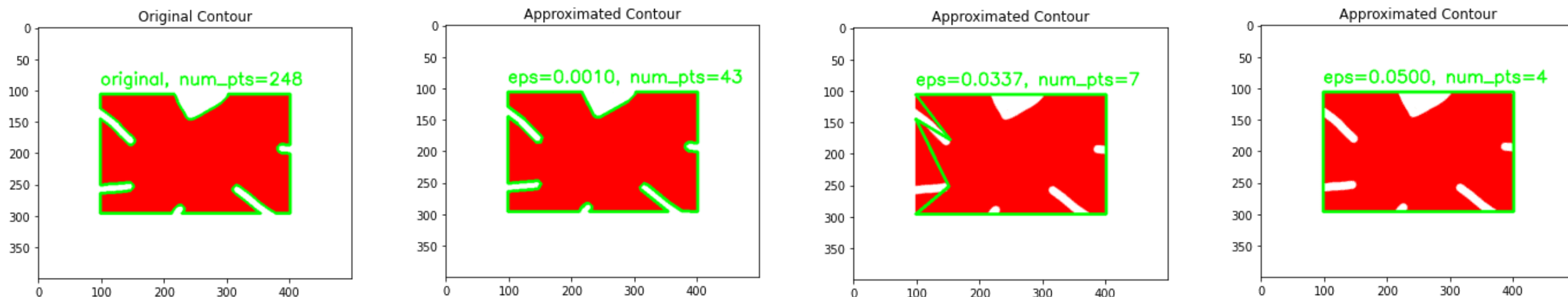
---

**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](mailto:quadeershaikh15.8@gmail.com)