Computer Vision

Dr. Sunita Dhavale, DIAT

Online Training & Certification Course on Artificial Intelligence & Machine Learning
Defence Institute of Advanced Technology (DU), Pune.

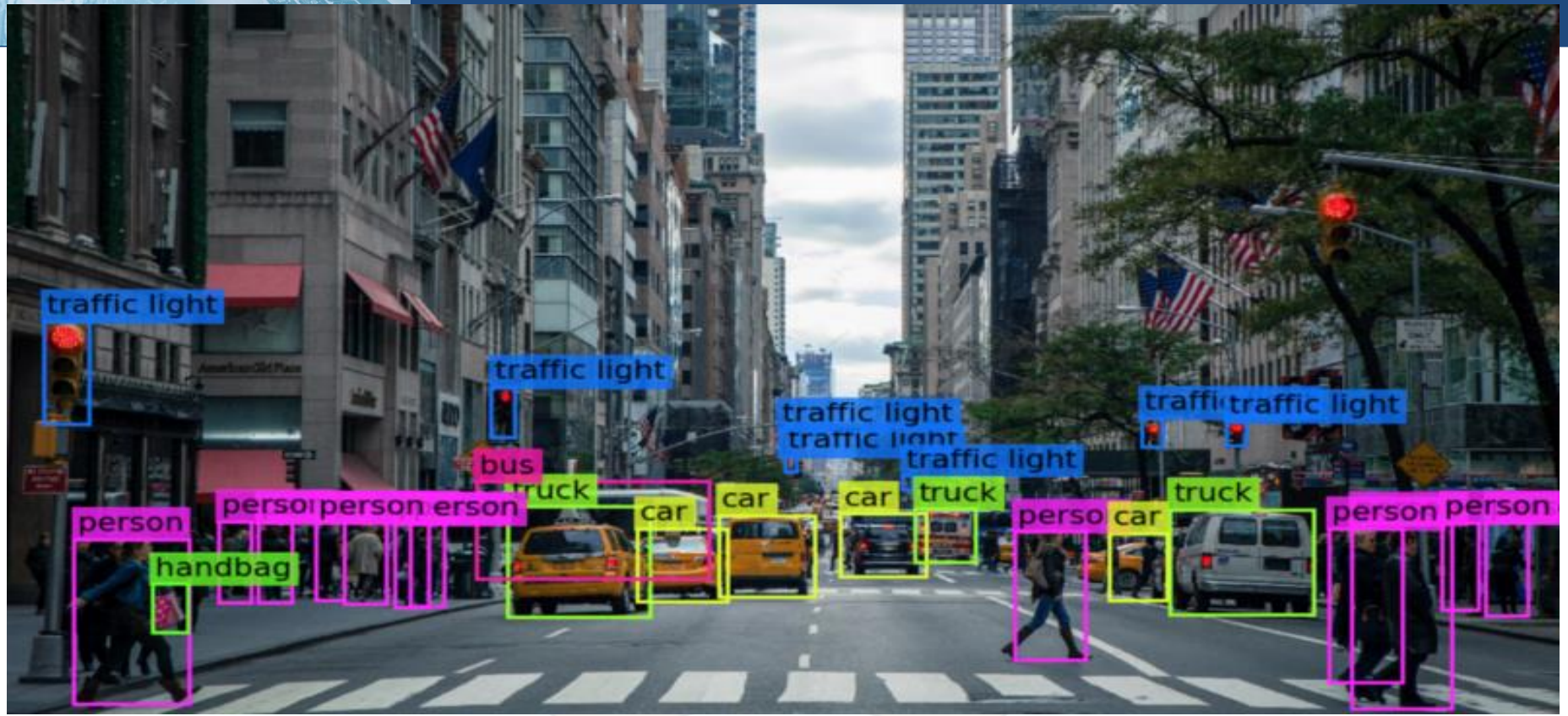
Computer Vision

Dr Sunita Dhavale

Introduction to Image Processing Techniques

Online Training & Certification Course on AI & ML
Defence Institute of Advanced Technology (DU), Pune.

Introduction



Syllabus

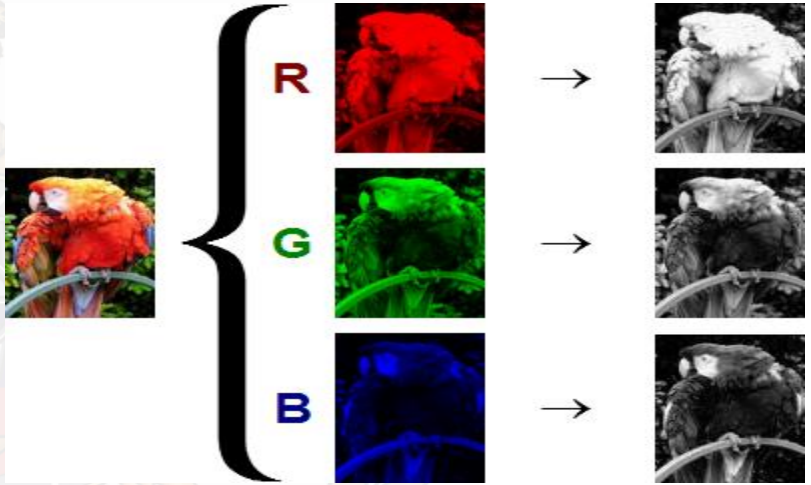
Part I (10 Hrs)	Part II (10 Hrs)
Introduction to Image processing techniques Images, Noise, Convolution, Filtering Thresholding techniques, Image segmentation Edge Detection techniques Interest Point Detection, Harris Corner Detector SIFT, Histograms of Oriented Gradients Binary shape analysis, connectedness, object labeling and counting Boundary tracking procedures, active contours Boundary descriptors, chain codes, Fourier descriptors, region descriptors, moments Hough Transform	Optical Flow, Motion Models, Global Motion KLT Tracking, Mean-Shift Tracking Deep Sort Camera Model and Calibration Fundamental Matrix, Stereo Images 3 D Image processing Deep learning for Chest X-ray Image analysis Face Recognition based on video Human activity detection based on video Audio/speech, based personality detection/prediction

Outline of Presentation

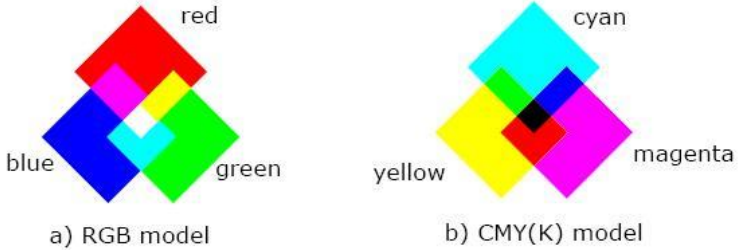
- Introduction
- Images
- Color Spaces
- Color Models
- Python and OpenCV

Images

- Pixels
- Grayscale
- RGB
- Black and White
- Resolution
- Size



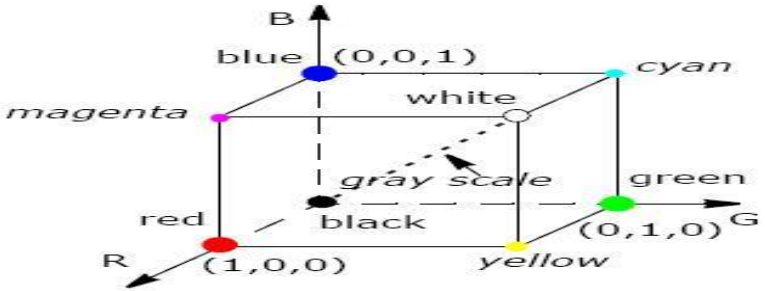
Color Spaces



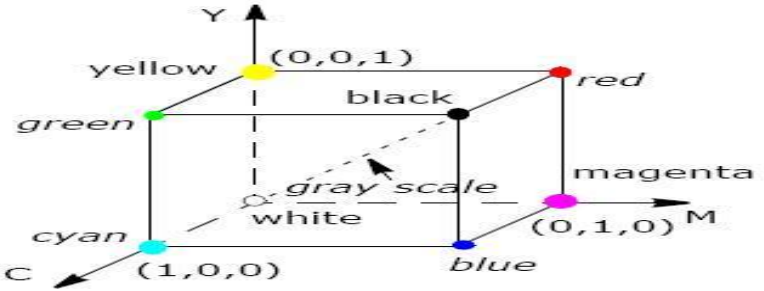
a) RGB model

b) CMY(K) model

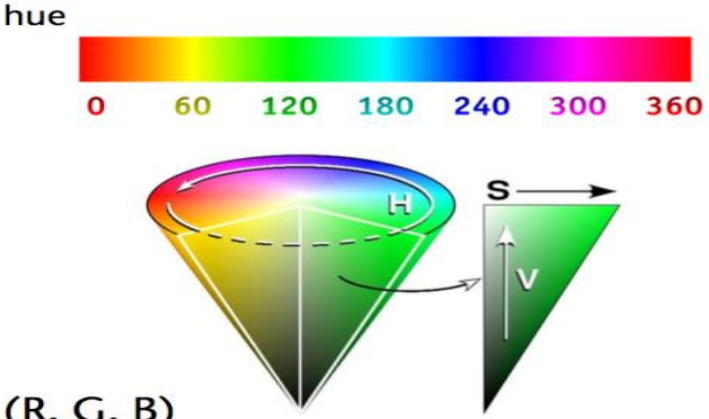
a) RGB



b) CMY



Color Models



- colour cone
  - H = hue / colour in degrees $\in$ [0,360]
  - S = saturation $\in$ [0,1]
  - V = value $\in$ [0,1]
- conversion RGB $\rightarrow$ HSV
  - V = max = max (R, G, B), min = min (R, G, B)
  - S = (max – min) / max (or S = 0, if V = 0)
  - $$H = 60 \times \begin{cases} 0 + (G - B) / (\max - \min), & \text{if } \max = R \\ 2 + (B - R) / (\max - \min), & \text{if } \max = G \\ 4 + (R - G) / (\max - \min), & \text{if } \max = B \end{cases}$$

    $H = H + 360$, if $H < 0$

Python and OpenCV

- On Windows-> Install Anaconda, Spyder, Python framework, OpenCV, skimage, matplotlib, numpy
- Anaconda powershell -> conda info --envs
- Use pip install on python prompt or conda create newenv, conda activate newenv, conda install package on anaconda command prompt
- OpenCV -> open source computer vision, Version 3.4.3 of OpenCV, in conjunction with version 3.7 of Python
- Start anaconda navigator
- Start spyder -> Scientific Python Development Environment
- To verify the installation is error-free->
  - >>import cv2
  - >>print(cv2.__version__)

Try Following Code:

```
from PIL import Image
Img=Image.open('E:/computervision/Lena.png') #read Image
import matplotlib.pyplot as plt
plt.imshow(Img) #display image
import numpy as np
print(np.shape(Img)) #see the size (512, 512, 3)
#convert into numpy array
img1=np.asarray(Img)
#read only one channel
plt.imshow(img1[:,:,1],cmap='gray')
#see pixel values
img1[1:10,1:10,1]
#see each channel of RGB color image
r,g,b=Img.split()
img1=np.asarray(g)
plt.imshow(Img)
```

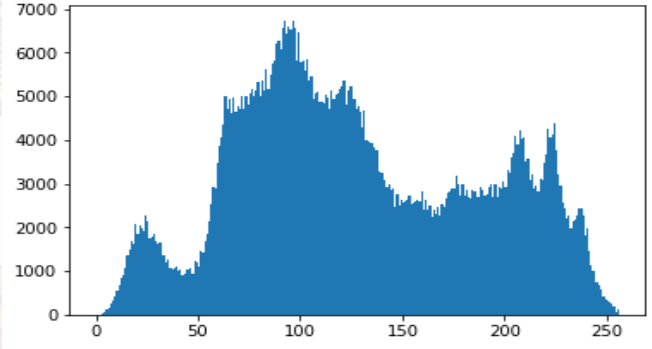

```
array([[137, 137, 136, 138, 129, 138, 134, 140, 136],
       [137, 137, 136, 138, 129, 138, 134, 140, 136],
       [137, 137, 136, 138, 129, 138, 134, 140, 136],
       [137, 137, 136, 138, 129, 138, 134, 140, 136],
       [140, 131, 130, 136, 133, 132, 133, 136, 134],
       [134, 141, 133, 134, 137, 132, 128, 134, 137],
       [133, 129, 132, 131, 133, 129, 131, 131, 137],
       [129, 133, 133, 134, 134, 130, 132, 139, 131],
       [130, 133, 134, 128, 127, 129, 130, 135, 128]],
dtype=uint8)
```

Try:

- from skimage.color import rgb2hsv
- hsvimg=rgb2hsv(Img)
- print(img1[1:10,1:10,0])
- print(hsvimg[1:10,1:10,0])
- print(hsvimg[1:10,1:10,1])
- print(hsvimg[1:10,1:10,2])
- #if RGB=255 then H=S=0 and V=1

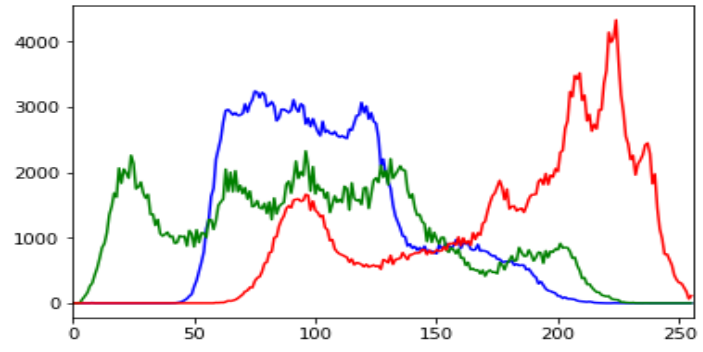
Histogram

- another representation for images, is the frequency of intensity value in the image, captures the distribution of gray levels in the image
- def histogram(im):
  - h = np.zeros(255)
  - for row in im.shape[0]:
  - for col in im.shape[1]:
  - val = im[row, col]
  - h[val] += 1



Try:

```
import numpy as np
import cv2
from matplotlib import pyplot as plt
img = cv2.imread("E:/computervision/Lena.png")
#plot a histogram
histogram_image = cv2.calcHist([img],[0],None,[256],[0,256])
hist,bins = np.histogram(img.ravel(),256,[0,256])
np.shape(hist)  ## (256,)
hist[1:10]  ## array([  0,   0,  11,  65, 111, 164, 261, 315, 431], dtype=int64)
#flaten the histogram
plt.hist(img.ravel(),256,[0,256])
plt.show()
#view color channels
color = ['b','g','r']
#seperate the colors and plot the histogram
for i, col in enumerate(color):
    hist = cv2.calcHist([img], [i], None, [256], [0,256])
    plt.plot(hist, color = col)
    plt.xlim([0,256])
plt.show ()
```



- Homework: find mean, median, mode, std, var, count, min , max on histogram (use numpy)

Reference Material

- 1. E. R. Davies, “Computer & Machine Vision”, Fourth Edition, Academic Press, 2012.
- 2. R. Szeliski, “Computer Vision: Algorithms and Applications”, Springer 2011.
- 3. Simon J. D. Prince, “Computer Vision: Models, Learning, and Inference”, Cambridge University Press, 2012.
- 4. Mark Nixon and Alberto S. Aquado, “Feature Extraction & Image Processing for Computer Vision”, Third Edition, Academic Press, 2012.
- 5. Sunita Dhavale, “Advanced Image-Based Spam Detection and Filtering Techniques”, Book Published by CyberTech: An Imprint of MKP Technologies, Hershey, PA, USA IGI Global, March 2017, ISBN13: 9781683180135|ISBN10: 1683180135|EISBN13: 9781683180142|DOI: 10.4018/978-1-68318-013-5.
- 6. Gonzalez and R. Woods Digital Image Processing, Addison-Wesley Publishing Company, 1992, p 442.

<<Epilogue>>

- We will meet in next scheduled lecture.
- Start using python to check various image properties.
- Feel free to ask your questions.
- Email: sunitadhavale@diat.ac.in

Thank You!