Value Added Program - Python Vision Techniques

AN INDUSTRIAL INTERNSHIP TRAINING REPORT

Submitted by

Andrew John

18BEC1278

ECE3099 – INDUSTRIAL INTERNSHIP

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

in

ELECTRONICS AND COMMUNICATION ENGINEERING

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

OCTOBER 2021

School of Electronics Engineering

DECLARATION BY THE CANDIDATE

I hereby declare that the Industrial Internship Report entitled “**Value Added Program - Python Vision Techniques”** submitted by me to VIT, Chennai in partial fulfillment of the requirement for the award of the degree of **Bachelor of Technology** in **Electronics and Communication Engineering** is a record of bonafide industrial training undertaken by me under the supervision of **Dr. Sathiya Narayanan Sekar, SENSE Assistant Professor, VIT, Chennai** and **Dr. Annis Fathima, SENSE Associate Professor, VIT, Chennai.** I further declare that the work reported in this report has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Chennai **Signature of the Candidate**

Date: 05/10/2021

Register Number: 18BEC1278
Student Name: Andrew John

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

CHENNAI

CERTIFICATE OF COMPLETION

This is to certify that Mr./Ms.……………………ANDREW JOHN……………………(18BEC1278)……….of ……………B.TECH ELECTRONICS AND COMMUNICATION ENGINEERING…………………… has successfully completed the Value Added Course **"Python for Vision Techniques"** organized by the **School of Electronics Engineering (SENSE),** Vellore Institute of Technology (VIT) – Chennai, from 25-February-2021 to 10-April-2021. His/her consolidated score is 90 out of 100.

Dr. Annis Fathima A	**Dr. Sathya Narayanan S**	**Dr. Sivasubramanian A**	**Dr. Kanchana Bhaaskaran**
Faculty Coordinator	Faculty Coordinator	Dean-SENSE	Pro Vice Chancellor

VIT – Recognised as Institution of Eminence (IoE) by Government of India

VIT - A place to learn; A chance to grow

School of Electronics Engineering

BONAFIDE CERTIFICATE

This is to certify that the Industrial Internship Report entitled "**Value Added Program - Python Vision Techniques**" submitted by **Andrew John (18EC1278)** to VIT University, Chennai in partial fulfillment of the requirement for the award of the degree of **Bachelor of Technology** in **Electronics and Communication Engineering** is a record of bonafide internship undertaken by him/her fulfills the requirements as per the regulations of this institute and in my opinion meets the necessary standards for submission. The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Signature of the Examiner

Date:

Signature of the Examiner

Date:

Head of the Department (B.Tech ECE)

ACKNOWLEDGEMENT

I wish to thank those who were involved in the successful completion of my **Value Added Program - Python Vision Techniques** at VIT Chennai and I also thank for the opportunity given to me by **Dr. Sathiya Narayanan Sekar, SENSE Assistant Professor, VIT, Chennai and Dr. Annis Fathima, SENSE Associate Professor, VIT, Chennai**, School of Electronics Engineering, for their consistent encouragement and valuable guidance offered to us in a pleasant manner throughout the course of the Value Added Program work. I am also grateful for the freedom given to me to learn as per my interests which made my internship a worthwhile experience.

I am extremely grateful to **Dr. Sivasubramanian. A**, Dean of School of Electronics Engineering, VIT Chennai, for extending the facilities of the School towards my Value Added Program and for his unstinting support.

I express my thanks to our Head of the Department **Dr. Vetrivelan. P,** for his support throughout the course of this Value Added Program.

I would also like to thank my parents, for being my motivation to take up this internship; and last, but not the least, management at Vellore Institute of Technology (VIT), Chennai, for providing me with such an avenue to help realize how interesting it is to work in today's industry.

ANDREW JOHN

Table Of Contents

List of Figures

List of Tables

S.No	Table Description	Page No.

List of Abbreviations:

1. OpenCV - Open Computer Vision
2. ANN - Artificial Neural Network
3. CNN - Convolutional Neural Network
4. RNN - Recurrent Neural Network
5. ML - Machine Learning
6. DL - Deep Learning
7. TL - Transfer Learning
8. AI - Artificial Intelligence

ABSTRACT

Computer vision is a field of artificial intelligence (AI) that enables computers and systems to derive meaningful information from digital images, videos and other visual inputs — and take actions or make recommendations based on that information. If AI enables computers to think, computer vision enables them to see, observe and understand.

Scientists and engineers have been trying to develop ways for machines to see and understand visual data for about 60 years. Experimentation began in 1959 when neurophysiologists showed a cat an array of images, attempting to correlate a response in its brain. They discovered that it responded first to hard edges or lines, and scientifically, this meant that image processing starts with simple shapes like straight edges.

The scientific discipline of computer vision is concerned with the theory behind artificial systems that extract information from images. The image data can take many forms, such as video sequences, views from multiple cameras, multidimensional data from a 3D scanner, or medical scanning device. The technological discipline of computer vision seeks to apply its theories and models to the construction of computer vision systems.

This course is focused on how to use OpenCV - Python to process images and apply some of the well-known computer vision techniques, get some useful information from the said images, and infer insightful takeaways. Along with the techniques performed, various real-time applications of computer vision are discussed where state-of-the-art technologies are used along with it.

CHAPTER 1

Introduction

1.1 About VIT

Founded in 1984, VIT has made a mark in the field of higher education in India imparting quality education in a multi-cultural ambience, intertwined with extensive application-oriented research. VIT was established with the aim to provide quality higher education on par with International Standards. It persistently seeks and adopts innovative methods to improve the quality of higher education on a consistent basis. VIT was established by well-known educationist and former parliamentarian, Dr. G. Viswanathan, Founder and Chancellor, a visionary who transformed VIT into a center of excellence in higher technical education. The Govt. of India recognized VIT as an Institution of Eminence (IoE). VIT Chennai is ably spearheaded by Mr. Sankar Viswanathan, Vice-President, Ms. Kadhambari S. Viswanathan, Assistant Vice-President, Dr Rambabu Kodali, Vice-Chancellor and Dr. V. S. Kanchana Bhaaskaran, Pro-Vice Chancellor. They share in the mission to make VIT a global center towards academic and research excellence.

The focus is to:

- To maximize the interactive Industrial Connectivity.
- To create Centers of Excellence in niche areas.
- To enrich Technological and Managerial Human Capital nurtured in a multicultural ambience.
- To provide a common platform for the agglomeration of ideas of personnel from various walks of life for learning enrichment.
- To create opportunities and exploit the available resources to benefit industry and society.
- To encourage participation in the National Agenda of knowledge building.
- To foster International collaborations for mutual benefits in areas of research.

1.2 About School - SENSE

The School of Electronics Engineering at VIT was established for imparting the state-of-the-art education, training and research in the field of Electronics and Communication Engineering and allied areas. It offers B.Tech Programmes in Electronics and Communication Engineering and Electronics and Computer Engineering, M.Tech Programmes in VLSI Design and Embedded Systems, and Ph.D. in the related domains of ECE. The expertise of the faculty members includes VLSI Design, Communication Engineering, Embedded Systems, Nano-electronics and Nano-technology, Photonics, Signal Processing, Machine Learning, Artificial Intelligence and Data Analytics.

1.3 About the Value Added Program

The motive is to enlighten them on start-of-the-art computer vision techniques and their implementation using Python. The course gave good insights into understanding images. Basic array operations on images using python were performed. Spatial domain operations and image enhancement using python gave good insights into various filters and operators used in image enhancement. Image segmentation and morphology using python helped in detecting edge and corner discrepancies in images and along with this image classification using python through k-Means was performed. CNN for vision applications and Deep Learning concepts in image processing was delivered through guest faculty. State-of-the-art Computer Vision applications were also delivered through the following guest faculty.

CHAPTER 2

Introduction to Image Processing and Basic Array Operations

2.1 Studying the Color-Plane information

In this given task, the layers of color-plane were studied. A color image is represented and stored as a set of three matrices each of size MXN. Each matrix represents a colour plane. Thus if an RGB model is used, we have a red image, blue image and a green image and thus 3 corresponding matrices. The RGB and CMY colour models can be visualized as forming a colour cube shown below. Here, red, green and blue form the three orthogonal edges of the cube while cyan, magenta and yellow form the opposite set of edges of the same cube. Note that the corner (S) where the RGB edges meet corresponds to black colour while the corner (W) where the CMY edges meet corresponds to the white colour. Any point within this cube can therefore be specified in terms of 3 coordinates, namely RGB or CMY values. The diagonal line that connects the black and white points will correspond to the grayscale.

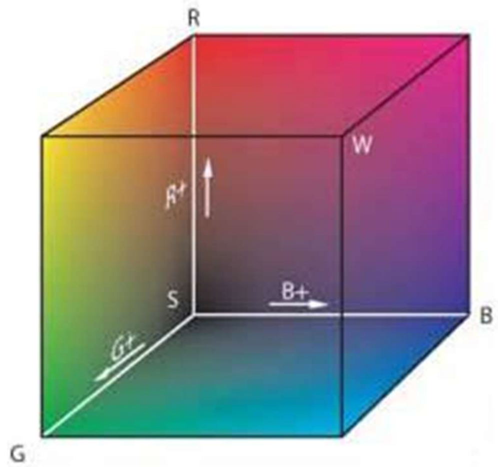


Fig-1: RGB Color Space

Relationship between the RGB and CMY colour model is as follows.

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

There are two ways for processing in colour domain

(a) *Scalar processing*: Process each plane of colour model individually. In this processing, one can process only one or two planes and leave the remaining planes unchanged. For example if we want to modify the red component present in the colour image, then modify only the red plane in the RGB colour model and leave other planes unchanged. After processing the image is converted to RGB space for display.

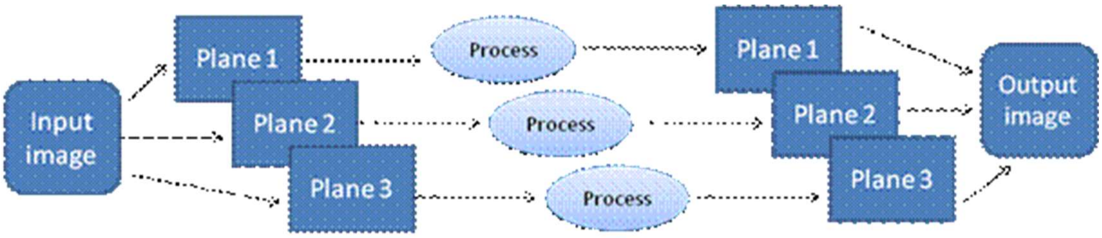


Fig-2: Scalar Processing

(b) *Vector processing* : Consider each pixel of the image as a three element vector, each element corresponding to information from each of the color planes. Instead of processing the image in each plane separately , as we did in the previous case. all planes of colour model are processed simultaneously.

In this task, the following was achieved; where a RGB Layered picture was taken and the aim was to visualize the layers Red, Blue and Green separately.

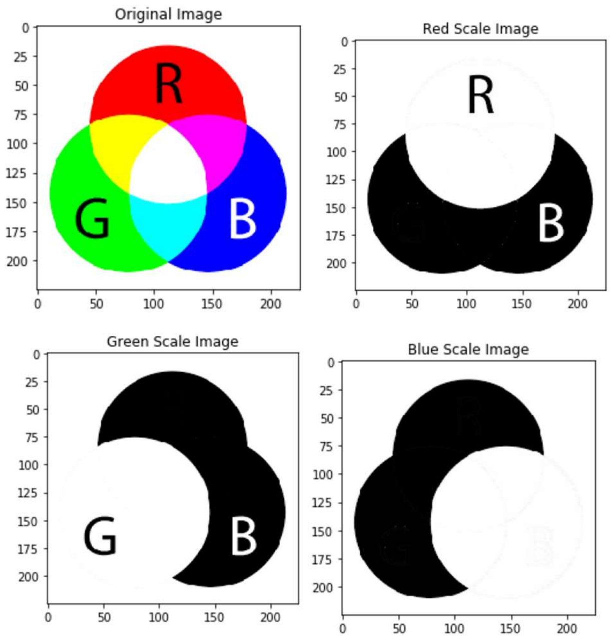


Fig-3: Separate Red, Green and Blue Planes

INFERENCE: The plane that was taken separately, for example, Red, is whitened and the other 2 planes appear black. Similarly it goes for Blue and Green planes.

2.2 <u>Studying the Histogram</u>

The (intensity or brightness) histogram of a digital image with intensity levels in the range [0, L - 1] is a discrete function

$$h(r_k) = n_k, \qquad k = 0, 1, 2, \ldots, L - 1$$

where r_k is the k^{th} intensity value and n_k is the number of pixels in the image with intensity r_k.
The histogram shows how many times a particular intensity level appears. In other words, it shows the distribution of pixel values. Image contrast and brightness can be

assessed from image histogram. The normalized histogram, $p(r_k) = n_k / n$; where n is the total number of pixels in the image, is an estimate of the probability of occurrence of intensity level r_k.

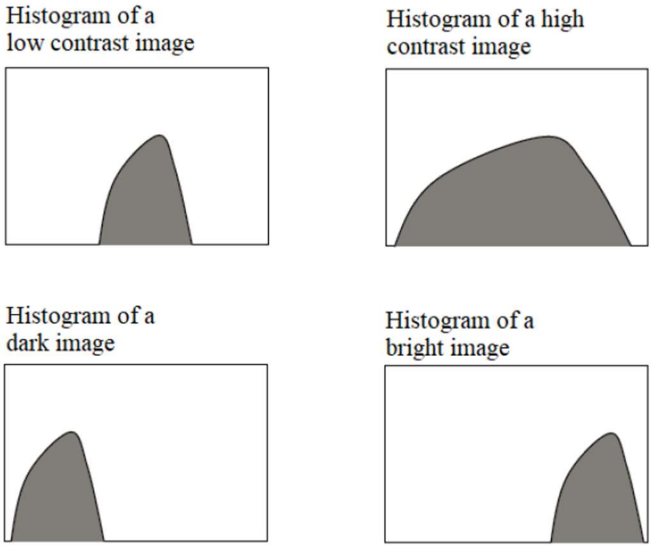


Fig-4: Image contrast and brightness assessment from image histogram.

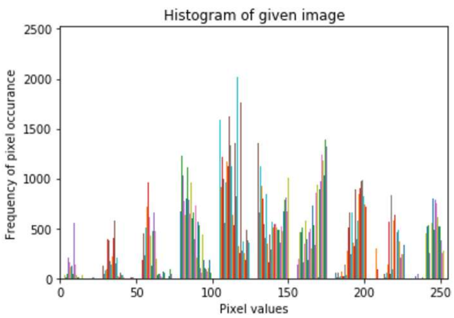
Two types of histogram processing: histogram equalization and histogram matching. The histogram equalization transforms the intensity values such that the histogram of the output image is **nearly flat**. The discrete form of the transformation is:

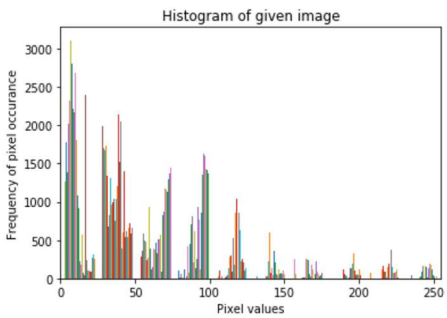
$$s_k = (L-1)\sum_{j=0}^{k} \frac{n_j}{n}, \qquad k = 0, 1, 2, \ldots, L - 1.$$

The discrete histogram equalization in general does not yield a flat histogram. The net result is contrast enhancement. The histogram matching transforms the intensity values such that the output image has a specified histogram.

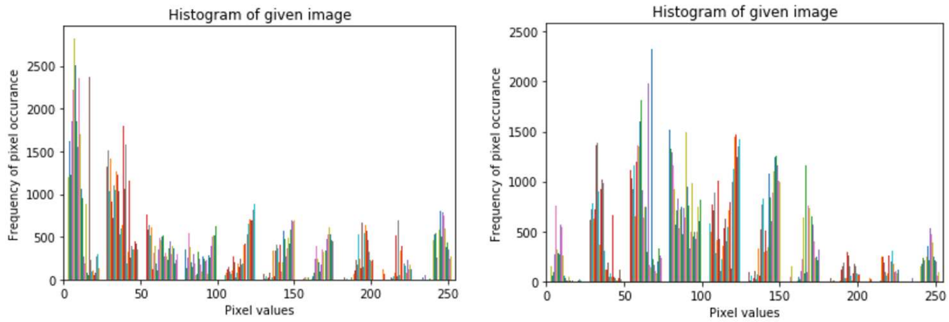
APPLICATIONS: CT lung studies, normalization of MRI images, etc.

In this task, the following was achieved; different images with varying contrast levels (high and low) and brightness levels (dark and bright) were given as input and the histogram of all the images was obtained and assessed.





(a) *(b)*



(c) *(d)*

Fig-5: a-High Brightness, b-Low Brightness, c-High Contrast, d- Low Contrast

INFERENCE: Ideal cases for a Picture with Low Contrast, High Contrast, Dark Image and Bright Image is shown in the fig-3. We can see that the above histograms have the locus of the graphs that is almost similar to that of Ideal case.

2.3 Converting a Grayscale Image to Binary Image

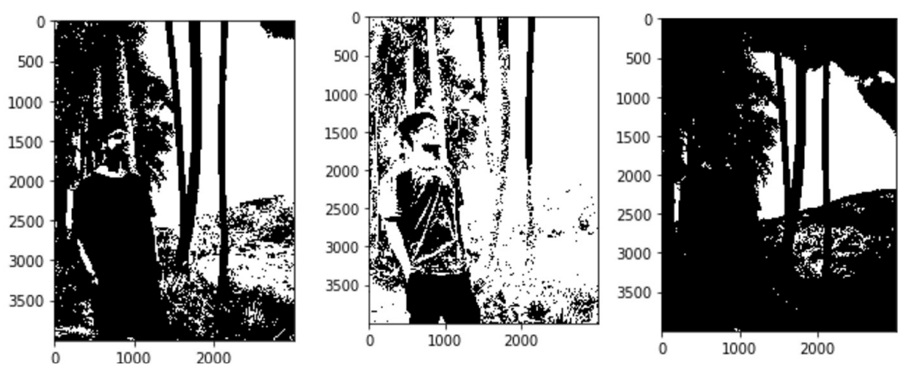
Thresholding is the simplest method of image segmentation and the most common way to convert a grayscale image to a binary image.

In thresholding, we select a threshold value and then all the gray level value which is below the selected threshold value is classified as 0 (black i.e., background) and all the gray levels which are equal to or greater than the threshold value are classified as 1 (white i.e foreground).

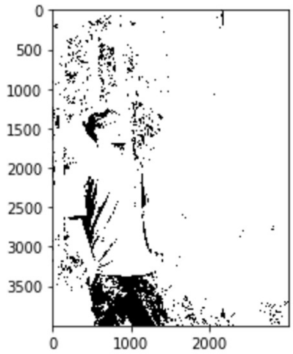
$$g(x,y) = \begin{cases} 1 & \text{if } f(x,y) \geq T \\ 0 & \text{otherwise} \end{cases}$$

Here, g(x, y) represents the threshold image pixel at (x, y) and f(x, y) represents grayscale image pixel at (x, y).

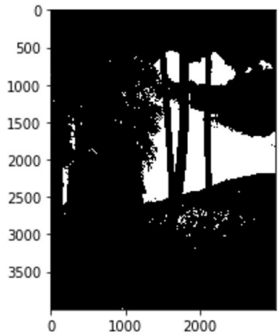
In this task, the following was achieved; The same image at different threshold values was taken. Maximum threshold was given as 255 and minimum was given as 0.



(a) (b) (c)



(d)



(e)

Fig-6: Threshold set at a-100, b-50, c-150, d-25, e-175

INFERENCE: If the pixel is more than the threshold, that part of the array will fill up with ones and ones represent white color. If the pixel is less than the threshold, that part of the array won't be filled up with ones, it'll be zeros and zeros represent black color.

CHAPTER 3

Spatial Domain Operations and Image Enhancement using Python

3.1 Gaussian and Median Filter

The Gaussian smoothing operator is a 2-D convolution operator that is used to `blur' images and remove detail and noise. In this sense it is similar to the mean filter, but it uses a different kernel that represents the shape of a Gaussian (`bell-shaped') hump. This kernel has some special properties which are detailed below. The Gaussian distribution in 1-D has the form:

$$G(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$$

where σ is the standard deviation of the distribution. We have also assumed that the distribution has a mean of zero (i.e. it is centered on the line x=0). The distribution is illustrated in Fig-7.

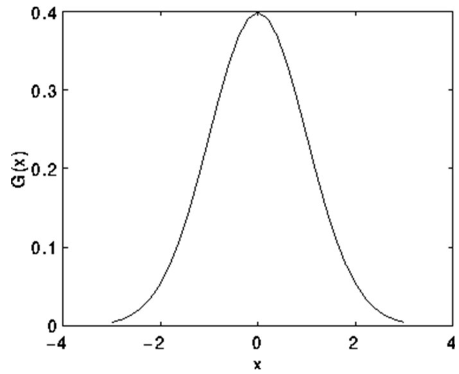


Fig-7: 1-D Gaussian distribution with mean 0 and σ *=1*

Median filtering is a nonlinear process useful in reducing impulsive, or salt-and-pepper noise. It is also useful in preserving edges in an image while reducing random noise. Impulsive or salt-and pepper noise can occur due to a random bit error in a communication channel. In a median filter, a window slides along the image, and the median intensity value of the pixels within the window becomes the output intensity of the pixel being processed. The median filter is normally used to reduce noise in an image, somewhat like the mean filter. However, it often does a better job than the mean filter of preserving useful detail in the image.

In this task, the following was achieved; The image without noise and with external noise was fed at different σ, size, and internal noise (raised, reasonable & reduced when external noise was not given) values and the outputs were observed. Both the gaussian and median filtering was performed.

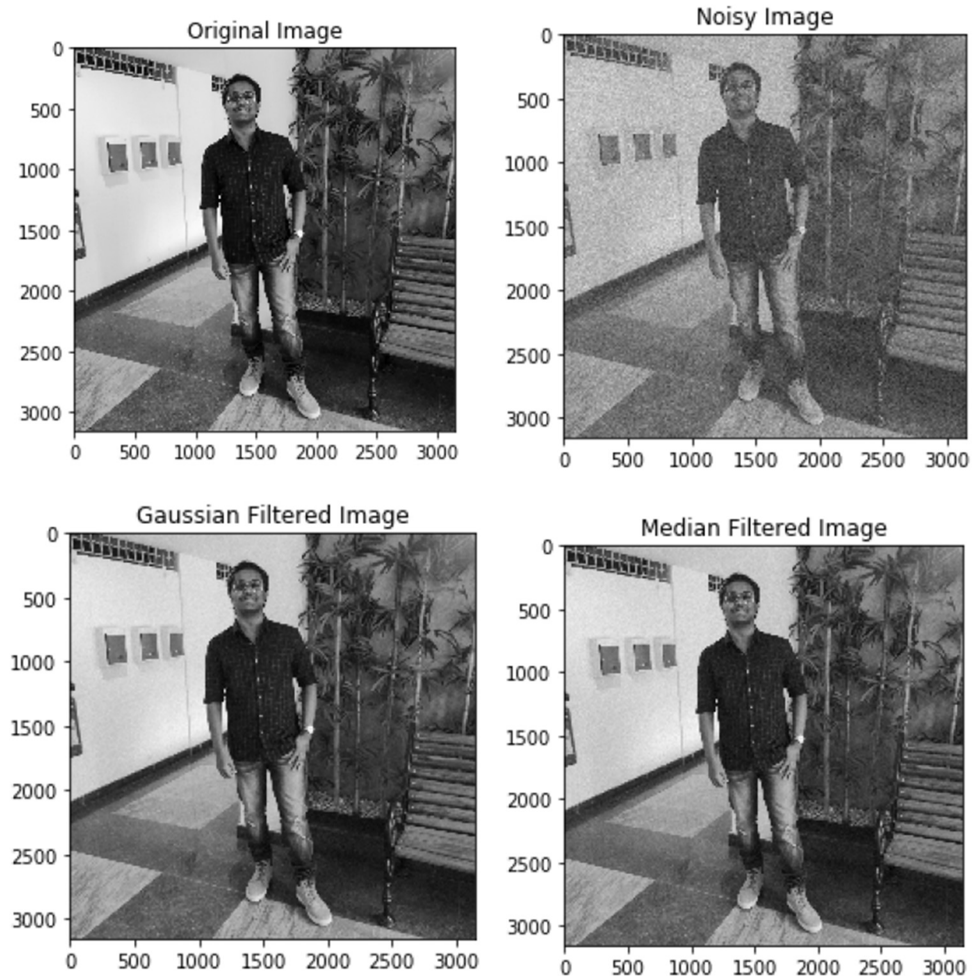
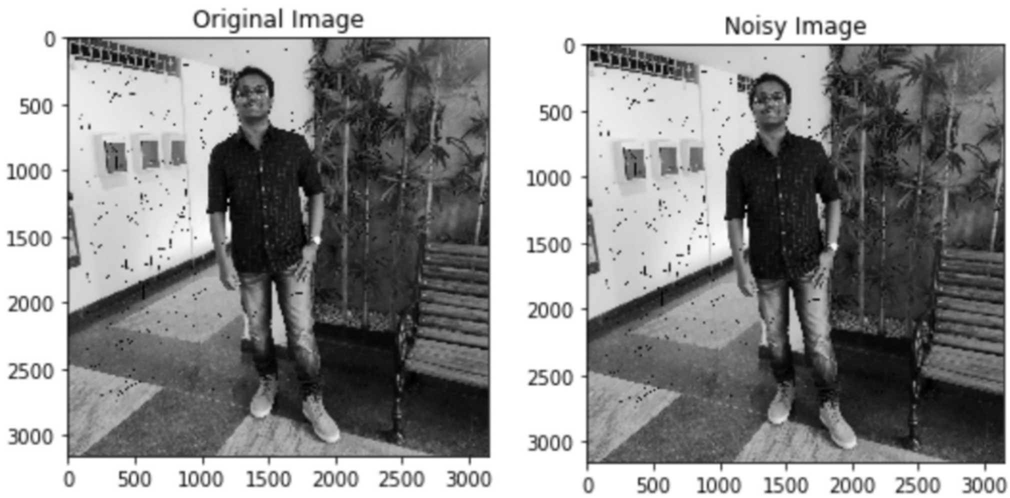


Fig-8: Without External Noise, Internal Noise = 10, sigma=1.5, size=5



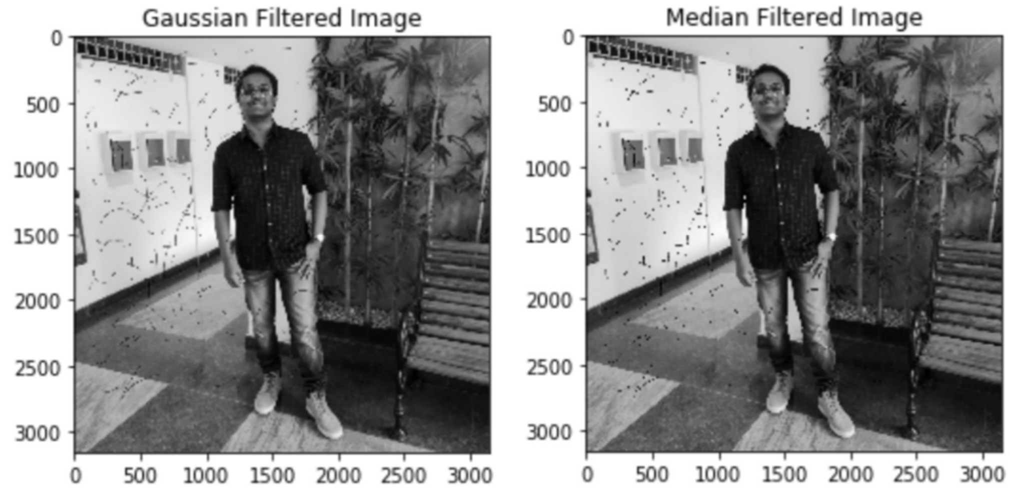


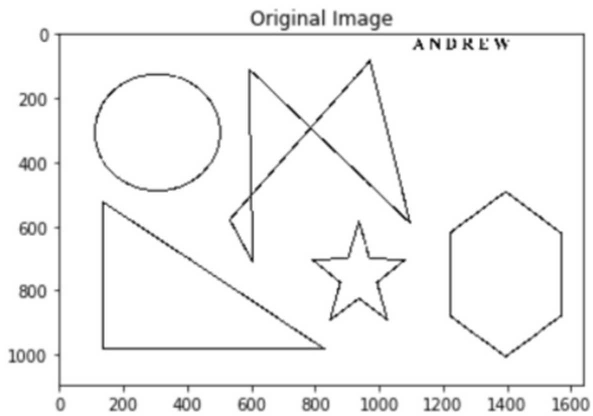
Fig-9: With External Noise, Internal Noise = 0, sigma=1.5, size=5

INFERENCE: We can observe that as we increase the sigma value the picture gets more and more blurred. We can observe that as we increase the median size, these pictures get smoothed but along with that we are starting to observe a loss in the important details of the image. Taking all these into consideration we can infer that gaussian filtering is better than median filtering.

3.2 <u>Minimum and Maximum Filter</u>

The Minimum Filter: The transformation replaces the central pixel with the darkest one in the running window. For example, if you have text that is lightly printed, the minimum filter makes letters thicker.

The Maximum Filter: The maximum and minimum filters are shift-invariant. Whereas the minimum filter replaces the central pixel with the darkest one in the running window, the maximum filter replaces it with the lightest one. For example, if you have a text string drawn with a thick pen, you can make the sign skinnier.
In this task, the following was achieved; Images with clear edges were given and minimum and maximum filter of value 5 was applied and the following was obtained with zero noise.



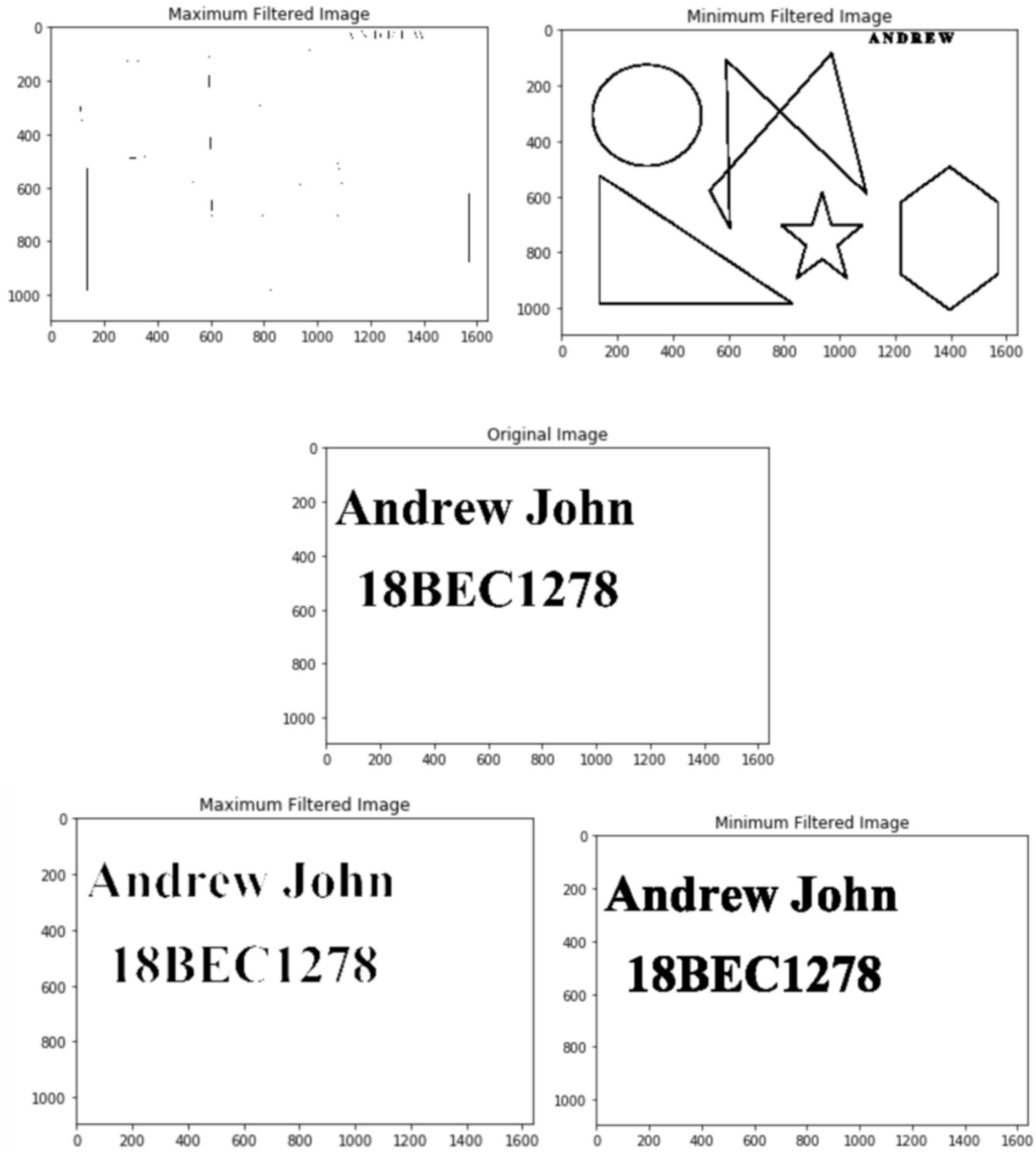


Fig-10: Maximum and Minimum Filters applied to Shapes and Texts

INFERENCE: We can see the clear thinning in Maximum filters meanwhile thickening in Minimum filters as per definition of the filters.

3.3 <u>Sharpening Images - Laplacian Operator on Gaussian Operator</u>

Sharpening as the name suggests is used to sharpen and highlight the edges and make the transitioning of features and details more significant. However sharpening doesn't take into account whether it is highlighting the original features of the image or the noise associated with it. It enhances both.

Blurring vs Sharpening

(a) Blurring : Blurring/smooth is done in spatial domain via taking average of the pixels of its neighbours , thereby producing a blurring effect. It is a process of integration.

(b) *Sharpening* : Sharpening is used to find the difference by the neighborhood and enhance them even more. It is a process of differentiation.

Here Sharpening is done through Laplacian of Gaussian Operator where the image is convolved with the Gaussian smoothing kernel and the resulting smoothened image is convolved with a Laplacian kernel.

In this task, the following was achieved; sharpening of images in both Black and White format and Color format with external noise.

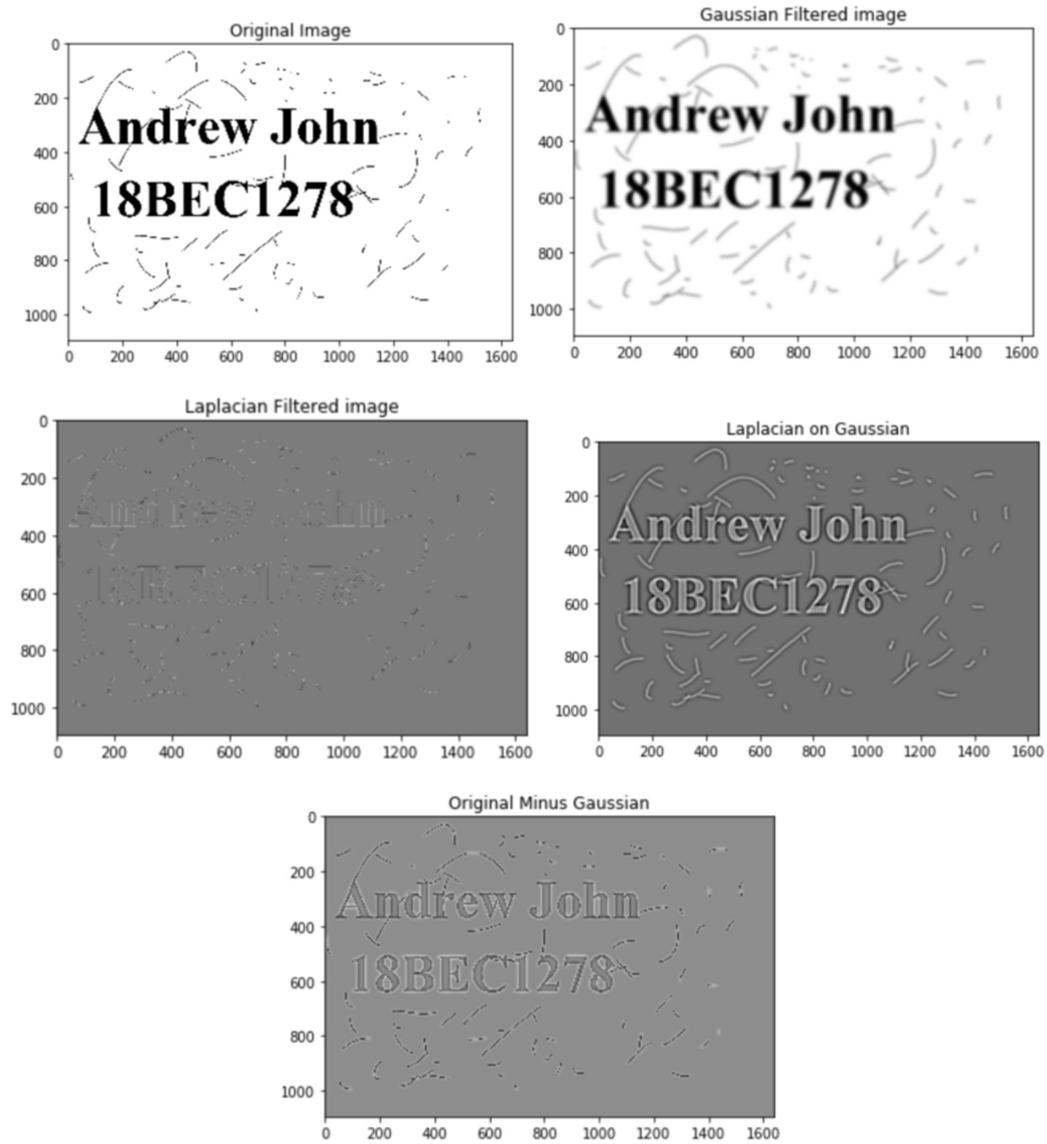


Fig-11: Color Image treatment with External Noise, Sharpened (Sigma=5)

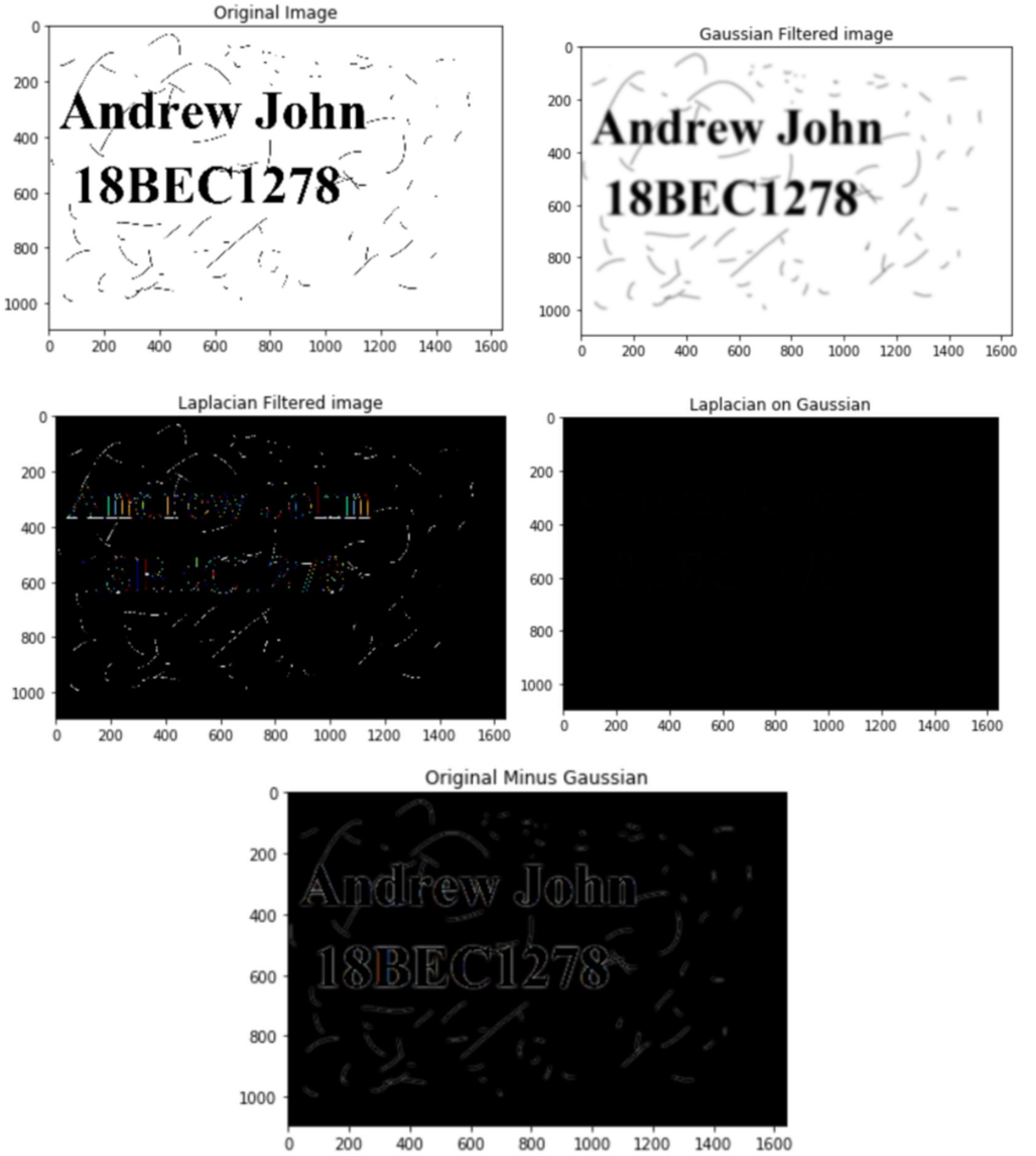


Fig-12: Black & White Image treatment with External Noise, Sharpened (Sigma=5)

3.4 <u>Sobel and Prewitt Operators</u>

Prewitt Operator: Prewitt operator is used for edge detection in an image. It detects two types of edges

- Horizontal Edges
- Vertical Edges

Edges are calculated by using the difference between corresponding pixel intensities of an image. All the masks that are used for edge detection are also known as derivative masks. Prewitt operator provides us two masks one for detecting edges in horizontal direction and another for detecting edges in an vertical direction.

Sobel Operator: The sobel operator is very similar to the Prewitt operator. It is also a derivative mask and is used for edge detection. Like Prewitt operator sobel operator is also used to detect two kinds of edges in an image:

- Vertical direction
- Horizontal direction

The difference between Sobel and Prewitt operators is that in Sobel operators the coefficients of masks are not fixed and they can be adjusted according to our requirement unless they do not violate any property of derivative masks.

In this task, the following was achieved; applying Sobel and Prewitt Operators in both Black and White format and Color format with no external Noise (the same results will be yielded with external noise, where the horizontal and vertical noises will be enhanced horizontally and vertically, respectively)

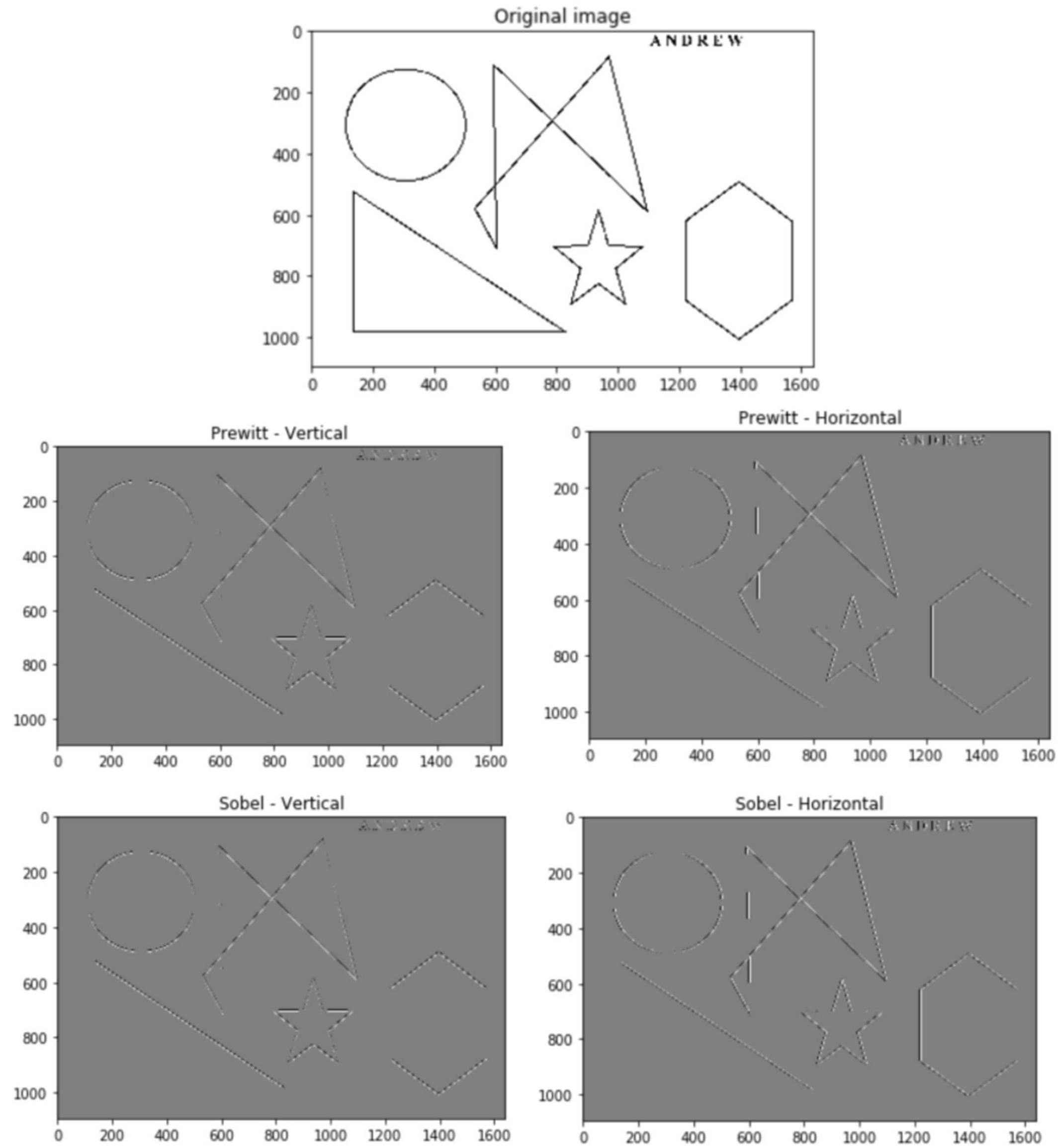


Fig-13: Color Image Treatment with no External Noise

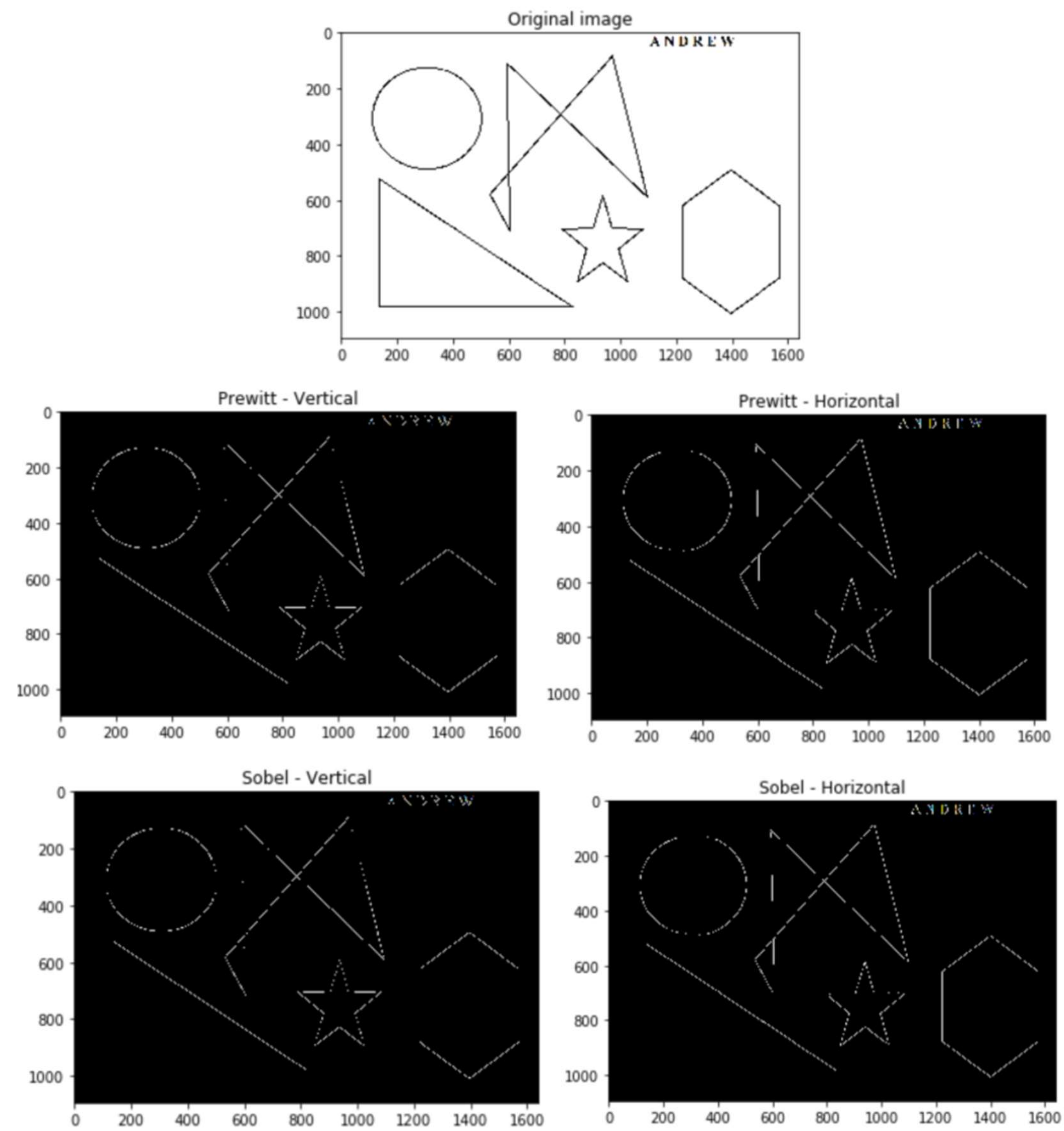


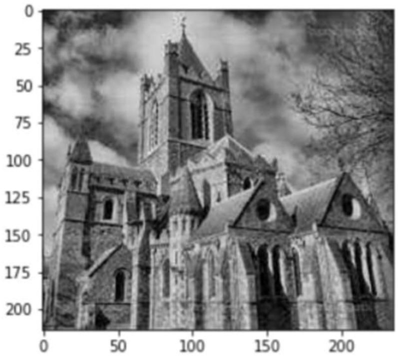
Fig-14: Black & White Image Treatment with no External Noise

3.5 Brightness Function

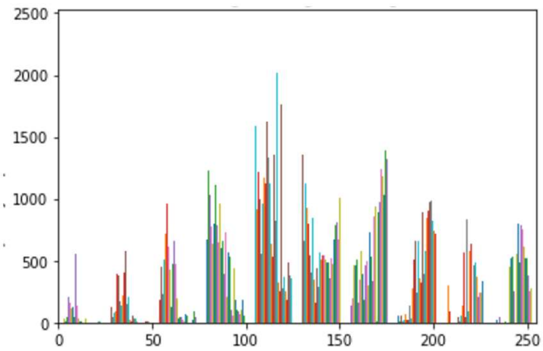
Changing the brightness of an image is a commonly used point operation. In this operation, the value of each and every pixel in an image should be increased/decreased by a constant. To change the brightness of a video, the same operation should be performed on each frame in the video.

If you want to increase the brightness of an image, you have to add some positive constant value to each and every pixel in the image. If you want to decrease the brightness of an image, you have to subtract some positive constant value from each and every pixel in the image.

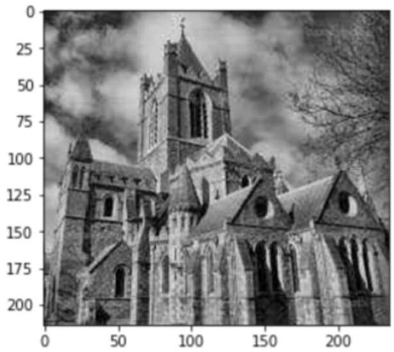
In this task, the following was achieved; Brightness function of an Image was taken and drastic change in histogram of the picture was observed.



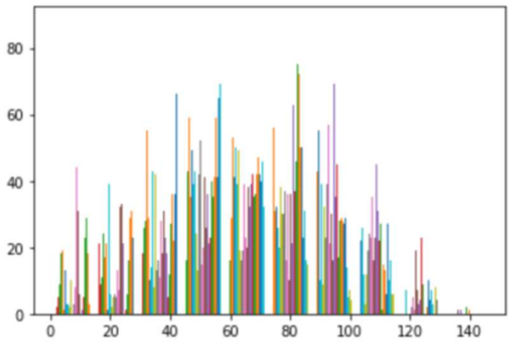
(initial image - a)



(initial graph - a)



(brightness image - b)



(brightness graph - b)

Fig-15: After applying Brightness Function, drastic change is observed in Graph - b

INFERENCE: Here we can observe that , as we add 100 images to the original image, Histogram values have jumped from 140 to 240. Also the image looks a little bit brighter.

3.6 <u>Gamma Transformation</u>

Gamma correction controls the overall brightness of an image. Images which are not properly corrected can look either bleached out, or too dark. Trying to reproduce colors accurately also requires some knowledge of gamma. Varying the amount of gamma correction changes not only the brightness, but also the ratios of red to green to blue.

Power-law (Gamma) transformation: The general form of the power-law transformation is

$$s = cr^{\gamma}$$

where c and γ are positive constants. This transformation is also known as gamma correction. A variety of devices used for image capturing, printing and display respond according to power law. The optimal value for γ is device-dependent.

In this task, the following was achieved; Gamma function was used to the input image where as we increase the value of Gamma we can see the image changes to a bit darker spectrum.

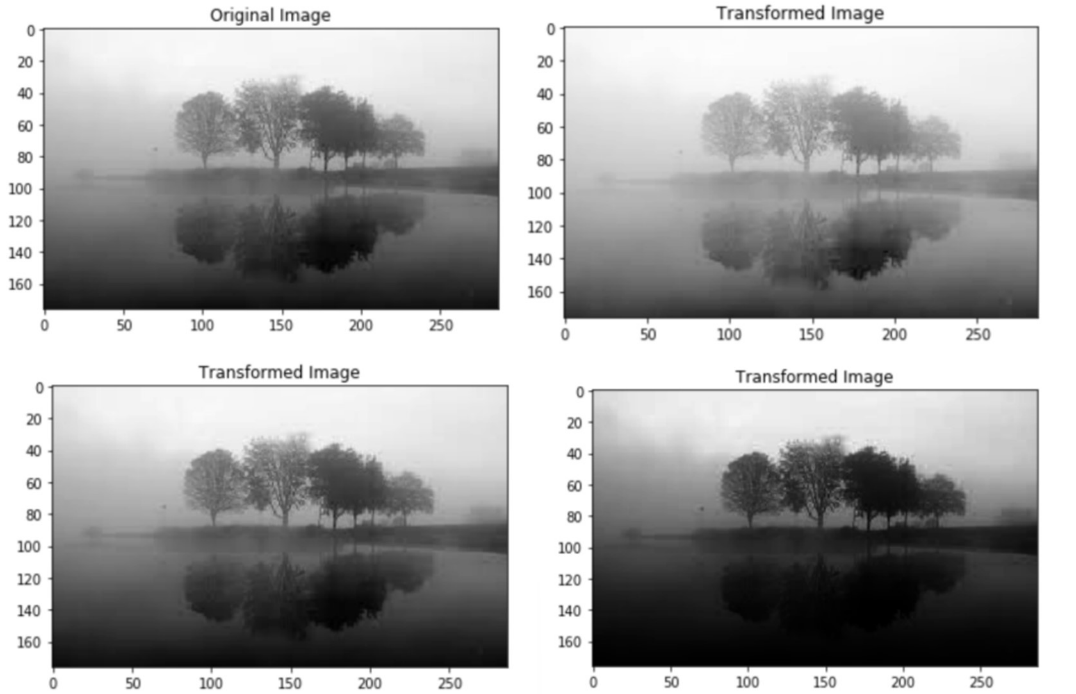


Fig - 16: Gamma Function at different γ values (Original Image, and γ at 0.5, 1.2, 2,2)

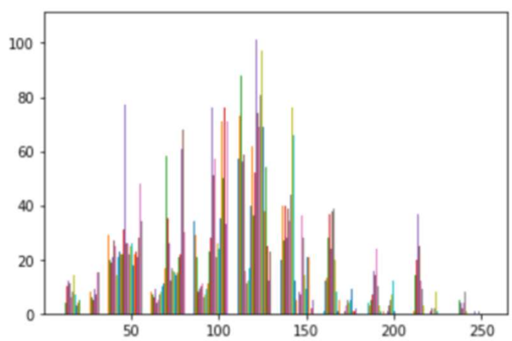
INFERENCE: As the value of gamma increases the less bright/foggy image becomes dark. We can see that the trees are getting darker and darker as we increase the gamma value

3.7 <u>Contrast Stretching</u>

Contrast stretching (often called normalization) is a simple image enhancement technique that attempts to improve the contrast in an image by `stretching' the range of intensity values it contains to span a desired range of values, e.g. the the full range of pixel values that the image type concerned allows. It differs from the more sophisticated histogram equalization in that it can only apply a linear scaling function to the image pixel values. As a result the 'enhancement' is less harsh. (Most implementations accept a gray level image as input and produce another gray level image as output.)

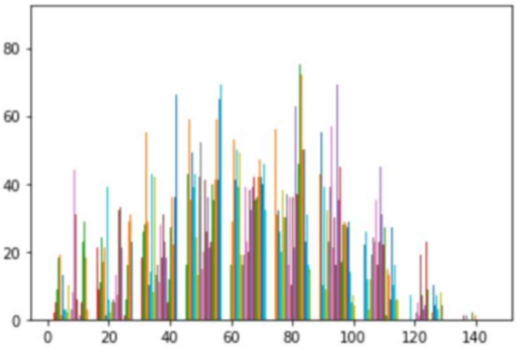
In this task, the following was achieved; Contrast stretching/normalization was performed where input image histogram should have range changes and the image needs to be a bit brighter than the original initial image.

(initial image - a)



(initial graph - a)

(normalized image - b)



(normalized graph - b)

Fig-17: After performing Contrast Stretching, range change is observed in Graph - b

INFERENCE: In the above histograms we can observe a range of gaps have been stretched and also the background is more clear than the first pic.

3.8 Log Transformation

Logarithmic transformation of an image is one of the gray level image transformations. Log transformation of an image means replacing all pixel values, present in the image, with its logarithmic values. Log transformation is used for image enhancement as it expands dark pixels of the image as compared to higher pixel values. The general form of log transformation function is

$$s = T(r) = c*\log(1+r)$$

where, 's' and 'r' are the output and input pixel values and c is the scaling constant represented by the following expression (for 8-bit)

$$c = 255/(\log(1 + max\_input\_pixel\_value))$$

The value of c is chosen such that we get the maximum output value corresponding to the bit size used. e.g for 8 bit image, c is chosen such that we get max value equal to 255.

In this task, the following was achieved; when the log transformation was applied, the histogram value of the original image must be flipped and needs to be on the higher side when the initial is on the lower side and vice-versa.

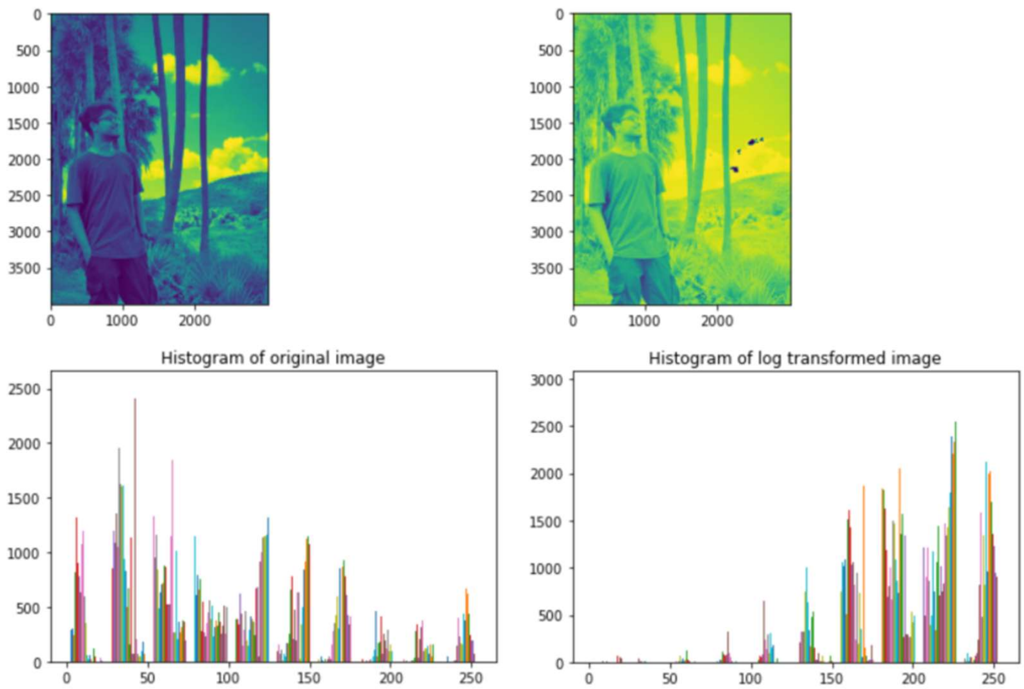


Fig-18: After applying Log transformation, values have been flipped

INFERENCE: The values have been flipped and the image has been transformed when the log function was applied.

CHAPTER 4

Image Segmentation and Image Morphology using Python

4.1 Canny Edge Detection on the Captured Images(s)

The Canny edge detector is an edge detection operator that uses a multi-stage algorithm to detect a wide range of edges in images. The Canny filter is a multi-stage edge detector. It uses a filter based on the derivative of a Gaussian in order to compute the intensity of the gradients.The Gaussian reduces the effect of noise present in the image. Then, potential edges are thinned down to 1-pixel curves by removing non-maximum pixels of the gradient magnitude. Finally, edge pixels are kept or removed using hysteresis thresholding on the gradient magnitude.

The Canny has three adjustable parameters: the width of the Gaussian (the noisier the image, the greater the width), and the low and high threshold for the hysteresis thresholding.

In this task, the following was achieved; Canny Edge Detection of input images was detected in different RGB Border colors.

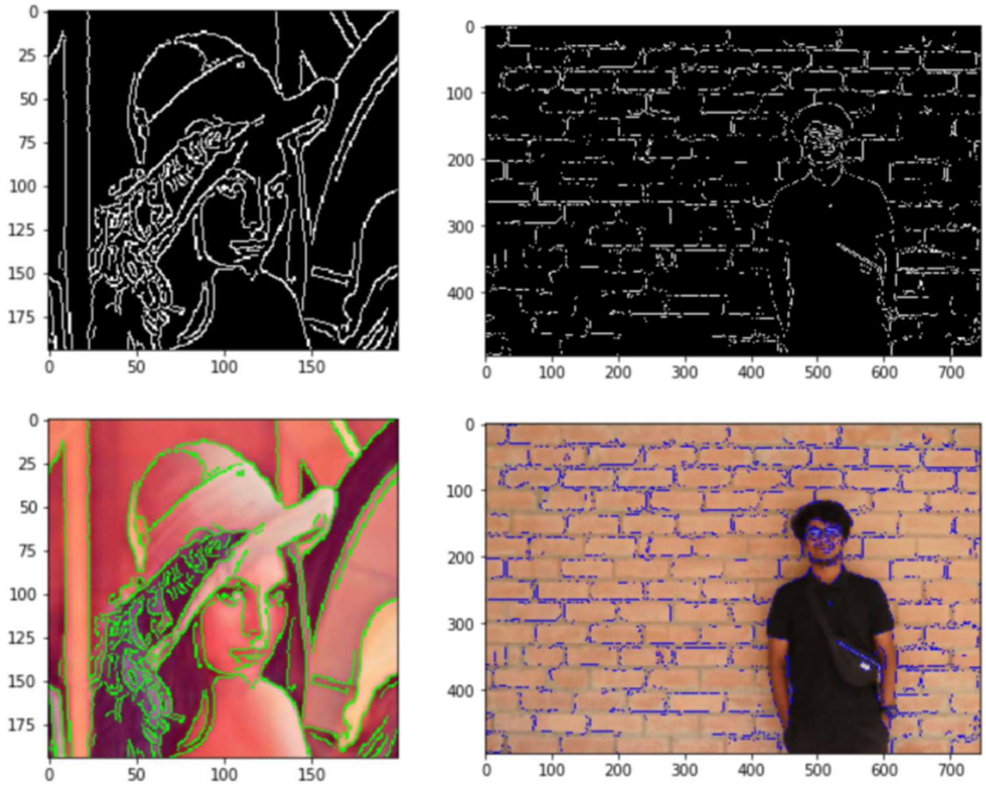


Fig-19: Canny Edge Detected Images

The general criteria for edge detection include:

- Detection of edge with low error rate, which means that the detection should accurately catch as many edges shown in the image as possible

- The edge point detected from the operator should accurately localize on the center of the edge.
- A given edge in the image should only be marked once, and where possible, image noise should not create false edges.

INFERENCE: Threshold value is image dependent. As you increase the upper threshold value more the image is smoothened .

4.2 Harris Corner Detection on the Captured Images(s)

The Harris corner detector is a corner detection operator that is commonly used in computer vision algorithms to extract corners and infer features of an image. Harris' corner detector takes the differential of the corner score into account with reference to direction directly, instead of using shifting patches for every 45 degree angle, and has been proved to be more accurate in distinguishing between edges and corners.

Commonly, Harris corner detector algorithm can be divided into five steps.

1. Color to grayscale
2. Spatial derivative calculation
3. Structure tensor setup
4. Harris response calculation
5. Non-maximum suppression

In this task, the following was achieved; Corners was detected when the Harris Corner function was applied along pictures that had curvature as well as linear pictures.

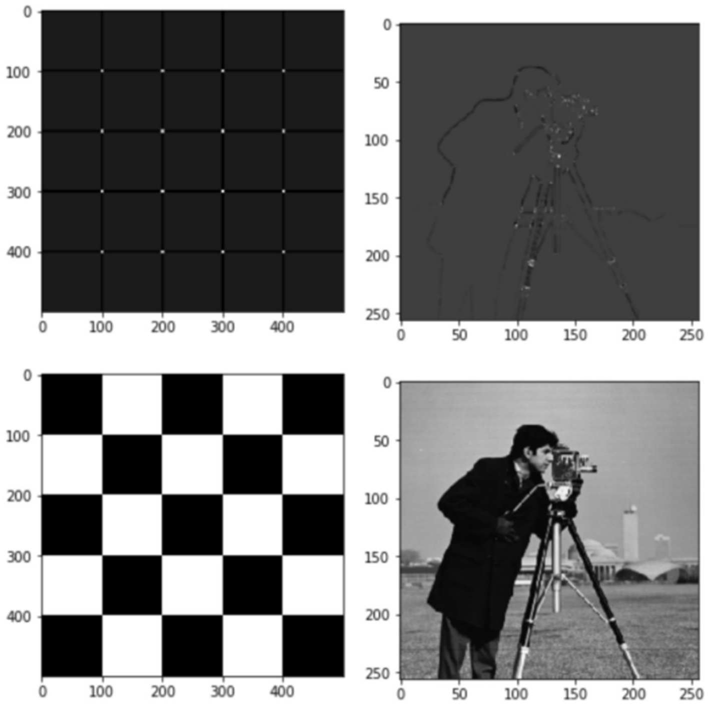


Fig-20: Harris Corner Detected Images

INFERENCE: When we upload the picture , through Harris corner detection it detects all the edges of the image and finally the image is converted into RGB image where only red, blue ,green colors of the image are shown

4.3 Hough Line Detection on Task - 1's Edge Detected Image(s)

The Hough transform is a feature extraction technique used in image analysis, computer vision, and digital image processing. The purpose of the technique is to find imperfect instances of objects within a certain class of shapes by a voting procedure.

This voting procedure is carried out in a parameter space, from which object candidates are obtained as local maxima in a so-called accumulator space that is explicitly constructed by the algorithm for computing the Hough transform. The simplest case of Hough transform is detecting straight lines. In general, the straight line y = mx + b can be represented as a point (b, m) in the parameter space.

In this task the following was achieved; On the edge detected images from Canny Detection, Hough transform was applied and lines along the image were highlighted in RGB Colors.

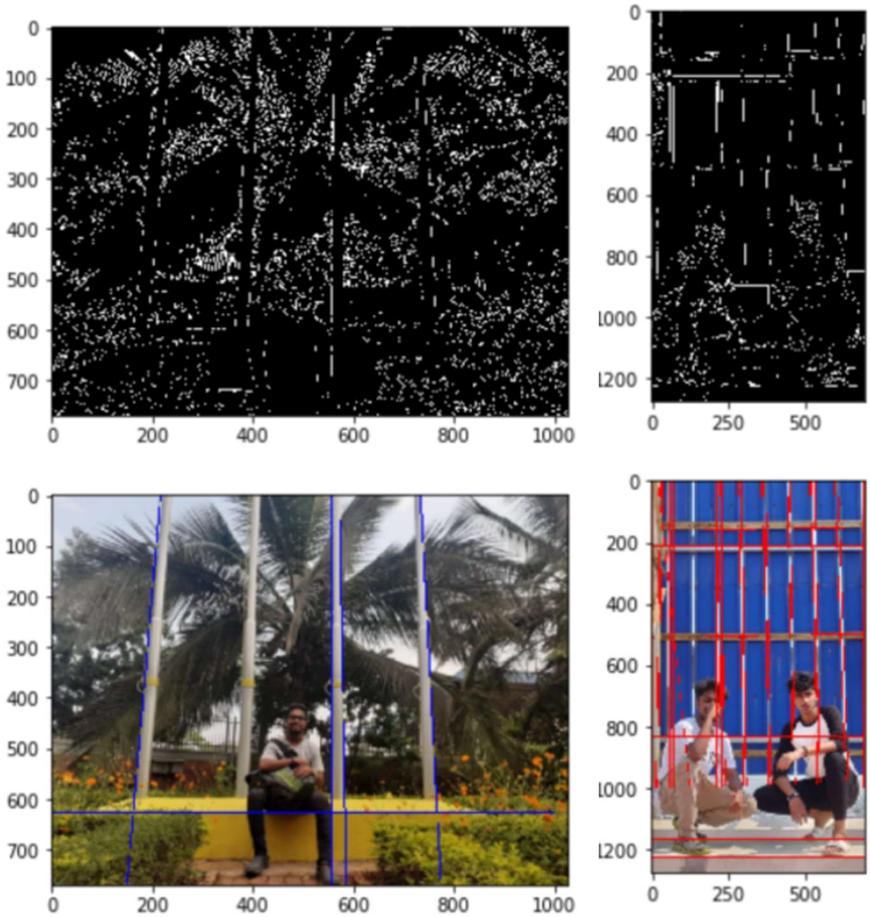


Fig-21: Hough Line Transformed Images

INFERENCE : As we increase the value of 'd' the image gets more blurred. This makes it tough to detect the edges . As a result the red lines in the image get reduced

whereas in the image with ‘d’ value as 2 , the image is more clear and hence it detects more straight lines as edges and there are more red lines in the output.
Hence we have applied the HOUGH transform and detected the lines

4.4 Hough Circle Detection on the Captured Images(s)

The circle Hough Transform (CHT) is a basic feature extraction technique used in digital image processing for detecting circles in imperfect images. The circle candidates are produced by “voting” in the Hough parameter space and then selecting local maxima in an accumulator matrix.

In a two-dimensional space, a circle can be described by:

$$(x-a)^2 + (y-b)^2 = r^2$$

where (a,b) is the center of the circle, and r is the radius. If a 2D point (x,y) is fixed, then the parameters can be found according to the above equation. The parameter space would be three dimensional, (a, b, r). And all the parameters that satisfy (x, y) would lie on the surface of an inverted right-angled cone whose apex is at (x, y, 0).

In this task, the following was achieved; circles were detected similarly to how line was detected in the above Hough transformation. Both real life images and paint images were given as input. The radius of each circle was also detected.

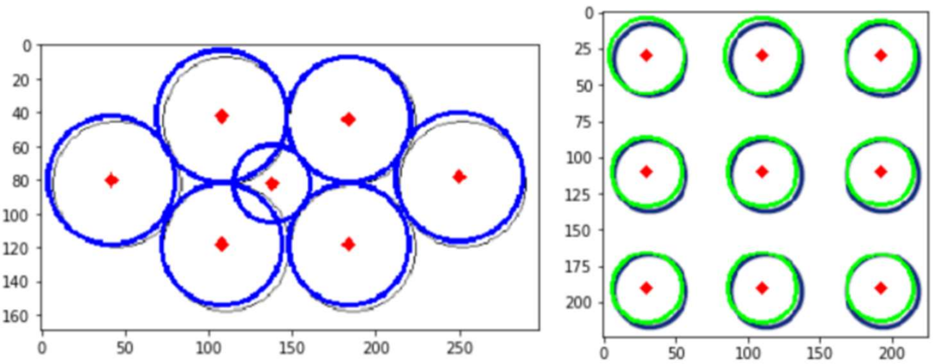


Fig-22: Hough Circle Transformed Images

4.5 k-Means Clustering

k-Means clustering is a method of vector quantization, originally from signal processing, that aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean (cluster centers or cluster centroid), serving as a prototype of the cluster.

Here k-Means is used for Image Segmentation, where Image segmentation is the process of partitioning an image into multiple different regions (or segments). The goal is to change the representation of the image into an easier and more meaningful image. Here k-Means aims to partition N observations into K clusters in which each observation belongs to the cluster with the nearest mean. A cluster refers to a collection of data points aggregated together because of certain similarities. For image segmentation, clusters here are different image colors.

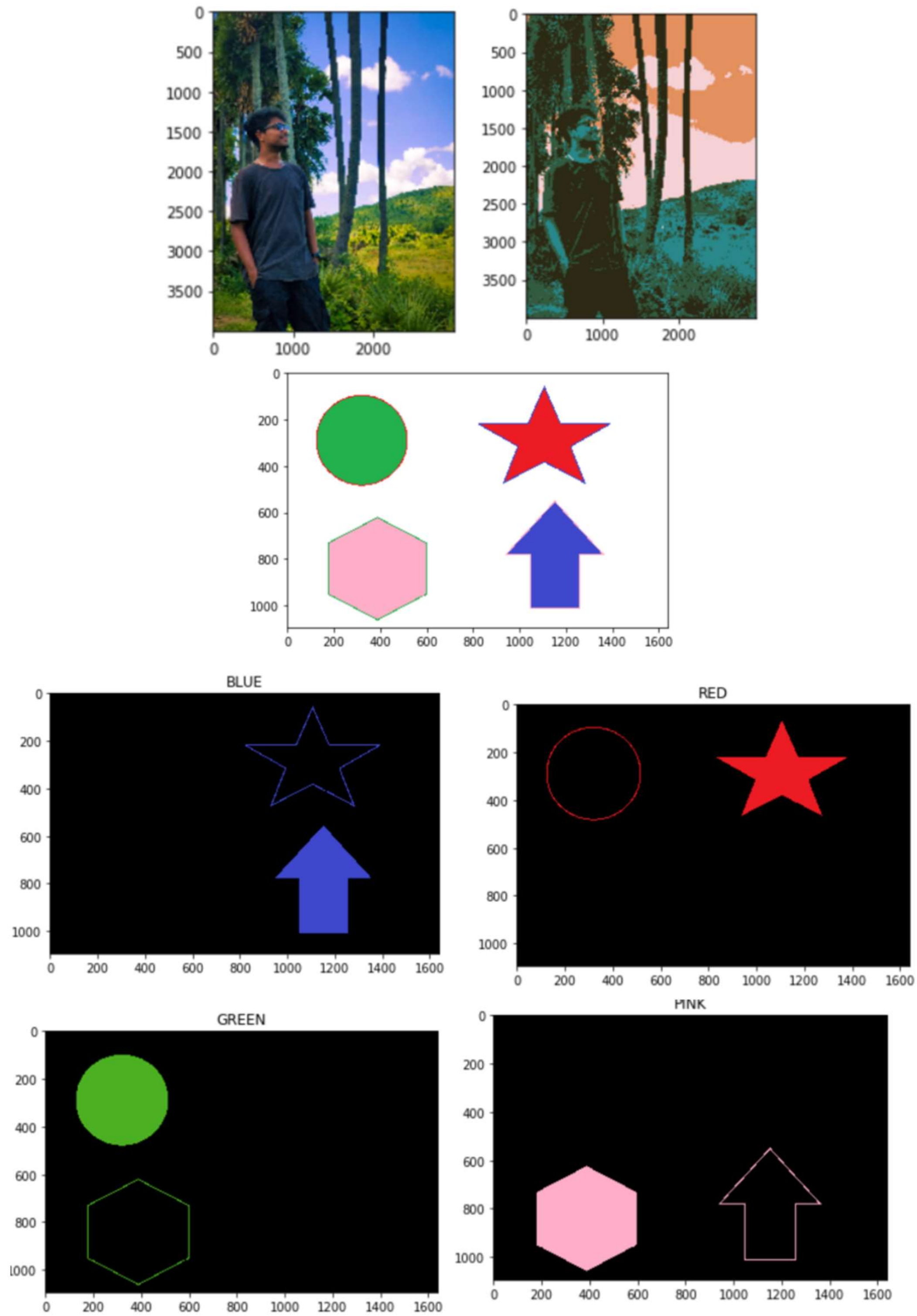


Fig-23: k-Means clustering used for Image Segmentation and Color Segmentation

INFERENCE: When we upload images we note that only the red , blue ,green color portions of the image are highlighted since we cluster the RGB Colors into a segment. Hence we have performed k-means clustering on images and observed the output.

4.6 Morphological Operations

Morphological Operations is a broad set of image processing operations that process digital images based on their shapes. In a morphological operation, each image pixel is corresponding to the value of another pixel in its neighborhood. By choosing the shape and size of the neighborhood pixel, you can construct a morphological operation that is sensitive to specific shapes in the input image.

Types of Morphological operations:

- *Dilation*: Dilation adds pixels on the object boundaries.
- *Erosion*: Erosion removes pixels on object boundaries.
- *Open*: The opening operation erodes an image and then dilates the eroded image, using the same structuring element for both operations.
- *Close*: The closing operation dilates an image and then erodes the dilated image, using the same structuring element for both operations.

In this task, the following was achieved; Dilation, Opening, Erosion, and Closing was performed on the input image.

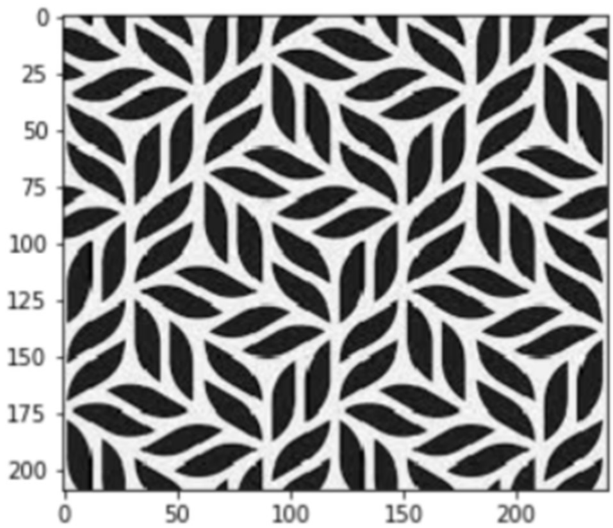


Fig-24: Input Image for Morphological Operations

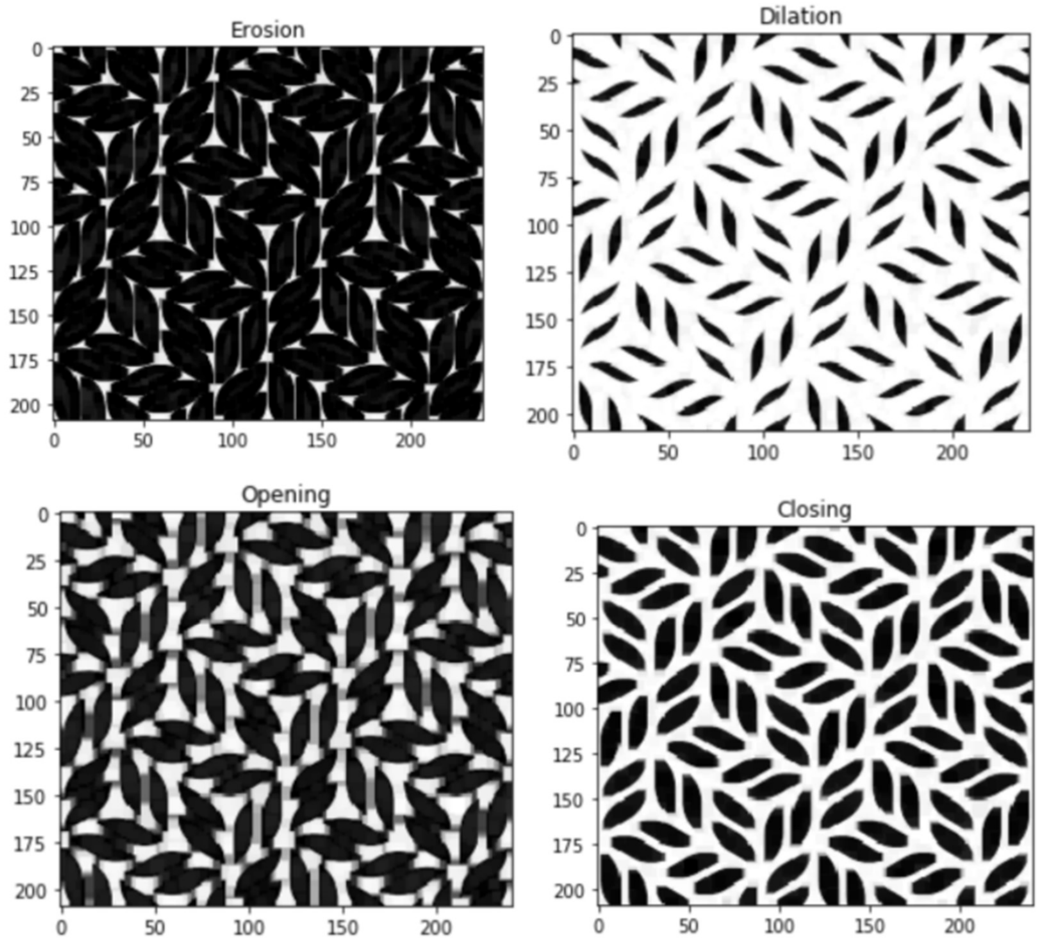


Fig-25: Image after Morphological Operations

INFERENCE: Erosion has thickened (removed pixels) each petal boundaries while Dilation (added pixels) has thinned the boundaries of the petals, Opening has increased the pixel size of each element in the image whereas Closing has refined each element in the image.

CHAPTER 5

Deep Learning and CNN for Vision Applications

Concepts of Neural Network, Deep Learning, CNN & Transfer Learning was discussed by the Guest Lecturer

Neural networks are parallel computing devices, which is basically an attempt to make a computer model of the brain. The main objective is to develop a system to perform various computational tasks faster than the traditional systems. These tasks include pattern recognition and classification, approximation, optimization, and data clustering.

Artificial Neural Network is an efficient computing system whose central theme is borrowed from the analogy of biological neural networks. ANNs are also named as "artificial neural systems," or "parallel distributed processing systems," or "connectionist systems." ANN acquires a large collection of units that are interconnected in some pattern to allow communication between the units. These units, also referred to as nodes or neurons, are simple processors which operate in parallel.

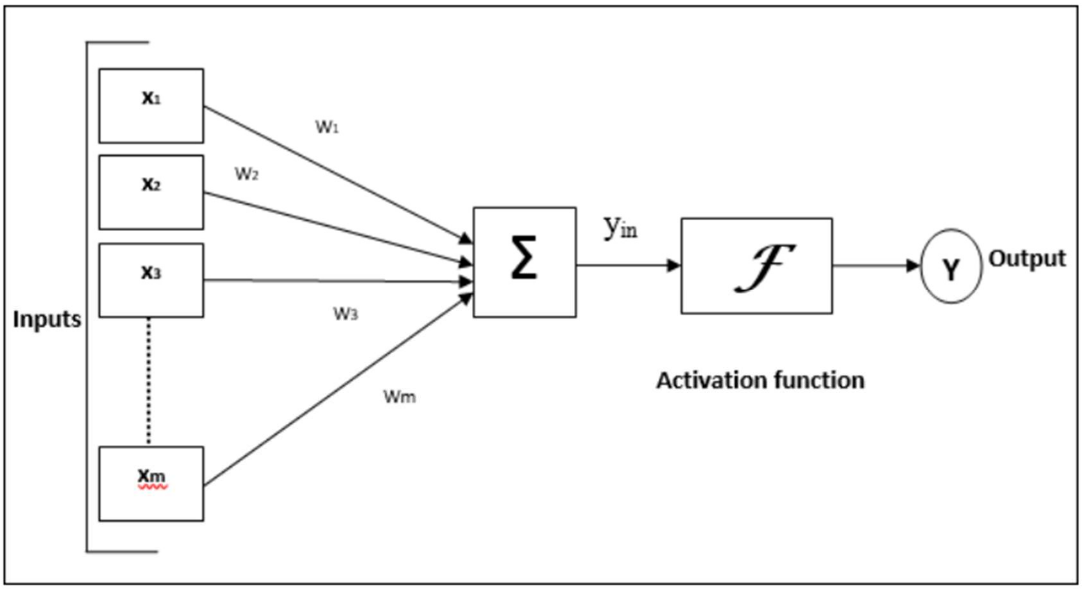


Fig-26: Model of an Artificial Neural Network

Deep structured learning or hierarchical learning or deep learning in short is part of the family of machine learning methods which are themselves a subset of the broader field of Artificial Intelligence. Deep learning is a class of machine learning algorithms that use several layers of nonlinear processing units for feature extraction and transformation. Each successive layer uses the output from the previous layer as input.

Deep neural networks, deep belief networks and recurrent neural networks have been applied to fields such as computer vision, speech recognition, natural language processing, audio recognition, social network filtering, machine translation, and bioinformatics where they produced results comparable to and in some cases better than human experts have.

Convolutional Neural networks are designed to process data through multiple layers of arrays. This type of neural networks is used in applications like image recognition or face recognition. The primary difference between CNN and any other ordinary neural network is that CNN takes input as a two-dimensional array and operates directly on the images rather than focusing on feature extraction which other neural networks focus on.

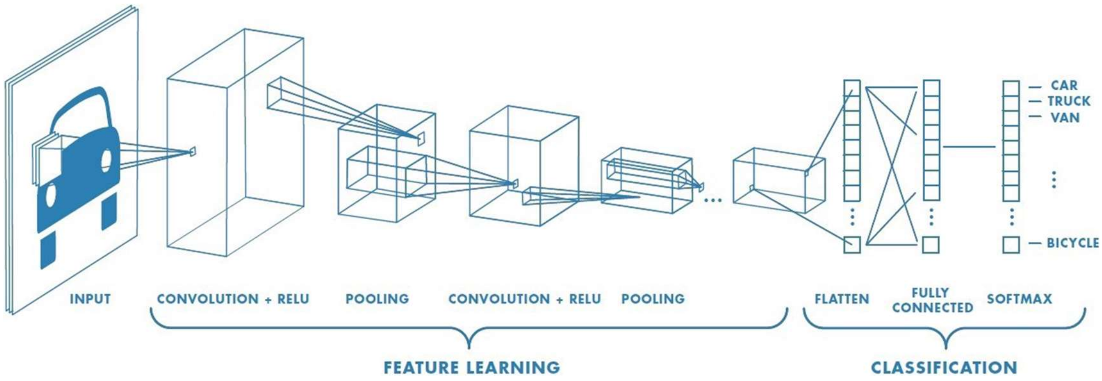


Fig-27: Convolutional Neural Network

Transfer learning (TL) is a research problem in machine learning (ML) that focuses on storing knowledge gained while solving one problem and applying it to a different but related problem. This area of research bears some relation to the long history of psychological literature on transfer of learning, although practical ties between the two fields are limited. From the practical standpoint, reusing or transferring information from previously learned tasks for the learning of new tasks has the potential to significantly improve the sample efficiency of a reinforcement learning agent.

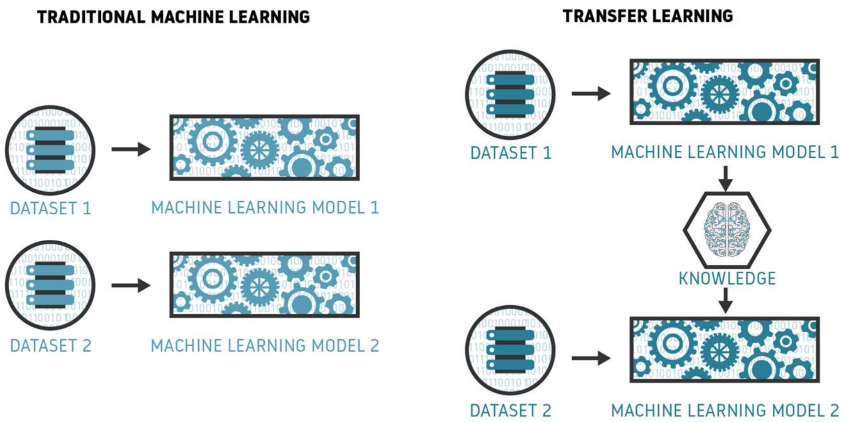


Fig-28: Transfer Learning and Traditional ML Difference.

CHAPTER 6

State-of-the-art Computer Vision Applications

<u>Computer Vision in Manufacturing-</u>

- Productivity Analytics:
    - Productivity analytics track the impact of workplace change, how employees spend their time and resources, and implement various tools. Such data can provide valuable insight into time management, workplace collaboration, and employee productivity. Computer Vision lean management strategies aim to objectively quantify and assess processes with cameras-based vision systems.

- Quality Management:
    - Smart camera applications provide a scalable method to implement automated visual inspection and quality control of production processes and assembly lines in smart factories. Hereby, deep learning uses real-time object detection to provide superior results (detection accuracy, speed, objectiveness, reliability) compared to laborious manual inspection.

<u>Computer Vision in Healthcare-</u>

- Cancer Detection:
    - Machine learning is incorporated in medical industries for purposes such as breast and skin cancer detection. For instance, image recognition allows scientists to detect slight differences between cancerous and non-cancerous images and diagnose data from magnetic resonance imaging (MRI) scans and inputted photos as malignant or benign.
- COVID-19 diagnosis:
    - Computer Vision can be used for coronavirus control. Multiple deep learning computer vision models exist for x-ray based COVID-19 diagnosis. The most popular one for detecting COVID-19 cases with digital chest x-ray radiography (CXR) images is named COVID-Net and was developed by Darwin AI, Canada.

<u>Computer Vision in Agriculture-</u>

- Farm Automation:
    - Technologies such as harvest, seeding, and weeding robots, autonomous tractors, and vision systems to monitor remote farms, drones for visual inspection can maximize productivity with labor shortages. The profitability can be significantly increased by automating manual inspection with AI vision, reducing the ecological footprint, and improving decision-making processes.

- Yield Assessment:
  - Through the application of computer vision technology, the functions of soil management, maturity detection, and yield estimation for farms have been realized. Moreover, the existing technology can be well applied to methods such as spectral analysis and deep learning.
  - Most of these methods have the advantages of high precision, low cost, good portability, good integration, and scalability and can provide reliable support for management decision making. An example is the estimation of citrus crop yield via fruit detection and counting using computer vision.

<u>Computer Vision in Transportation-</u>

- Moving Violations Detection:
  - Law enforcement agencies and municipalities are increasing the deployment of camera-based roadway monitoring systems with the goal of reducing unsafe driving behavior. Probably the most critical application is the detection of stopped vehicles in dangerous areas.
  - Also, there is increasing use of computer vision techniques in smart cities that involve automating the detection of violations such as speeding, running red lights or stop signs, wrong-way driving, and making illegal turns.

- Traffic Flow Analysis:
  - Traffic flow analysis has been studied extensively for intelligent transportation systems (ITS) using invasive methods (tags, under-pavement coils, etc.) and non-invasive methods such as cameras.
  - With the rise of computer vision and AI, video analytics can now be applied to the ubiquitous traffic cameras, which can generate a vast impact in ITS and smart cities. The traffic flow can be observed using computer vision means and measure some of the variables required by traffic engineers.

<u>Computer Vision in Retail-</u>

- Customer Tracking:
  - Deep learning algorithms can process the video streams in real-time to analyze the customer footfall in retail stores. Camera-based methods allow re-using the video stream of common, inexpensive security surveillance cameras. Machine learning algorithms detect people anonymously and contactless to analyze time spent in different areas, waiting times, queueing time, and assess the service quality.

- Theft Detection:
  - Retailers can detect suspicious behavior such as loitering or accessing areas that are off-limits using computer vision algorithms that are autonomously analyzing the scene.

- Social Distancing:
    - To ensure safety precautions are being followed, companies are using distance detectors. A camera tracks employee or customer movement and uses depth sensors to assess the distance between them. Then, depending on their position, the system draws a red or green circle around the person. Learn more about Social Distancing Monitoring with deep learning.

<u>Computer Vision in Sports-</u>

- Player Pose Tracking-
    - AI vision can recognize patterns between human body movement and pose over multiple frames in video footage or real-time video streams. For example, human pose estimation has been applied to real-world videos of swimmers where single stationary cameras film above and below the water surface. Those video recordings can be used to quantitatively assess the athletes' performance without manually annotating the body parts in each video frame. Thus, Convolutional Neural Networks are used to automatically infer the required pose information and detect the swimming style of an athlete.
- Goal-Line Technology
    - Camera-based systems can be used to determine if a goal has been scored or not to support the decision-making of referees. Unlike sensors, the AI vision-based method is noninvasive and does not require changes to the typical football devices.
    - Such Goal-Line Technology systems are based on high-speed cameras whose images are used to triangulate the ball's position. A ball detection algorithm that analyzes candidate ball regions in order to recognize the ball pattern.

CHAPTER 7

Conclusion

At the end of the course we were able to : Identify appropriate computer vision techniques for various real-time projects, apply computer vision techniques for better image understanding, implement and debug python codes related to image enhancement, understand and implement object detection techniques in python, apply face detection algorithm in python and studied the importance of computer vision in real time world by understanding their applications in various fields.

REFERENCES

[1] https://chennai.vit.ac.in/

[2] https://chennai.vit.ac.in/events/

[3] https://www.wikipedia.org/

[4] https://en.wikipedia.org/wiki/Computer_vision

[5] https://www.tutorialspoint.com/index.htm

[6] https://towardsdatascience.com/

[7] https://opencv.org/

[8] https://www.python.org/

[9] https://jupyter.org/

[10] https://viso.ai/applications/computer-vision-applications/

[11] Piyush Singh, Abhishek Verma, John Sahaya Rani Alex, Disease and pest infection detection in coconut tree through deep learning techniques, Computers and Electronics in Agriculture, Volume 182, 2021, 105986, ISSN 0168-1699, https://doi.org/10.1016/j.compag.2021.105986

[12] A. Anand, M. A. Haque, J. S. R. Alex and N. Venkatesan, "Evaluation of Machine learning and Deep learning algorithms combined with dimentionality reduction techniques for classification of Parkinson's Disease," 2018 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), 2018, pp. 342-347, doi: 10.1109/ISSPIT.2018.8642776.

APPENDIX

Codes from the Value Added Program