DS JULY 2022 Batch
Module 20

Topics

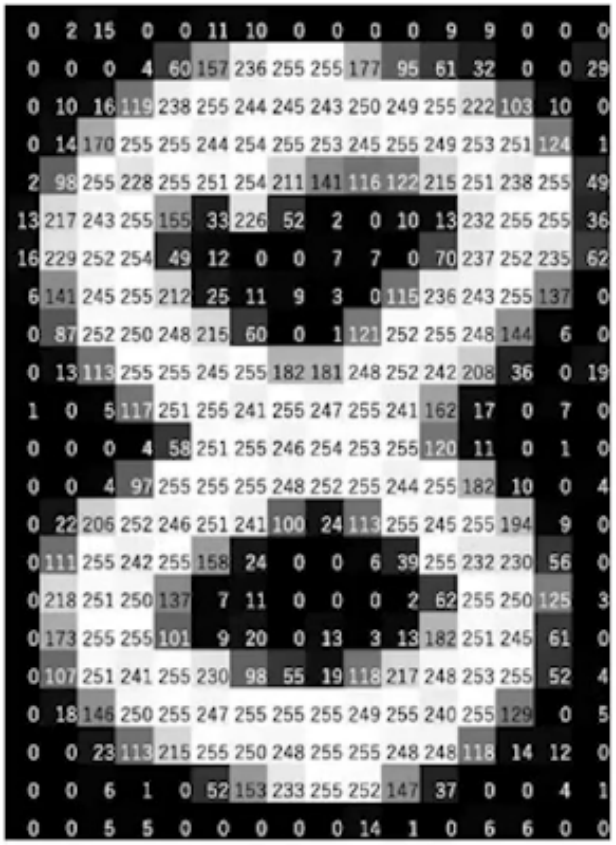
- Images as matrix
- Histogram of images
- Basic filters applied on the images

Image

Image

- **What is an image** : An image is an array, or a matrix, of square pixels (picture elements) arranged in columns and rows.
- An image is represented by its dimensions (height and width) based on the number of pixels. For example, if the dimensions of an image are 500 x 400 (width x height), the total number of pixels in the image is 200000.
- This pixel is a point on the image that takes on a specific shade, opacity or color. It is usually represented in one of the following:
- Grayscale - A pixel is an integer with a value between 0 to 255 (0 is completely black and 255 is completely white).
- RGB - A pixel is made up of 3 integers between 0 to 255 (the integers represent the intensity of red, green, and blue).

Image



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	119	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	98	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	113	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	158	24	0	0	6	39	255	232	230	56	0	
0	218	251	250	137	7	11	0	0	0	2	62	255	250	125	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	146	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

- **What Is Image Processing?**
- Image processing is the process of transforming an image into a digital form and performing certain operations to get some useful information from it. The image processing system usually treats all images as 2D signals when applying certain predetermined signal processing methods.
- **Types of Image Processing**
- There are five main types of image processing:
  - Visualization - Find objects that are not visible in the image
  - Recognition - Distinguish or detect objects in the image
  - Sharpening and restoration - Create an enhanced image from the original image
  - Pattern recognition - Measure the various patterns around the objects in the image
  - Retrieval - Browse and search images from a large database of digital images that are similar to the original image

Image : Fundamental Image Processing Steps

- **Image Acquisition :** Image acquisition is the first step in image processing. This step is also known as pre-processing in image processing. It involves retrieving the image from a source, usually a hardware-based source.
- **Image Enhancement :** Image enhancement is the process of bringing out and highlighting certain features of interest in an image that has been obscured. This can involve changing the brightness, contrast, etc.
- **Image Restoration :** Image restoration is the process of improving the appearance of an image. However, unlike image enhancement, image restoration is done using certain mathematical or probabilistic models.
- **Color Image Processing :** Color image processing includes a number of color modeling techniques in a digital domain. This step has gained prominence due to the significant use of digital images over the internet.
- **Compression :** Compression is a process used to reduce the storage required to save an image or the bandwidth required to transmit it. This is done particularly when the image is for use on the Internet.
- **Morphological Processing :** Morphological processing is a set of processing operations for morphing images based on their shapes.
- **Segmentation :** Segmentation is one of the most difficult steps of image processing. It involves partitioning an image into its constituent parts or objects.

Image : Color image

- In a colored image the number of matrices or the number of channels will be more. In this particular example, we have 3 matrices- **Red, Green, and Blue.**

Colour Image

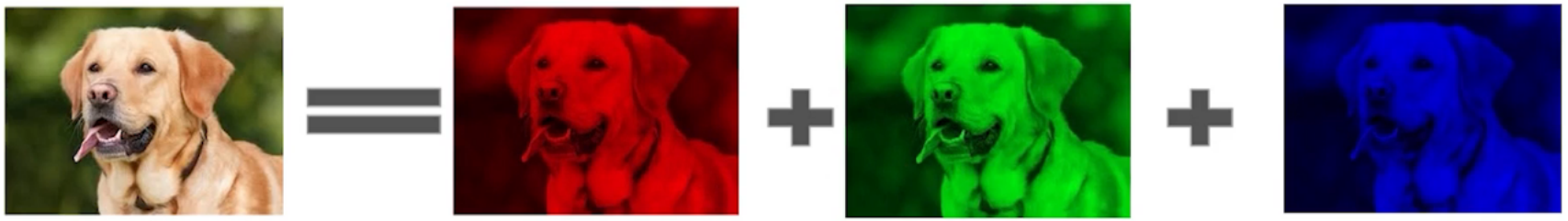
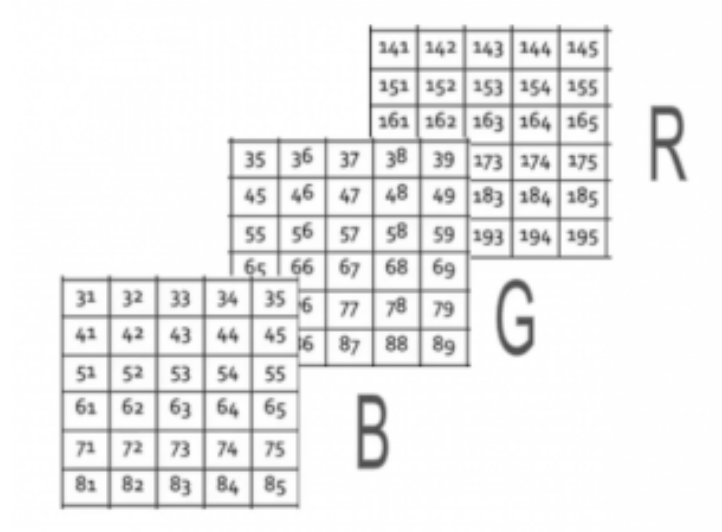


Image : Color image

- In a colored image the number of matrices or the number of channels will be more. In this particular example, we have 3 matrices- **Red, Green, and Blue.**
- Each of these metrics would again have values ranging from 0 to 255 where each of these numbers represents the intensity of the pixels or you can say that the shades of red, green, and blue. Finally, all of these channels or all of these matrices are superimposed so the shape of the image, when loaded in a computer, will be-



Histogram

Image : Histogram

- To determine the histogram of an image, we need to count how many instances of each intensity we have.
- So, a histogram will allow us to see how often each intensity occurs. In our example, the intensity 150 can be seen in three pixels, for this reason, it will have a higher frequency in the histogram (the corresponding bar's height is 3):

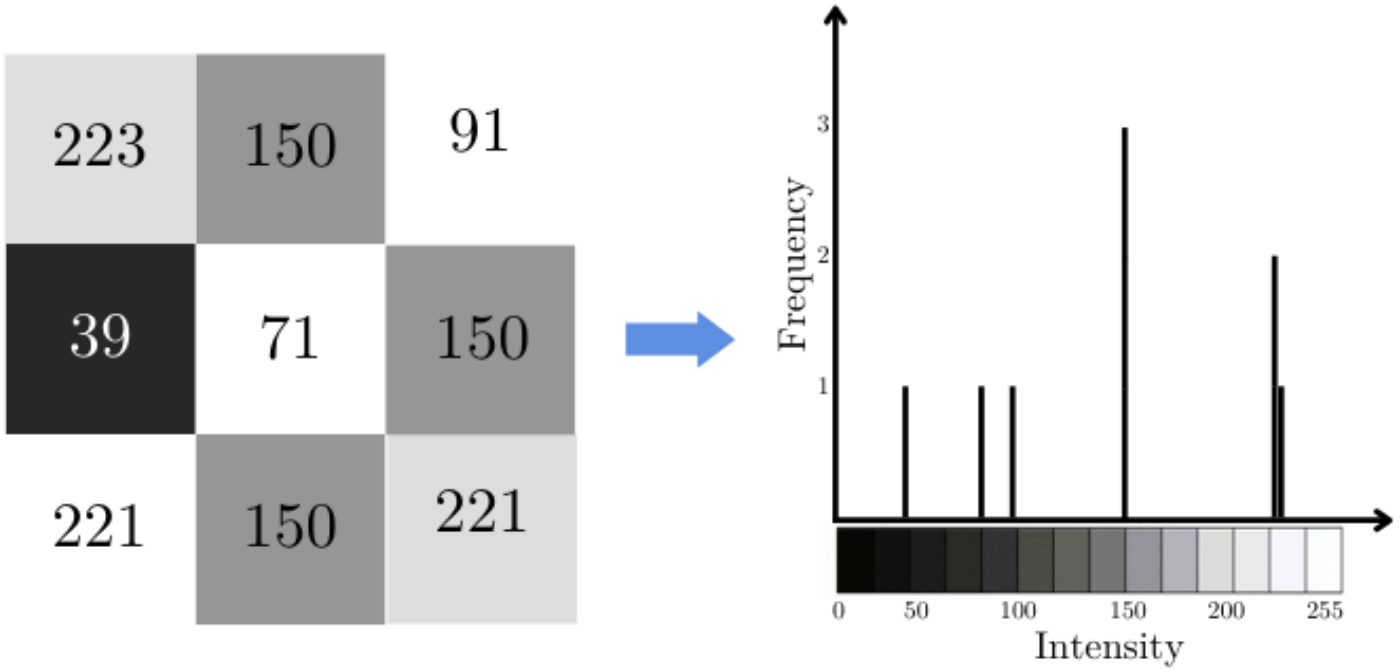
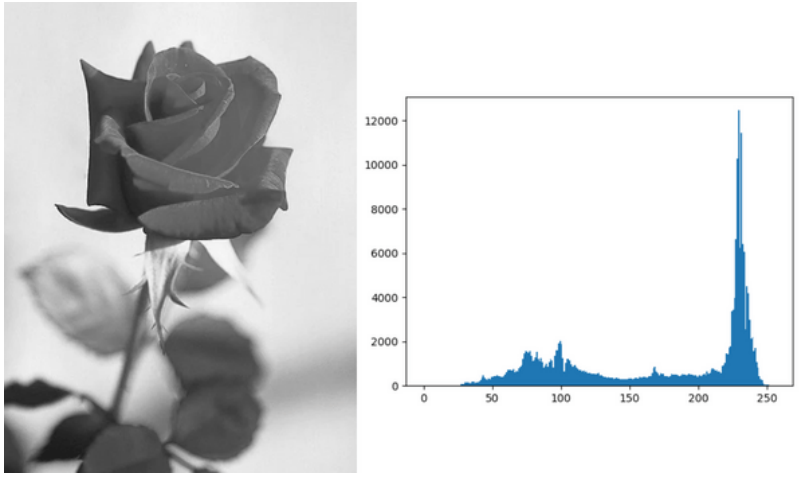
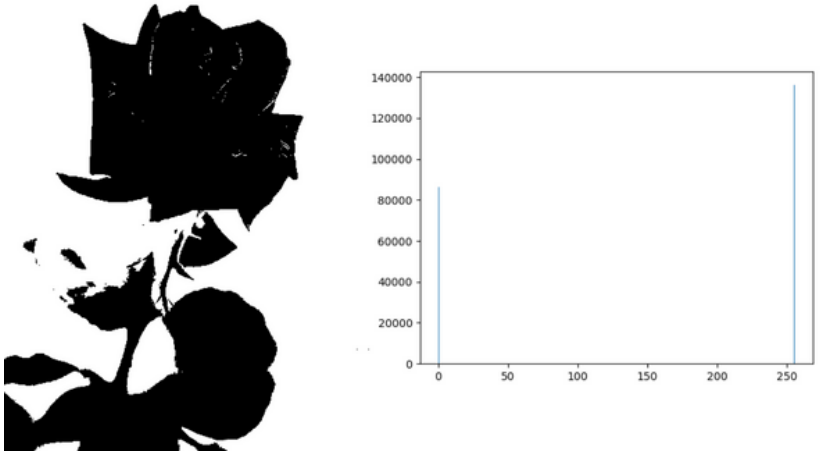


Image : Histogram : applications

- We can use histograms to define the threshold for image segmentation to isolate the background from an object.





- If we define our threshold as 156, and take every pixel > 156 to belong to the background, we'll get a binary image in which the rose is clearly separated:

Image : Histogram : applications

- In photography, we use histograms to enhance pictures by changing some of their properties. This might help us get clearer pictures or even more beautiful photos.

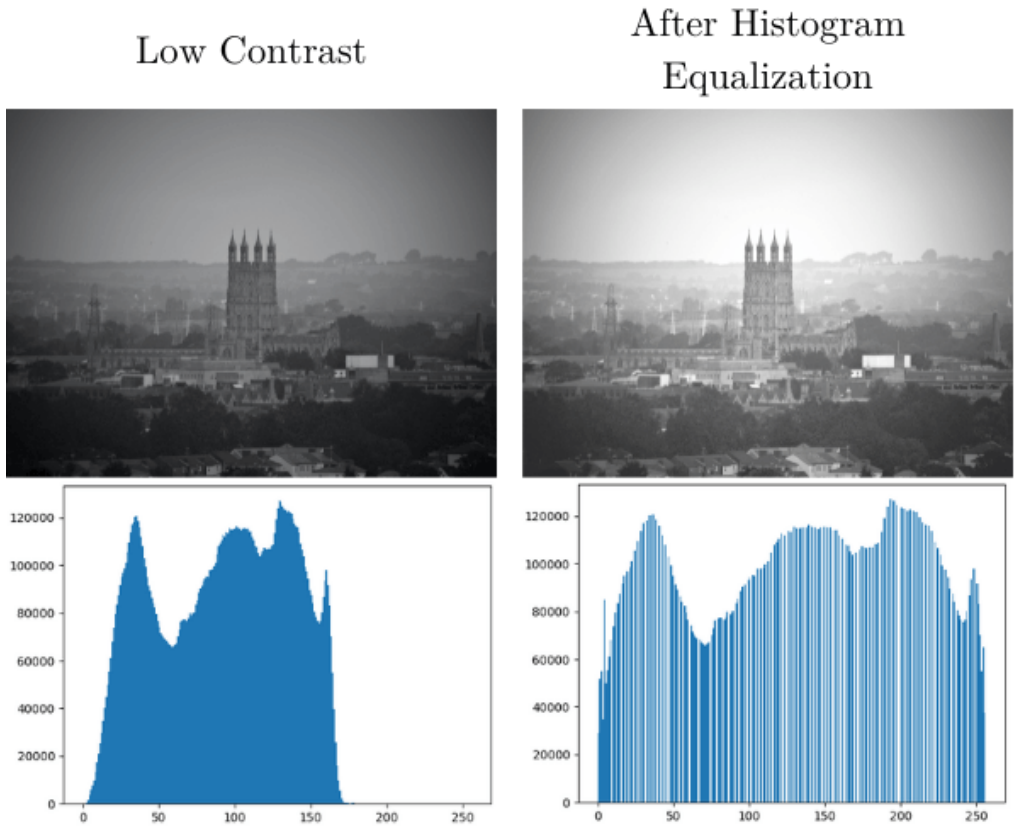
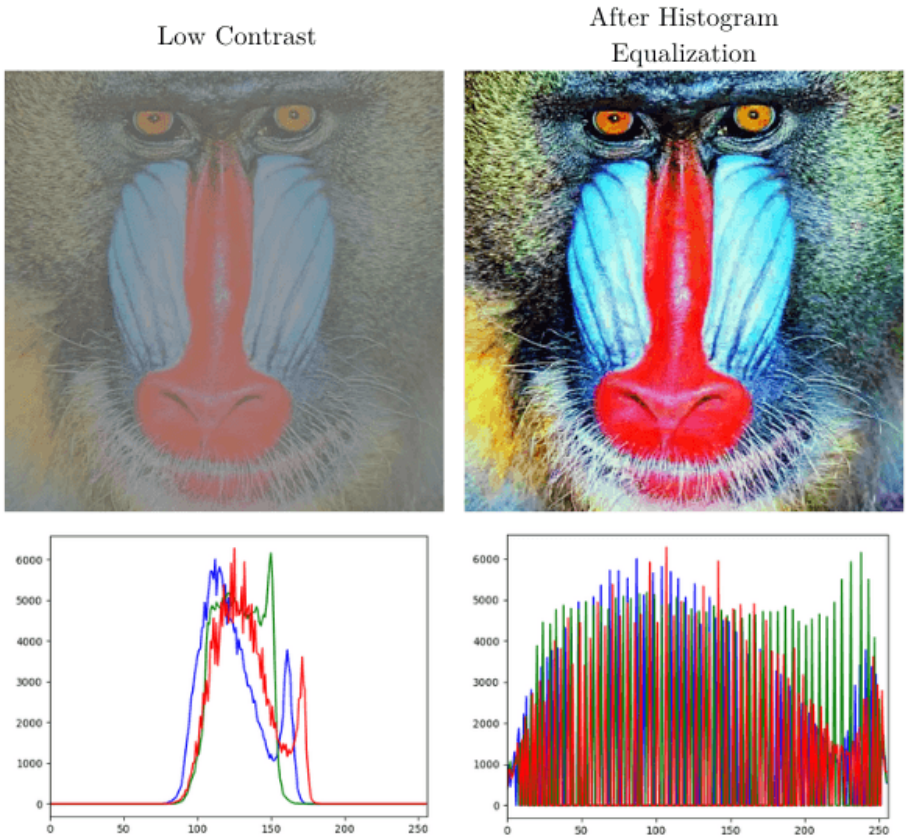


Image : Histogram : applications

- We can also perform histogram equalization in color images. In that case, the simplest approach is to equalize each RGB channel separately



Filters on image

Image filters

- **Averaging Filter**
- The Averaging Filter technique takes the average of all the pixels under the kernel area and replaces the central element. The functions cv2.blur() and cv2.boxFilter() can be used to perform the averaging filter. Both functions smooth an image using the kernel.
- Syntax :
    - **cv2.blur(image, ksize)**
    - **cv2.boxFilter(src, dst, depth, ksize, anchor, normalize, bordertype)**

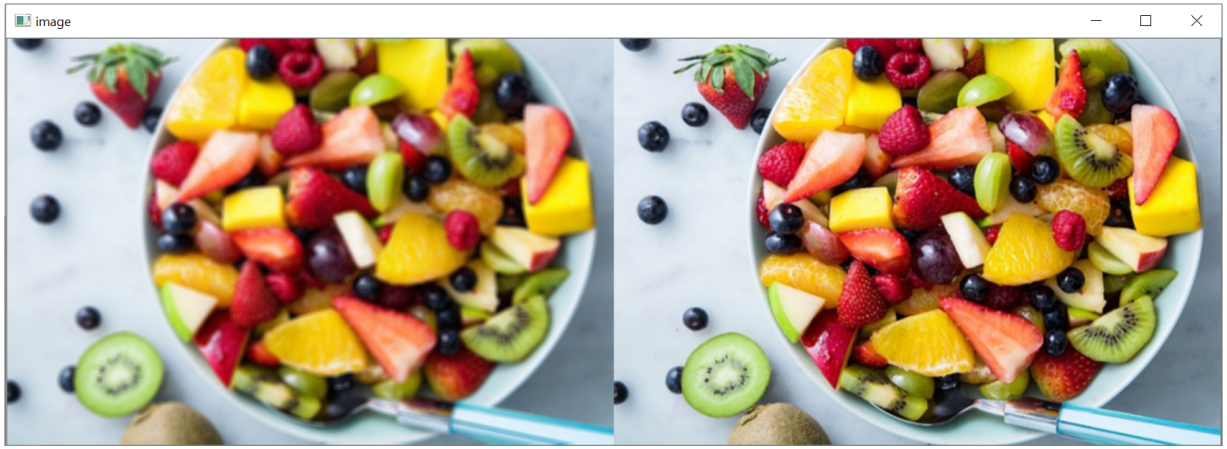


Image filters

- **Gaussian Filter**
- The OpenCV Gaussian filtering provides the cv2.GaussianBlur() method to blur an image by using a Gaussian Kernel. Each pixel in an image gets multiplied by a Gaussian Kernel. It means, a Gaussian Kernel is a square array of pixels.

- Syntax :
    - **cv2.GaussianBlur(src, ksize, sigma_x, dst, sigma_y, border_type)**

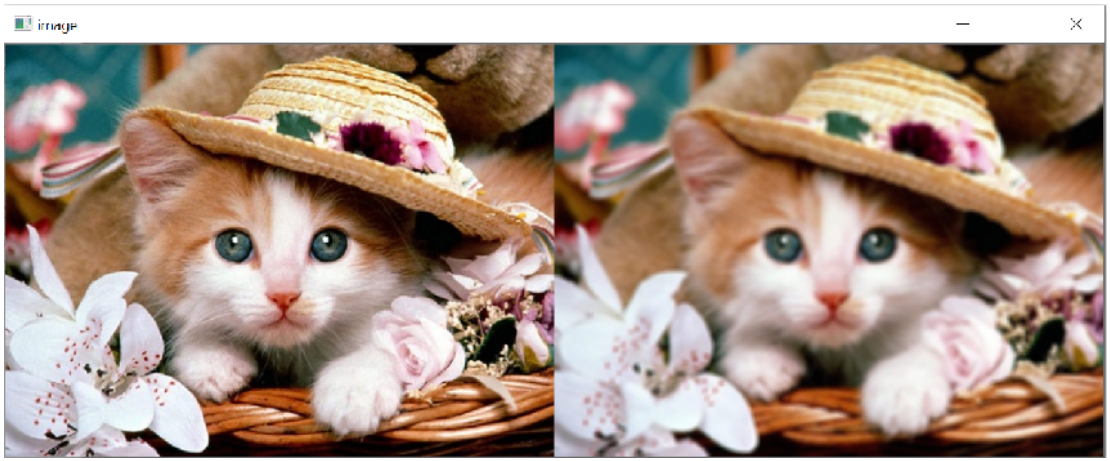


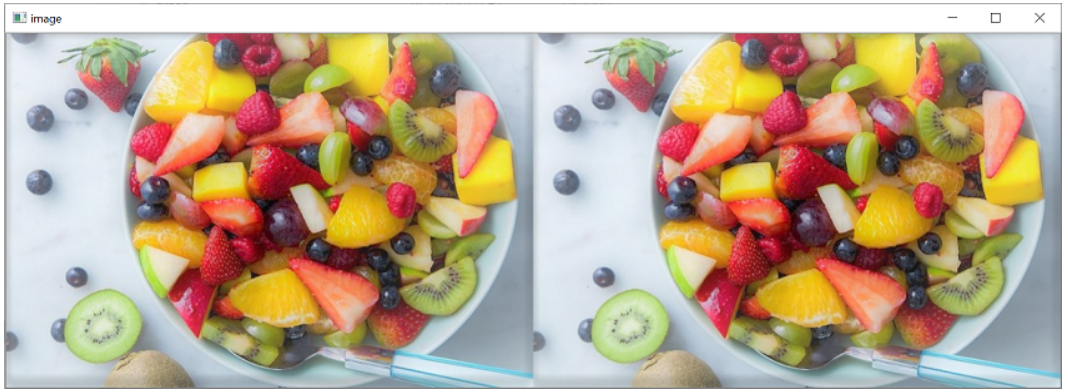
Image filters

- **Median Filtering**
- Python OpenCV provides the cv2.medianBlur() function to blur the image with a median kernel. This is a non-linear filtering technique. It is highly effective in removing salt-and-pepper noise. This takes the median of all the pixels under the kernel area and replaces the central component with this median value. Since we are taking a middle, the output image will have no new pixel esteems other than that in the input image.
- Syntax :
    - **cv2**.medianBlur(**image, ksize**)

Image filters

- **Bilateral Filter**
- Python OpenCV provides the cv2.bilateralFilter() function to blur the image with a bilateral filter. This function can be applied to reduce noise while keeping the edges sharp.

- Syntax :
    - `cv2.bilateralFilter(image, dst, d, sigmaColor, sigmaSpace)`




Thank you!