**Aim:** Processing Images with OpenCV 3

**Objective:** To Conversion between different color spaces, The Fourier Transform, High pass filter, Low pass filter, Converting between different color

**Theory:**

**Converting between different color spaces**

There are literally hundreds of methods in OpenCV that pertain to the conversion of color spaces In general, three color spaces are prevalent in modern day computer vision: gray BGR, and Hue, Saturation, Value (HSV). Gray is a color space that effectively eliminates color information translating to shades of gray: this color space is extremely useful for intermediate processing, such as face detection BGR is the blue-green-red color space, in which each pixel a three-element array, each value representing the blue, green, and red colors: web developers would be familiar with a similar definition of colors, except the order of colors is RGB. In HSV, hue is a color tone, saturation is the intensity of a color, and value represents its darkness (or brightness at the opposite end of the spectrum).

## The Fourier Transform

Much of the processing you apply to images and videos in OpenCV involves the concept of Fourier Transform in some capacity. Joseph Fourier was an 18th century French mathematician who discovered and popularized many mathematical concepts, and concentrated his work on studying the laws governing heat, and in mathematics, all things waveform. In particular, he observed that all waveforms are just the sum of simple sinusoids of different frequencies. In other words, the waveforms you observe all around you are the sum of other waveforms. This concept is incredibly useful when manipulating images, because it allows us to identify regions in images where a signal (such as image pixels) changes a lot, and regions where the change is less dramatic. We can then arbitrarily mark these regions as noise or regions of interests, background or foreground, and so on. These are the frequencies that make up the original image, and we have the power to separate them to make sense of the image and extrapolate interesting data.

## High pass filter

A high pass filter (HPF) is a filter that examines a region of an image and boosts the intensity of certain pixels based on the difference in the intensity with the surrounding pixels.

Take, for example, the following kemel:

[[0, -0.25, 0],

[-0.25, 1, -0.25],

[0,-0.25, 0]]

compared to all the immediate neighbors, the intensity of the central pixel will be boosted (or not) if a high level of changes are found. In other words, if a pixel stands out from the surrounding pixels, it will get boosted. This is particularly effective in edge detection, where a common form of HPF called high boost filter is used. Both high pass and low pass filters use a property called radius, which extends the area of the neighbors involved in the filter calculation

**Low pass filter**

If an HPF boosts the intensity of a pixel, given its difference with its neighbors, a low pass filter (LPF) will smoothen the pixel if the difference with the surrounding pixels is lower than a certain threshold. This is used in denoising and blurring. For example, one of the most popular blurring/smoothening filters, the Gaussian blut, is a low pass filter that attenuates the intensity of high frequency signals

## Conclusion

An image is a visual representation of anything, and computer vision could readily handle it. The term "AI field" is referred to by the acronym "cv." A large open-source library for image processing, machine learning, and computer vision is called OpenCV. It can analyse pictures and movies to find faces, objects, and even human handwriting.