DFT AND ITS APPLICATIONS

A MAJOR PROJECT REPORT

Submitted by

CH.EN.U4AIE20001	ANBAZHAGAN E
CH.EN.U4AIE20053	RAMYA POLAKI
CH.EN.U4AIE20069	TRINAYA KODAVATI
CH.EN.U4AIE20031	SMITHIN REDDY K
CH.EN.U4AIE20035	ABHIRAM KUNCHAPU

in partial fulfilment for the award of the degree

Of

BACHELOR OF TECHNOLOGY

IN

MATHEMATICS FOR INTELLIGENCE SYSTEM

श्रद्धावान् लभते ज्ञानम्

AMRITA SCHOOL OF ENGINEERING, CHENNAI

AMRITA VISHWA VIDYAPEETHAM

CHENNAI – 601103, TAMIL NADU

AMRITA VISHWA VIDYAPEETHAM

AMRITA SCHOOL OF ENGINEERING, CHENNAI, 601103

BONAFIDE CERTIFICATE

This is to certify that the major project report entitled "**DFT AND ITS APPLICATIONS"** submitted by

CH.EN.U4AIE20001	ANBAZHAGAN E
CH.EN.U4AIE20053	RAMYA POLAKI
CH.EN.U4AIE20069	TRINAYA KODAVATI
CH.EN.U4AIE20031	SMITHIN REDDY K
CH.EN.U4AIE20035	ABHIRAM KUNCHAPU

in partial fulfilment of the requirements for the award of the **bachelor of Master of Technology** in **COMPUTER SCIENCE ENGINERRING** is a bonafide record of the work carried out under my guidance and supervision at Amrita School of Engineering, Chennai.

Signature

Dr T Dilleswar Rao

CSE Dept professor

This project report was evaluated by us on ……………...

INTERNAL EXAMINER   EXTERNAL EXAMINER

ACKNOWLEDGEMENT:

I offer my sincere pranams at the lotus feet of Universal guru, MATA AMRITANANDAMAYI DEVI who blessed me with her grace to make this a successful project.

I express my deep sense of gratitude to Dr. P Shankar, Principal, Amrita School of Engineering, Chennai for his kind support. I would like to extend my gratitude and heartfelt thanks to Dr T Dilleswar Rao CSE Department and Dr. A Manikandan, , , for his constant help, suggestions and inspiring guidance. I am grateful to my guide, Department of MIS , for his invaluable support and guidance during the course of the major project work. I am also indebted to Dr. Prasanna Kumar, Chairperson of Department of AI, ASE, Chennai for his guidance.

I specially thank our director Sri I B Manikandan who supported us in every possible manner.

INDEX

Topic **Page. No**

DFT AND ITS APPLICATIONS

1. Abstract

This report consists of DFT and some of its applications. Image Processing is changing the nature of an image in order to improve its pictorial information for human interpretation, for autonomous machine perception. Digital image processing is a subset of the electronic domain wherein the image is converted to an array of small integers, called pixels, representing a physical quantity such as scene radiance, stored in a digital memory, and processed by computer or other digital hardware.

With the development of communication technology, voice communication has become a major communication media for people to transmit information more convenient. Audio noise reduction system is the system that is used to remove the noise from the audio signals. The report consisted of an audio file using record function and has been save Audio on a formula (WAV), and the report used matlab software to read sound and design low pass filter, then insert the audio signal with the noise signal into the filter and output a signal audio without noise.

2. Introduction

2.1 What is DFT?

The discrete Fourier transform (DFT) is a complex-valued function of frequency that turns a finite sequence of equally-spaced samples of a function into a same-length sequence of equally-spaced samples of the discrete-time Fourier transform (DTFT). The sampling interval for the DTFT is the reciprocal of the input sequence's duration. The DTFT samples are used as coefficients of complex sinusoids at the relevant DTFT frequencies in an inverse DFT, which is a Fourier series. The sample values in t are identical to those in the original input sequence. As a result, the DFT is referred to as a frequency domain representation of the input sequence. The DTFT is continuous (and periodic) if the original sequence encompasses all non-zero values of a function, while the DFT delivers discrete samples of one cycle. The DFT provides all the non-zero values of one DTFT cycle if the original sequence is one cycle of a periodic function.

The Fast Fourier Transform (FFT) is a mathematical approach for computing the Discrete Fourier Transform (DFT) of a sequence. The main difference between FT (Fourier Transform) and FFT is that FT considers a continuous signal as input, whereas FFT examines a discrete signal. DFT converts a sequence into its frequency constituents in the same way as FT converts a continuous signal into its frequency constituents.

Formulae:

DFT transform:

$$
\begin{aligned}
X_k &= \sum_{n=0}^{N-1} x_n \cdot e^{-\frac{i2\pi}{N}kn} \\
&= \sum_{n=0}^{N-1} x_n \cdot \left[\cos\left(\frac{2\pi}{N}kn\right) - i \cdot \sin\left(\frac{2\pi}{N}kn\right)\right],
\end{aligned}
$$

DFT is usually defined for a discrete function f(m,n) that is nonzero only over the finite region $0 \leq \mathrm{m} \leq \mathrm{M}-1$ and $0 \leq \mathrm{n} \leq \mathrm{N}-1$. The two-dimensional *M*-by-*N* DFT and inverse *M*-by-*N* DFT relationships are given by

$$
F(p,q) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m,n) e^{-j(2\pi/M)pm} e^{-j(2\pi/N)qn} \qquad \begin{array}{l} p = 0, 1, \ldots, M-1 \\ q = 0, 1, \ldots, N-1 \end{array}
$$

$$
f(m,n) = \frac{1}{MN} \sum_{p=0}^{M-1} \sum_{q=0}^{N-1} F(p,q) e^{j(2\pi/M)pm} e^{j(2\pi/N)qn} \qquad \begin{array}{l} m = 0, 1, \ldots, M-1 \\ n = 0, 1, \ldots, N-1 \end{array}
$$

Inverse transform:

The discrete Fourier transform is an invertible, linear transformation

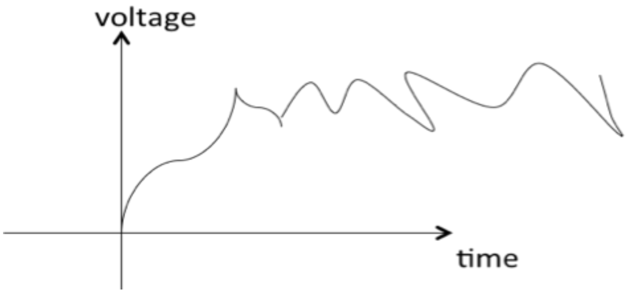
$$
\mathcal{F}: \mathbb{C}^N \to \mathbb{C}^N
$$

with C denoting the set of complex numbers. Its inverse is known as Inverse Discrete Fourier Transform (IDFT). In other words, for any N>0 , an N-dimensional complex vector has a DFT and an IDFT which are in turn N-dimensional complex vectors.

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k \cdot e^{i \frac{2\pi}{N} kn}$$

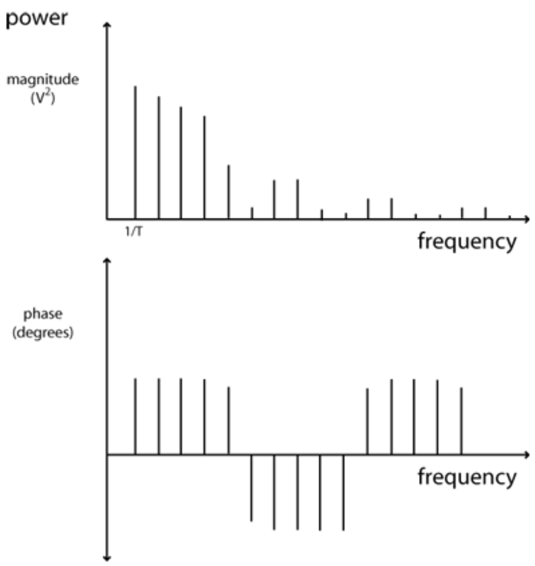
2.2 Time domain:

A time domain analysis is an analysis of physical signals, mathematical functions, or time series of economic or environmental data, in reference to time. Also, in the time domain, the signal or function's value is understood for all real numbers at various separate instances in the case of discrete-time or the case of continuous-time. time-domain graph can show how a signal changes with time. In general, when an analysis uses a unit of time, such as seconds or one of its multiples (minutes or hours) as a unit of measurement, then it is in the time domain.

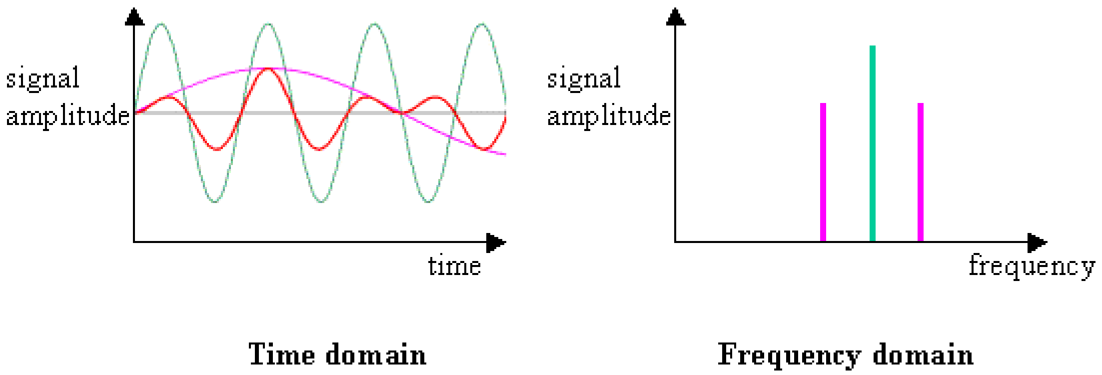


2.3 Frequency domain:

the frequency domain refers to the analysis of mathematical functions or signals with respect to frequency, rather than time. Over a range of frequencies, a frequency-domain graph illustrates how much of the signal is included inside each frequency band. The phase shift that must be applied to each sinusoid in order to recombine the frequency components and recover the original time signal can also be included in a frequency-domain representation.



Frequency Domain



Time domain vs frequency domain

3. Applications of DFT

The DFT is the most significant discrete transform, and it is utilised in many practical applications to perform Fourier analysis. Any quantity or signal that fluctuates over time, such as the pressure of a sound wave, a radio signal, or daily temperature data, sampled over a finite time interval, is referred to as a function in digital signal processing.

The samples in image processing can be the values of pixels along a raster picture's row or column. The DFT may also be used to solve partial differential equations quickly, as well as execute additional operations like convolutions and multiplying big numbers.

It may be implemented in computers using numerical techniques or even specialised hardware because it deals with a finite amount of data. Fast Fourier transform (FFT) methods are commonly utilised in these implementations; in fact, the words "FFT" and "DFT" are frequently interchanged. The initialism "FFT" may have been used for the misleading word "finite Fourier transform" prior to its present usage.

3.1 Noise Cancellation:

One of the most frequent approaches for removing noise from a spectrum is to use the spectrum subtraction method. For frequency analysis, the spectrum subtraction approach, like many other noise reduction methods, employs the discrete Fourier transform (DFT). In DFT, the frequency and time resolution are usually traded off.

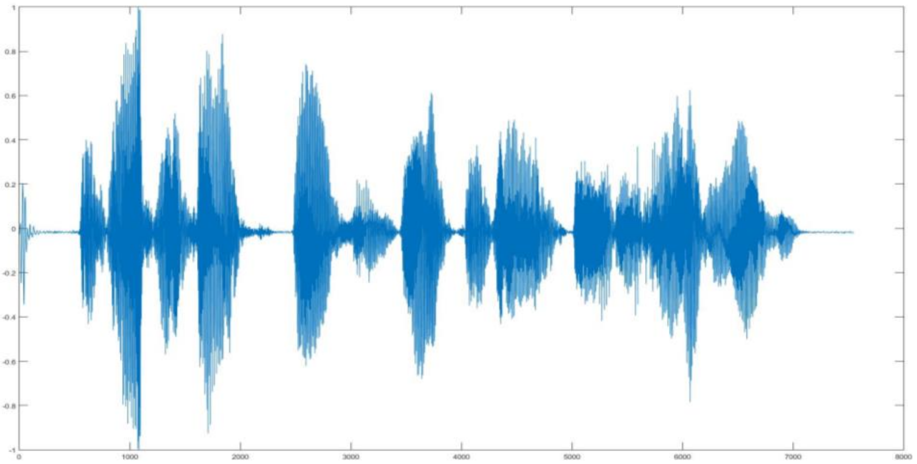
One way to reduce the error is to record the signal for longer or try to get the recording device closer to the source (or increase the amplitude of the signal). Occasionally, neither of these methods are possible, which is when other techniques need to be employed such as windowing or time/*frequency filtering.*

3.1.1) Audio Signal:

Audio signal processing is a subfield of signal processing that is concerned with the electronic manipulation of audio signals. Audio signals are electronic representations of sound waves—longitudinal waves which travel through air, consisting of compressions and rarefactions

It is a processing is an engineering field that focus on the computational methods for intentionally altering auditory signals or sounds

A multi-track recording or sound reinforcement operation uses an audio track, which is an audio signal communications channel in a storage device or mixing console.



Audio Signal

3.1.2) Signal:

In signal processing, a signal is a function that conveys information about a phenomenon

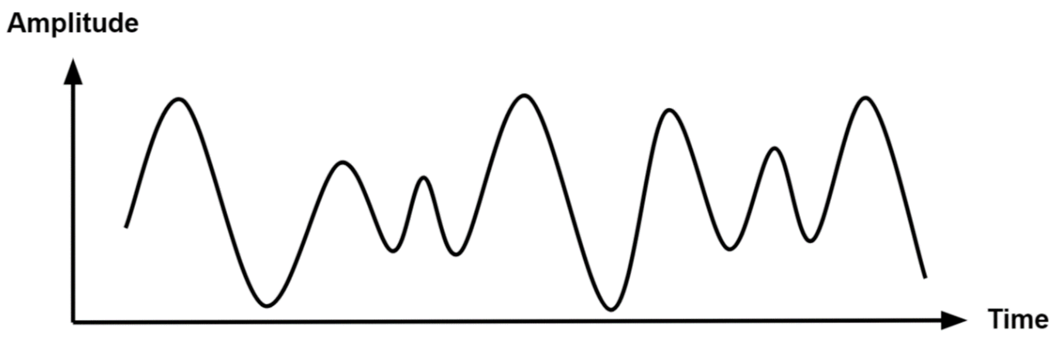
There are two types of Signals:

1) Analog Signal-

A continuous signal in which one time-varying quantity (such as voltage, pressure, etc.) reflects another time-based variable is called an analogue signal. To put it another way, one variable is a replica of the other.

We use analog signals in a wide variety of applications, such as:

- Audio recording and reproduction
- Live sound/amplification devices
- Older video signal transmission technologies (VGA, S-Video, etc.)
- Radio signals



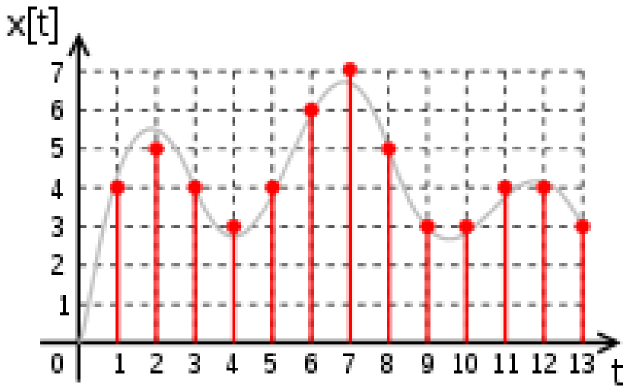
Analog signal

2) Digital Signal-

A digital signal is one that is used to represent data as a sequence of discrete values; it can only take on one of a finite number of values at any given time.

Digital Signal is primarily used-

- audio signal,
- speech processing
- RADAR
- seismology
- SONAR
- voice recognition
- some financial signals

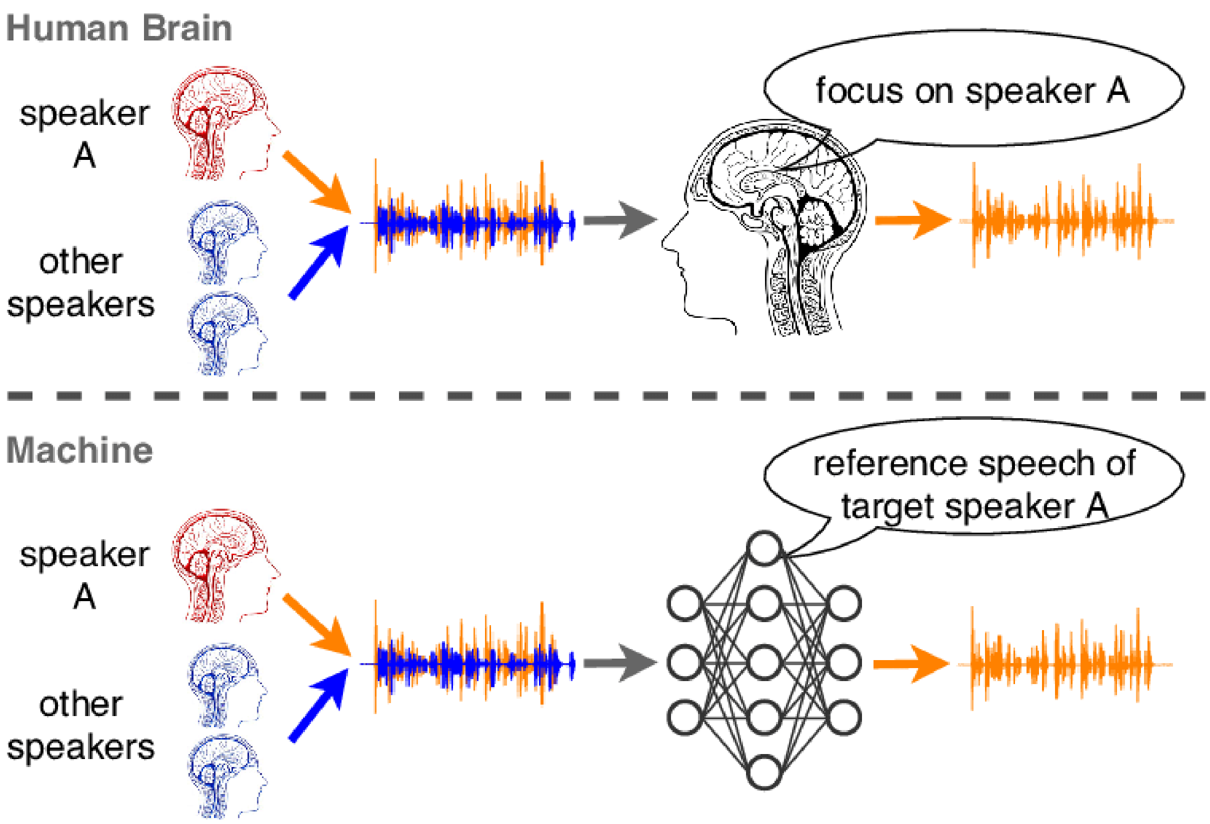


Digital Signal

3.1.3) Signal Processing-

Signal processing is a branch of electrical engineering that studies, modifies, and synthesises signals such as sound, pictures, and scientific measurements.

When a signal passes through a system, it is essentially processed. Signal processing, in other terms, is the term used to describe the processes that a system does. Processing refers to manipulating a signal in some way in order to extract usable information, such as when we use our ears as an input device and then use auditory pathways in the brain to retrieve the data.



Basic idea of Signal processing

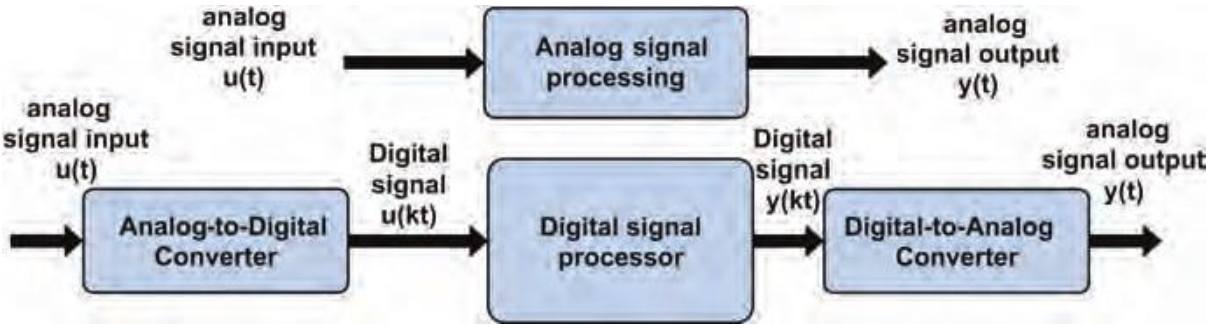
Types of Signal Processing:

1) Analog or Continues time Method:

It is a type of signal processing that uses analogue techniques to process continuous analogue signals. The term "analogue" refers to something that can be represented mathematically as a series of continuous values. Crossover filters in loudspeakers, stereo "bass," "treble," and "volume" controls, and TV "tint" controls are all examples of analogue signal processing.

2) **Digital signal processing (DSP)** :

It is the use of digital processing to conduct a wide range of signal processing functions, such as by computers or more specialised digital signal processors. The digital signals are a series of integers that represent samples of a continuous variable in a domain like time, space, or frequency.



3.1.4 Noise Removal using PSD:

- The FFT of the autocorrelation function of discrete-time noise samples reminds the causality of random stochastic signals.
- Noises of two different frequencies can have the same PDF, but PSD will be different.
- The computation of noise PSD gives the normalized frequency distribution of noise powers and helps the designer modify the circuit design, layout, and architecture for noise reduction.
- Noise power spectral density (PSD) analysis is a powerful tool to identify the harmonics and electromagnetic emissions in a circuit. PSD indicates the power of noise signals distributed over the frequency. Measuring the noises in the time domain and converting them into the frequency domain is like extracting useful information from bulk amounts of unprocessed data. The characterization of the noise in PSD analysis utilizes the Fast Fourier Transform (FFT) of the autocorrelation function of the discrete noise signal.
- The FFT approach is efficient in estimating the spectral information of dominant noise powers. The estimation of noise distributions through PSD analysis helps the hardware engineer adapt the circuit design so that the system works satisfactorily under a specified bandwidth of interest.

Noise in a circuit can be externally or internally generated. The noise gets superimposed into the circuit signals from power sources, ground lines, switching, or other devices. The operating temperature of a system also propagates noise as thermal noise. Based on the frequency spectrum, the noises can be classified as:

1. Wide-band noise: A wide range of amplitudes and frequency components are present in this type of noise.
2. Impulse noise: The sharp spikes that exponentially decay to both sides over a frequency band. It is intermittent and difficult to detect and analyze.

3. Frequency-specific noise: This type of noise is characterized by constant frequency and the noise amplitude is dependent on the distance of the affected location from the noise source.

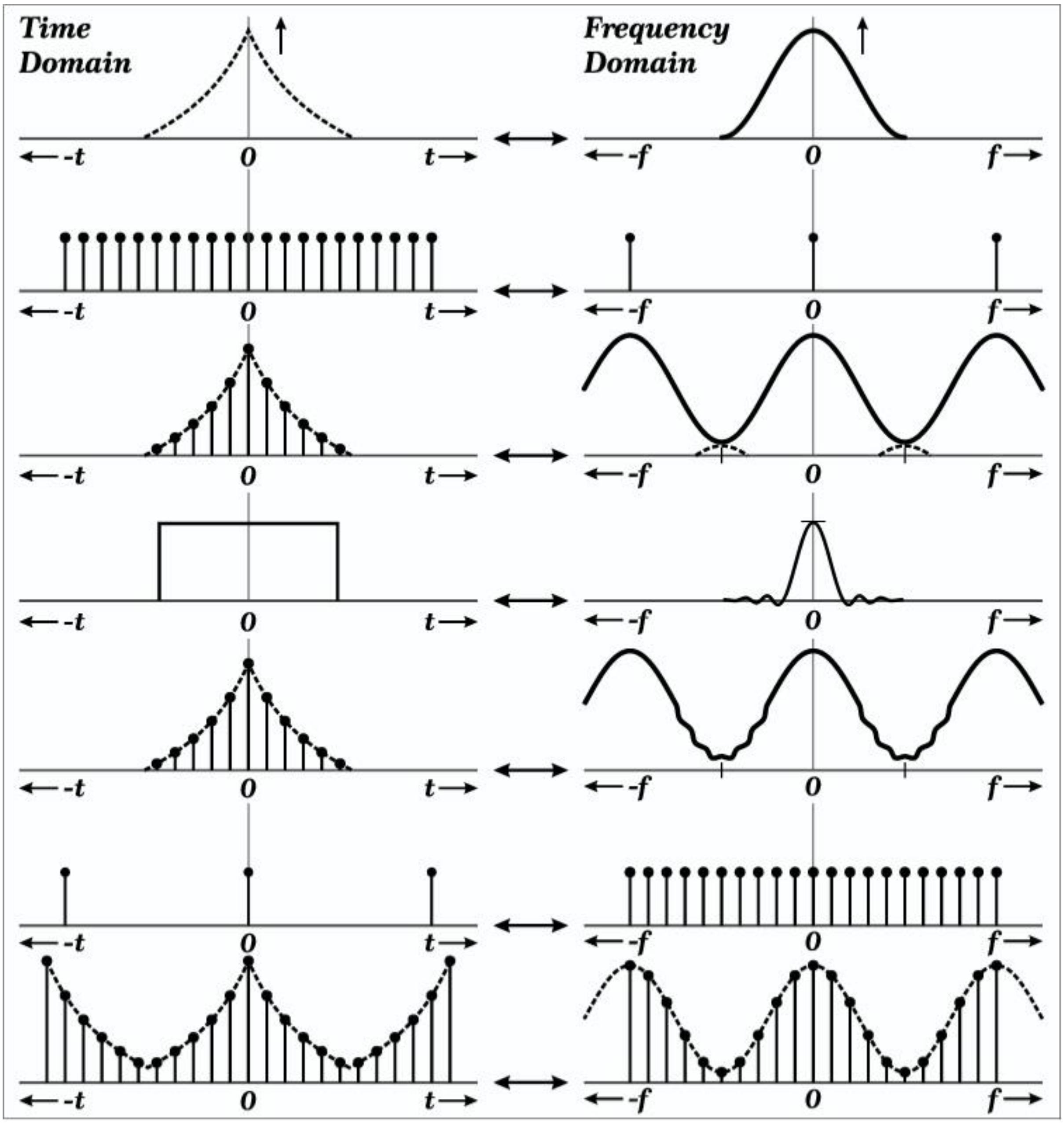
The PSD of a discrete-time noise signal is given by the FFT of its autocorrelation function, R(k).

Discrete

$$S(f) = \sum_{k=1}^{N} R(k) e^{-i2\pi f k}$$

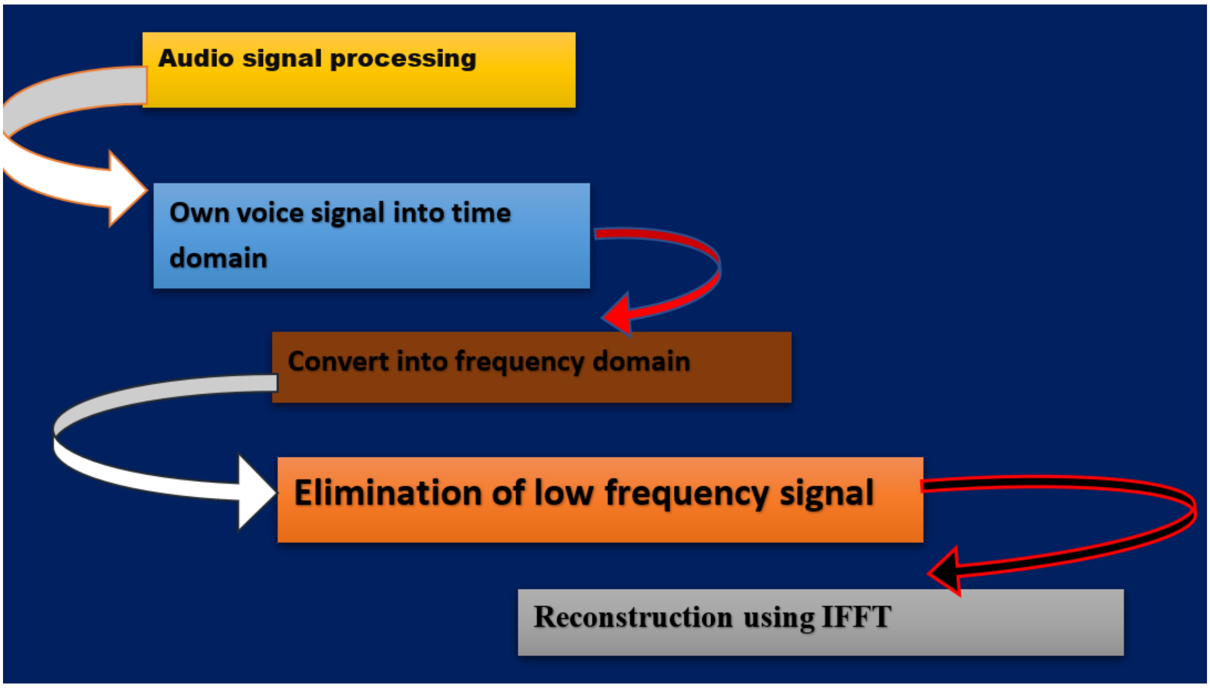
$$R(k) = \frac{1}{N} \sum_{n=1}^{N} x(n) x(n-k)$$

we know that PSD gives the noise powers (W) vs. frequency (Hz). The sampling of the noise consolidates the noise amplitude occurrences over sufficient time and transforms the analysis from continuous to the discrete-time domain. For discrete-time signals, FFT is the most convenient tool for calculating spectral noise power distribution. However, we go for the FFT of autocorrelation function instead of computing the FFT of the direct discrete-time noise signal.



The frequency-domain analysis illustrates the domination of noise frequencies in the system output.

3.1.5 Theory Behind Noise Removal:



3.2 Image Processing:

The Fourier Transform is an important image processing tool which is used to decompose an image into its sine and cosine components. The output of the transformation represents the image in the *Fourier* or frequency domain, while the input image is the spatial domain equivalent. In the Fourier domain image, each point represents a particular frequency contained in the spatial domain image.

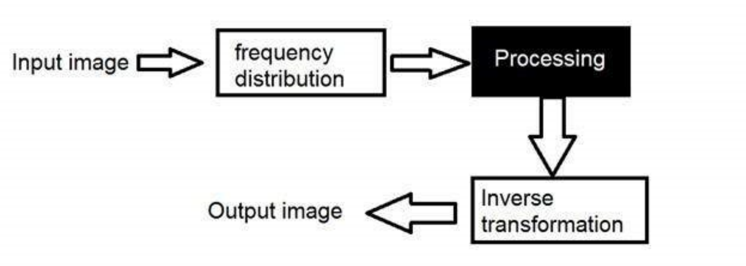
The Fourier Transform is used in a wide range of applications, such as image analysis, image filtering, image reconstruction and image compression.

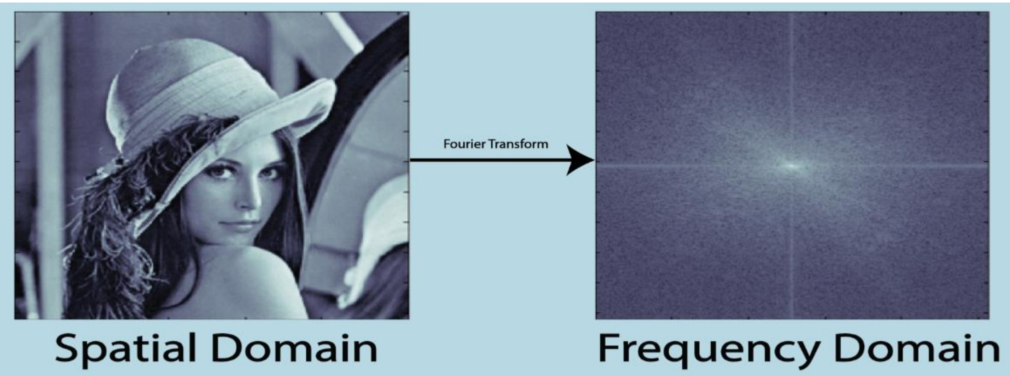
3.2.1 Fourier Transformed Image

A digital image is transformed from the spatial domain to the frequency domain in the frequency domain. Image filtering in the frequency domain is used to enhance images for a specific application. A frequency domain tool called a Fast Fourier transformation is used to translate the spatial domain to the frequency domain.

The Fourier transform defines the frequency domain as a space. In image processing, the Fourier transform has a wide range of applications. The main advantage of Fourier analysis is that very little information is lost from the signal during the transformation. The Fourier transform maintains information on amplitude, harmonics, and phase and uses all parts of the waveform to translate the signal into the frequency domain. Preservation of phase information by the Fourier transformation means that the signal can be transformed back into the time domain. The frequency domain analysis method is used to show how signal energy can be dispersed throughout a frequency range. The basic premise of frequency domain analysis is to compute the image's 2D discrete Fourier transform.

The image in the Fourier or frequency domain is represented by the output of the transformation, whilst the spatial domain equivalent is represented by the input image. Each point in the Fourier domain image indicates a frequency contained in the spatial domain image.





3.2.2 Image in Magnitude Domain:

The Fourier Transform produces a complex number valued output image which can be displayed with two images, either with the real and imaginary part or with magnitude and phase. In image processing, often only the magnitude of the Fourier Transform is displayed, as it contains most of the information of the geometric structure of the spatial domain image. However, if we want to re-transform the Fourier image into the correct spatial domain after some processing in the frequency domain, we must make sure to preserve both magnitude and phase of the Fourier image.

The Fourier domain image has a much greater range than the image in the spatial domain. Hence, to be sufficiently accurate, its values are usually calculated and stored in float values.

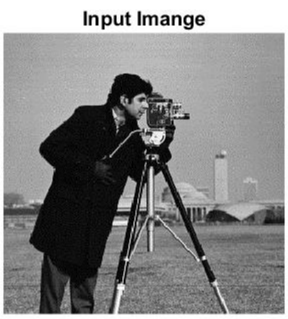
In most implementations the Fourier image is shifted in such a way that the DC-value (i.e. the image mean) F(0,0) is displayed in the center of the image. The further away from the center an image point is, the higher is its corresponding frequency.

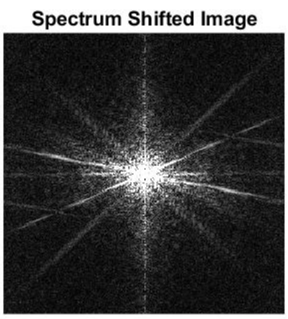
We start off by applying the Fourier Transform of given image .

The magnitude calculated from the complex result .So

We can see that the DC-value is by far the largest component of the image. However, the dynamic range of the Fourier coefficients (i.e. the intensity values in the Fourier image) is too large to be displayed on the screen, therefore all other values appear as black. If we apply a logarithmic transformation to the image we obtain is a spectral shifted frequency domain image

The result shows that the image contains components of all frequencies, but that their magnitude gets smaller for higher frequencies. Hence, low frequencies contain more image information than the higher ones.





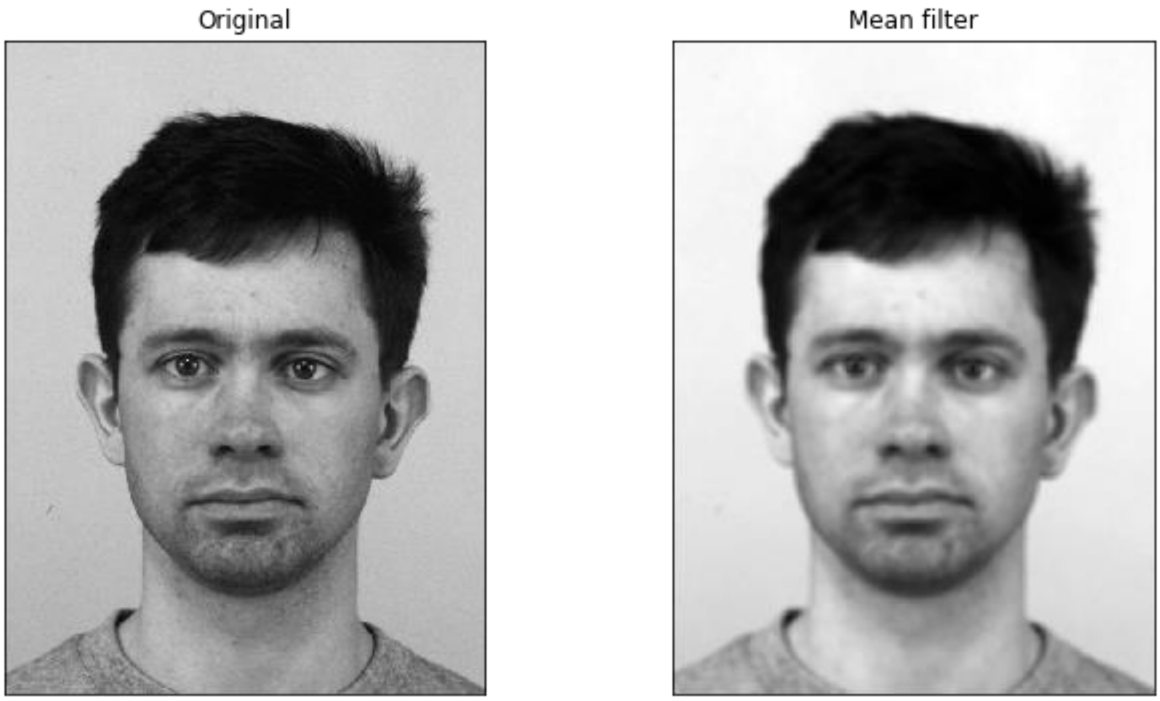
3.2.3 Filtering of Image:

In our code we used Gaussian blur to Filter the Image

Gaussian blur:

In image processing, a Gaussian blur (also known as Gaussian smoothing) is the result of blurring an image by a Gaussian function (named after mathematician and scientist Carl Friedrich Gauss).

It is a widely used effect in graphics software, typically to reduce image noise and reduce detail. The visual effect of this blurring technique is a smooth blur resembling that of viewing the image through a translucent screen, distinctly different from the bokeh effect produced by an out-of-focus lens or the shadow of an object under usual illumination.



Smoothing of an image

How It Works

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 Figure 1.

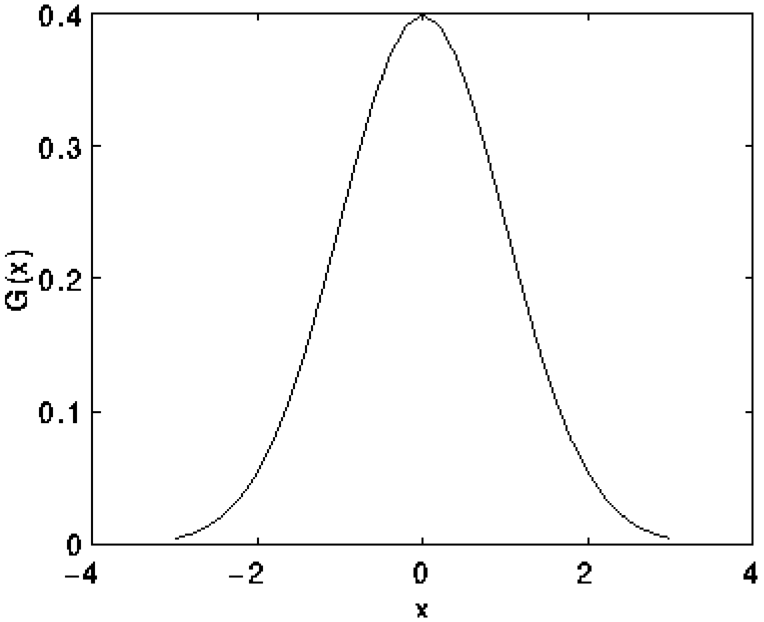


Figure 1 1-D Gaussian distribution with mean 0 and σ =1

In 2-D, an isotropic (i.e. circularly symmetric)

Gaussian has the form:

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

This distribution is shown in Figure 2.

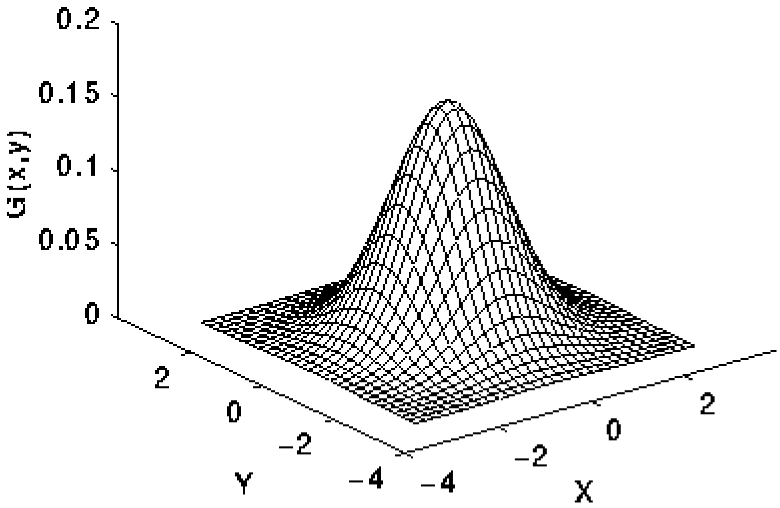


Figure 2 2-D Gaussian distribution with mean (0,0) and σ =1

The idea of Gaussian smoothing is to use this 2-D distribution as a `point-spread' function, and this is achieved by convolution. Since the image is stored as a collection of discrete pixels we need to produce a discrete approximation to the Gaussian function before we can perform the convolution. In theory, the Gaussian distribution is non-zero everywhere, which would require an infinitely large convolution kernel, but in practice it is effectively zero more than about three standard deviations from the mean, and so we can truncate the kernel at this point. Figure 3 shows a suitable integer-valued convolution kernel that approximates a Gaussian with a σ of 1.0. It is not obvious how to pick the values of the mask to approximate a Gaussian. One could use the value of the Gaussian at the centre of a pixel in the mask, but this is not accurate because the value of the Gaussian varies non-linearly across the pixel. We integrated the value of the Gaussian over the whole pixel (by summing the Gaussian at 0.001 increments). The integrals are not integers: we rescaled the array so that the corners had the value 1. Finally, the 273 is the sum of all the values in the mask.

$$\frac{1}{273}$$

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

Figure 3 Discrete approximation to Gaussian function with σ =1.0

Once a suitable kernel has been calculated, then the Gaussian smoothing can be performed using standard convolution methods. The convolution can in fact be performed fairly quickly since the equation for the 2-D isotropic Gaussian shown above is separable into x and y components. Thus the 2-D convolution can be performed by first convolving with a 1-D Gaussian in the x direction, and then convolving with another 1-D Gaussian in the y direction. (The Gaussian is in fact the only completely circularly symmetric operator which can be decomposed in such a way.) Figure 4 shows the 1-D x component kernel that would be used to produce the full kernel shown in Figure 3 (after scaling by 273, rounding and truncating one row of pixels around the boundary because they mostly have the value 0. This reduces the 7x7 matrix to the 5x5 shown above.). The y component is exactly the same but is oriented vertically.

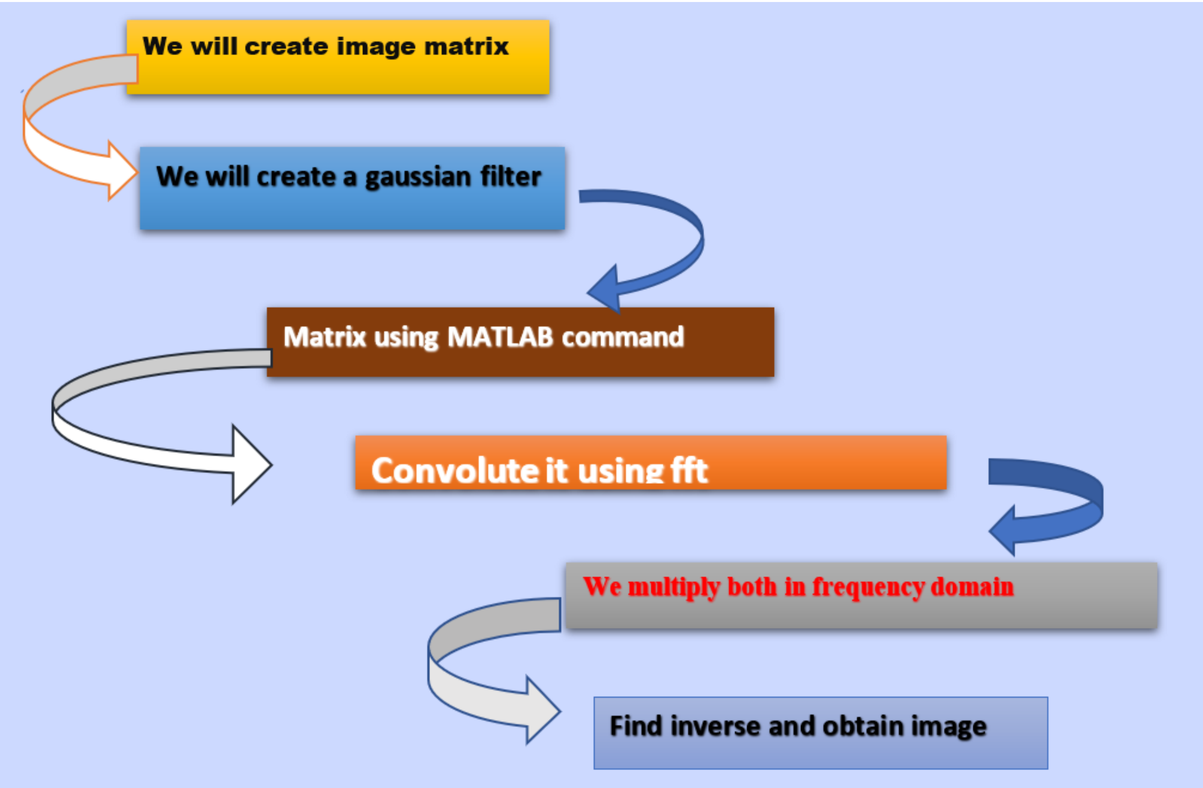
.006	.061	.242	.383	.242	.061	.006

Figure 4 One of the pair of 1-D convolution kernels.

A further way to compute a Gaussian smoothing with a large standard deviation is to convolve an image several times with a smaller Gaussian. While this is computationally complex, it can have applicability if the processing is carried out using a hardware pipeline.

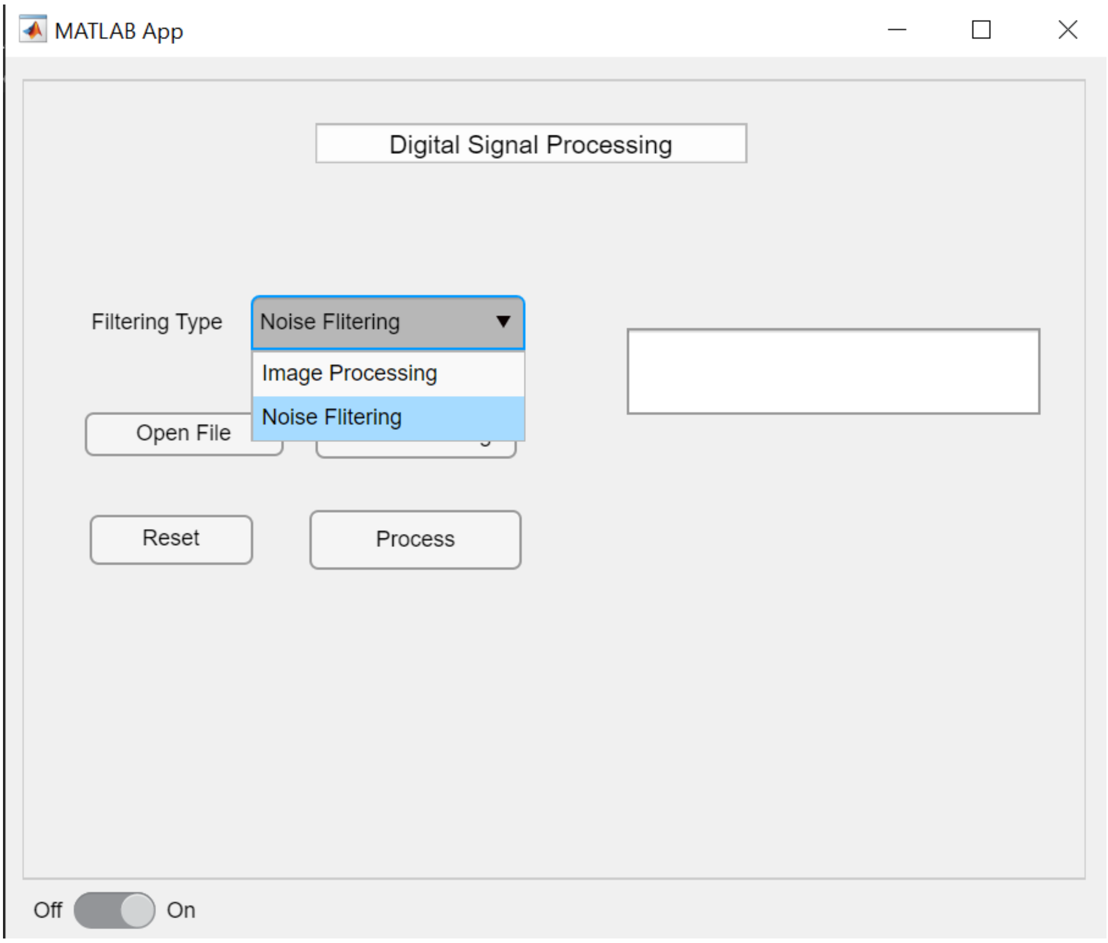
The Gaussian filter not only has utility in engineering applications. It is also attracting attention from computational biologists because it has been attributed with some amount of biological plausibility, e.g. some cells in the visual pathways of the brain often have an approximately Gaussian response.

3.2.4) Theory behind Image Filtering:



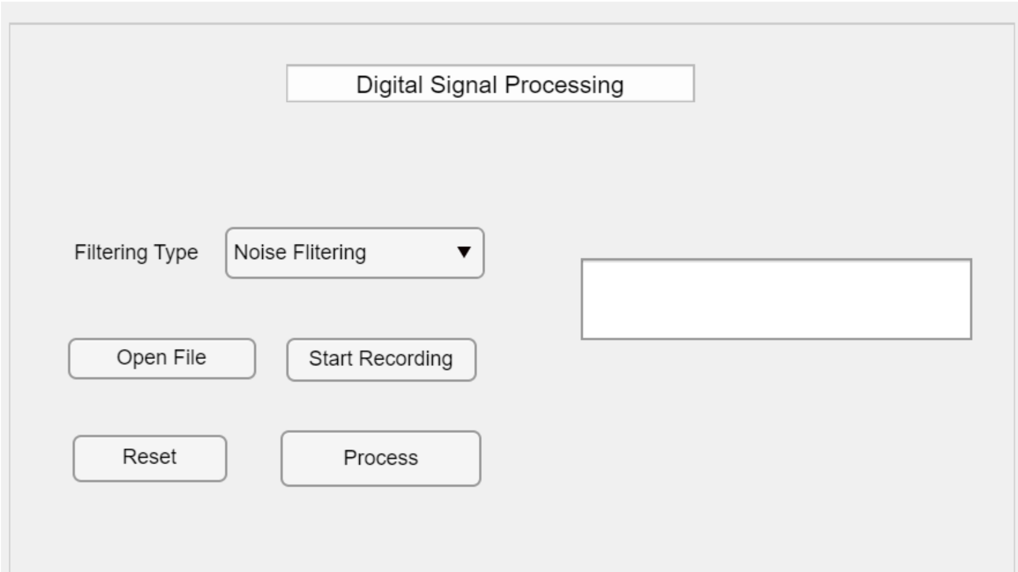
4) MATLAB GUI

ANIMATION FOR NOISE REMOVAL & IMAGE PROCESSING:

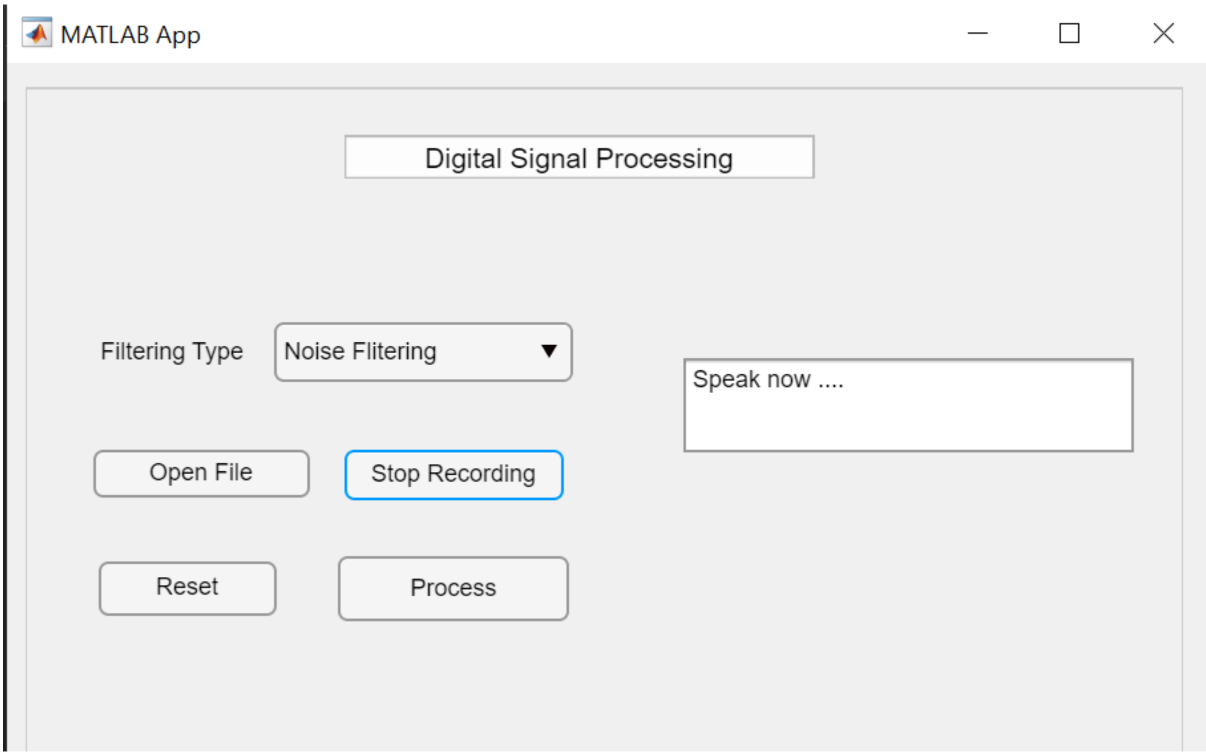


NOISE FILTERING:

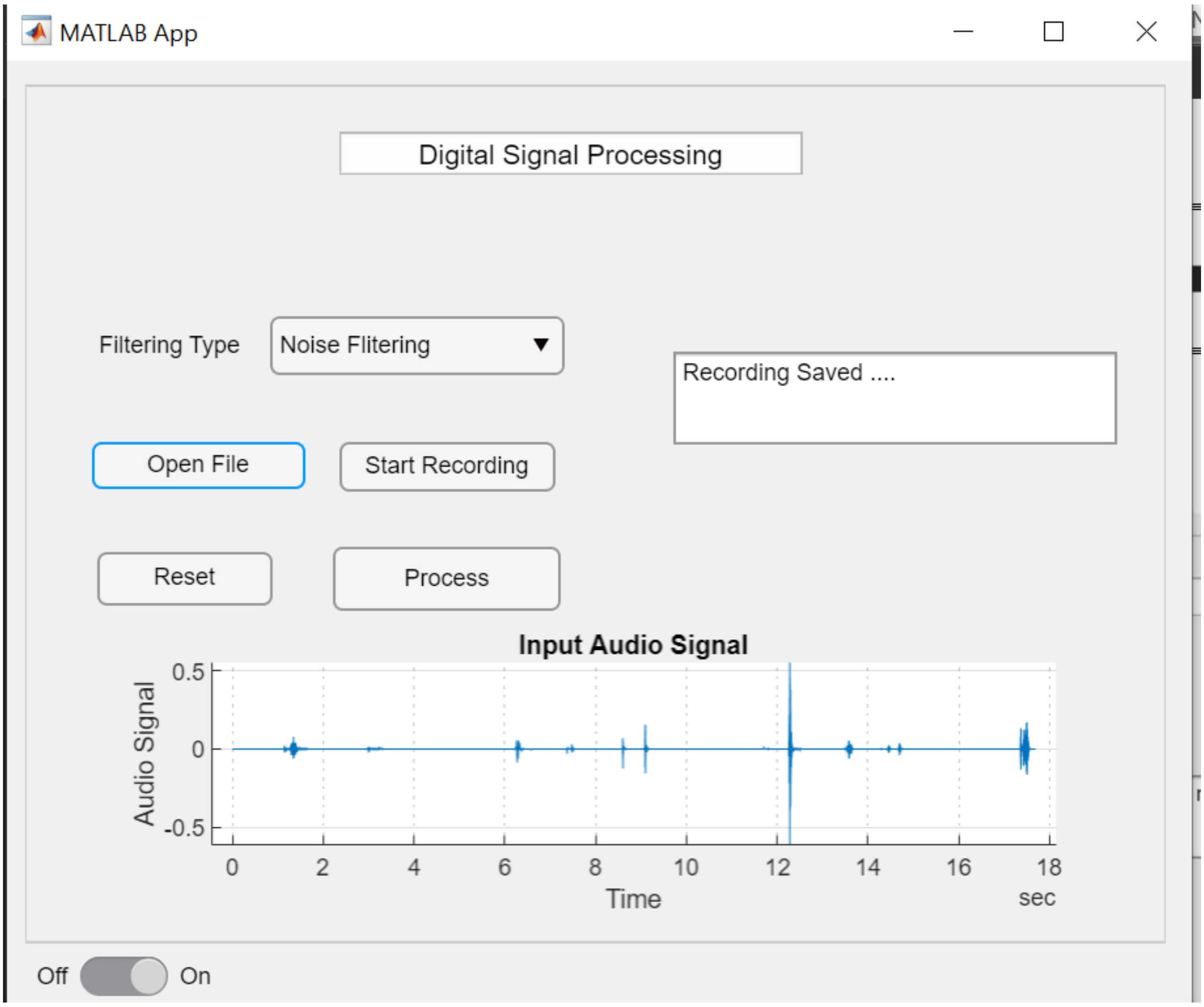
Step1: Click on start Recording



Step2: After Recording click on stop recording and save audio



Step 3: Open the saved File



Step4:Click on process

Step 5 : Identify the Power Spectral Density Function and write it in the value option and Click on Filter

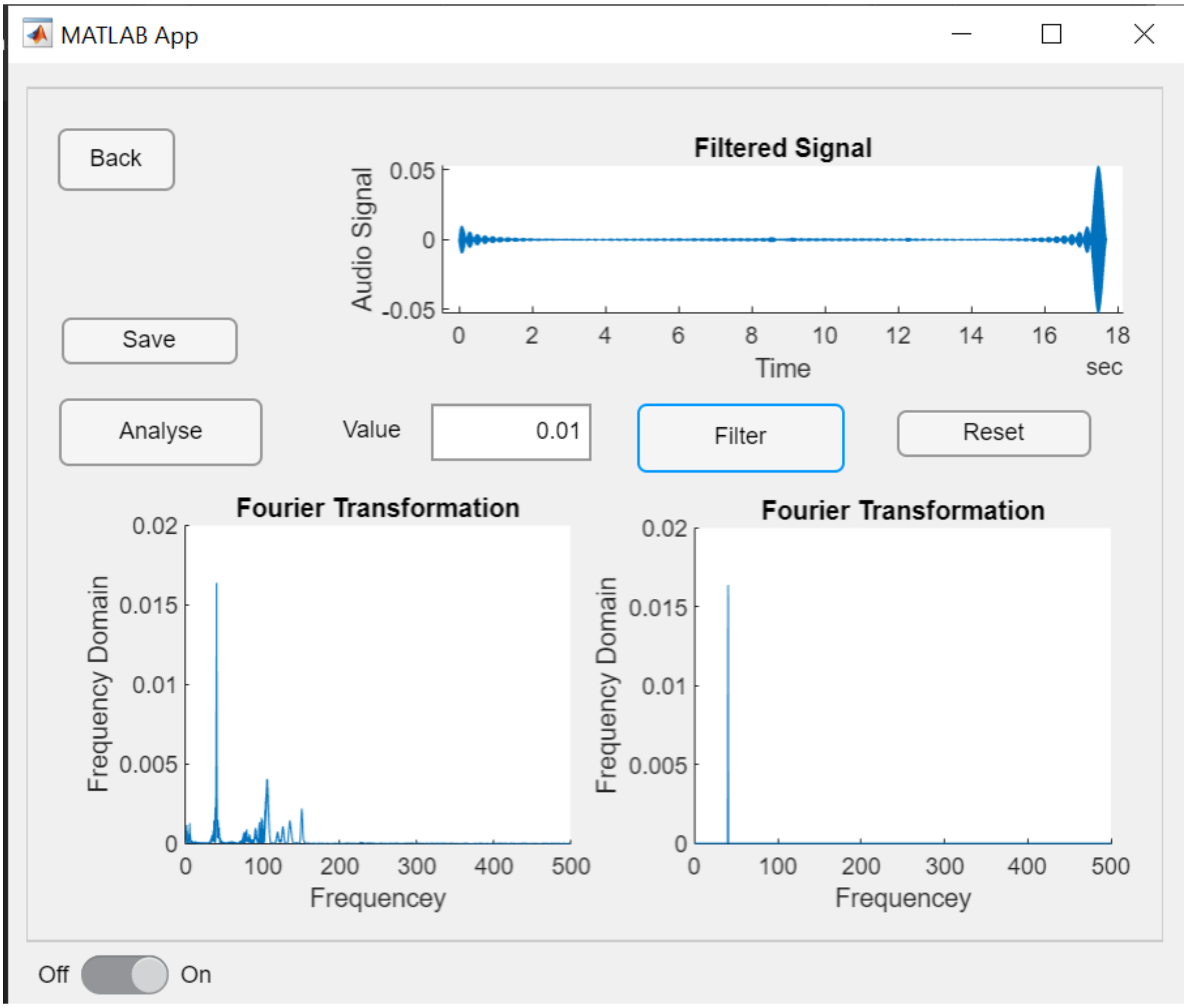
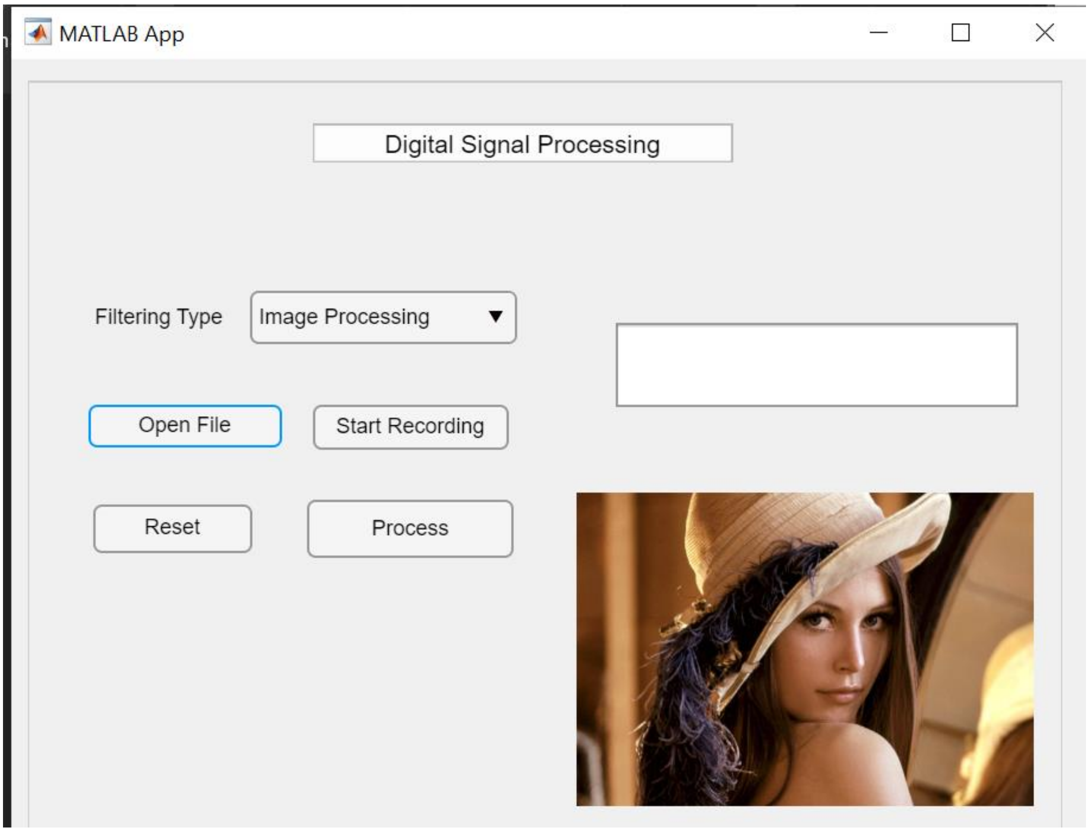
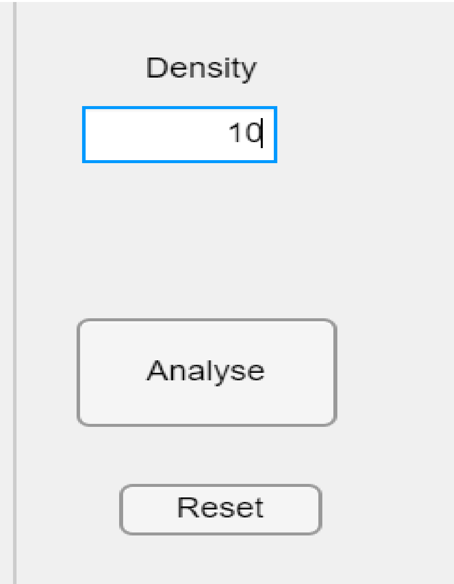


IMAGE PROCESSING:

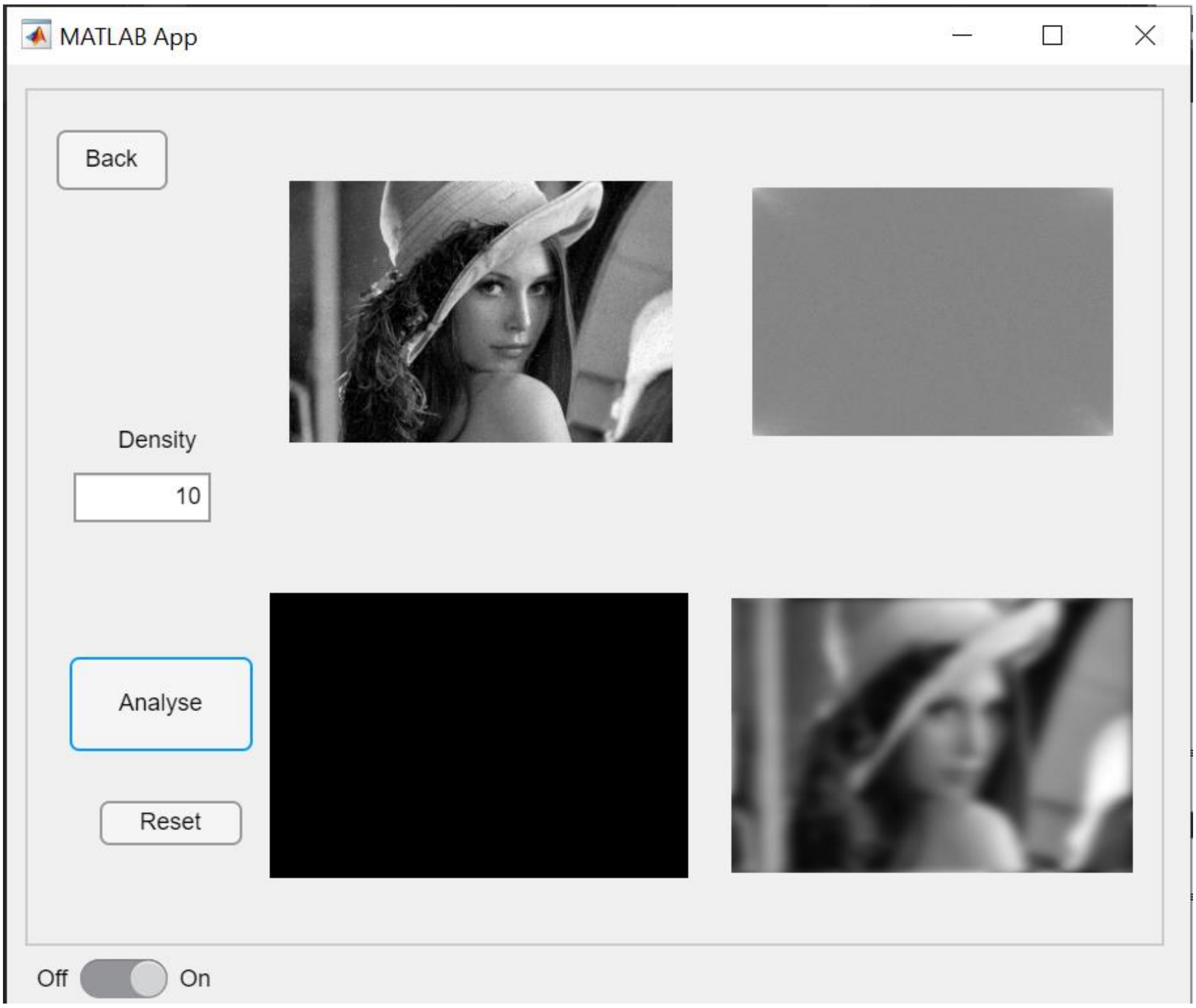
Step1: Open the Image and click on process



Step 2: Select the density



Step 3: Click on analyse



5) Conclusion:

We have presented a simple denoising method which produces high quality results. Here a guide image is used to denoise the noisy image. Denoising takes place in two domains, spatial domain and frequency domain. A bilateral filter is used in spatial domain. The image is processed by block DFT, block IDFT and filtering in frequency domain and Magnitude domain. The Gaussian blur technique is particularly useful to filter images with a lot of noise, since the results of the filtering showed a relative independence on the noise characteristics, and strong dependence on the variance value of the Gaussian kernel. In fact, if the image has a high SNR, the use of the technique investigated here could worsen the image. The Gaussian blur is better used when the original image has a low SNR. Besides, although filtering images with a large variance in the Gaussian function blurs and worsens the image,

The sound sample was passed on a low pass filter where a new audio signal was obtained after noise reduction, the low pass filter that is used to reduce the noise from the signals with the different frequency and ripple factor. These filters perform the filtering operations by the computation of difference equations. The coefficient equation and function as the low pass filter are given.

Thus, DFT has many applications, in this report we have shown some of its application.

6) References

https://en.wikipedia.org/wiki/Discrete_Fourier_transform

https://www.sciencedirect.com/topics/engineering/discrete-fourier-transform#:~:text=Applications%20of%20the%20DFT&text=First%2C%20the%20DFT%20can%20calculate,with%20this%20type%20of%20encoding.

https://learnemc.com/time-frequency-domain

https://en.wikipedia.org/wiki/Signal_processing#:~:text=Signal%20processing%20is%20an%20electrical,%2C%20images%2C%20and%20scientific%20measurements.

https://www.mathworks.com/help/images/noise-removal.html

https://in.mathworks.com/help/matlab/import_export/record-and-play-audio.html

https://www.cs.auckland.ac.nz/courses/compsci773s1c/lectures/ImageProcessing-html/topic1.htm

https://www.sciencedirect.com/topics/engineering/discrete-fourier-transform

https://www.youtube.com/watch?v=ey68GQTiWIE&ab_channel=Dr.AjayKumarVerma