APPLICATION OF FOURIER TRANSFORM ON IMAGE PROCESSING

A Thesis

Submitted in partial fulfillment of the requirements for the award of the Degree of

BACHELOR OF SCIENCE

IN

MATHEMATICS AND COMPUTING

BY

SANDEEP SAURAV

(IMH/10014/21)

DEPARTMENT OF MATHEMATICS

BIRLA INSTITUTE OF TECHNOLOGY

MESRA-835215,

RANCHI

APPROVAL OF THE GUIDE

Recommended that the thesis entitled **"APPLICATION OF FOURIER TRANSFORM ON IMAGE PROCESSING"** presented by **Mr. Sandeep saurav** under my supervision and guidance be accepted as fulfilling this part of the requirements for the award of Degree of Master of science.

To the best of my knowledge, the content of this thesis did not form a basis for the award of any previous degree to anyone else.

Date: 06/04/2024

(**Dr. Syeda Darakhshan Jabeen)**
Assistant Professor
Department of Mathematics
Birla Institute of Technology Mesra,
Ranchi-835215.

DECLARATION CERTIFICATE

I certify that

a) The work contained in the thesis is original and has been done by myself under the general supervision of my supervisor.
b) The work has not been submitted to any other Institute for any other degree or diploma.
c) I have followed the guidelines provided by the Institute in writing the thesis.
d) I have conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute.
e) Whenever I have used materials (data, theoretical analysis, and text) from other sources, I have given due credit to them by citing them in the text of the thesis and giving their details in the references.
f) Whenever I have quoted written materials from other sources, I have put them under quotation marks and given due credit to the sources by citing them and giving required details in the references.

SANDEEP SAURAV

IMH/10014/21

CERTIFICATE OF APPROVAL

This is to certify that the work embodied in this thesis entitled **"APPLICATION OF FOURIER TRANSFORM ON IMAGE PROCESSING"**, is carried out by **Sandeep saurav (IMH/10014/21)** has been approved for the degree of **Bachelor of Science in Mathematics and computing** of Birla Institute of Technology, Mesra, Ranchi.

Date: 06/05/2024

Place: Ranchi

Dr. Syeda Darakhshan Jabeen

(Internal Examiner) (**External Examiner)**

Dr. Soubhik Chakraborty

(Chairman & Head of Department)

ABSTRACT

In several areas of mathematical physics and engineering mathematics, the Fourier transform and related computer methods, including the fast Fourier transform (FFT), have been essential. The fundamental idea put forth by Joseph Fourier in the early 19th century—that functions can be represented as a linear combination of sine and cosine basis functions—lays the foundation for various transformations. This ground-breaking concept led to the creation of operator theory, approximation theory, and Hilbert spaces, which in turn produced the FFT algorithm, the mainstay of computer mathematics.

The FFT has enabled effective and real-time signal processing, image and audio compression, worldwide communication networks, and sophisticated data analysis, revolutionizing several disciplines during the past 200 years. But as these fields' challenges have become more complicated, specialized bases have arisen to better handle a variety of computational geometries and signal processing tasks, like the singular value decomposition (SVD) and wavelets.

This dissertation delves into the multifaceted applications of Fourier and wavelet transforms, highlighting their pivotal roles in modern mathematics and engineering. It explores how these transformations facilitate the manipulation of equations into coordinate systems where expressions simplify, decouple, and become more amenable to computation and analysis. The dissertation also examines the impact of these transformations across various domains, including data analysis, dynamical systems, and control, underscoring their significance in advancing scientific and technological frontiers.

ACKNOLEDGEMENT

I want to sincerely thank Dr. Syeda Darakhshan Jabeen, my project guide, for all of his help and assistance with my thesis. I gained a lot from working under his direction. I owe his efforts for helping me have a thorough understanding of this topic and a keen interest in learning more. For the duration of my studies at BIT, Mesra, Ranchi, his inspiration, encouragement, and support have been essential.

I would like to express my profound appreciation to Dr. Soubhik Chakraborty, Head of the Mathematics Department at BIT, Mesra, Ranchi, for providing me with the many resources I needed to finish my thesis. Additionally, I would want to express my gratitude to all the mathematics department faculty members who have assisted with the project, whether directly or indirectly. Additionally, I want to thank my friends at BIT, Mesra, Ranchi, and all the staff—technical and non-technical—who have been a huge assistance to me during the course.

Lastly, I would like to convey my sincere gratitude to my parents for their unwavering support and constant encouragement over the years I studied. Without them, this achievement would not have been feasible.

I apologize and sincerely thank everyone who has helped me but whose names I have not been able to mention.

Thank you.

DATE: 06/04/2024

Sandeep saurav

(IMH/10014/21)

CONTENTS

1. Introduction

The Fourier transform is a fundamental tool in image processing, providing deep insights into the spatial frequency content of images and opening a wide range of transformative applications. This revolutionary tool, which has its roots in the graceful mathematics of Fourier series, approximates any periodic function as the sum of sinusoidal functions, revealing the essential idea that complicated phenomena may be broken down into smaller, more manageable parts.

1.1. Fourier Series and Discrete Fourier Transform

When it was first developed, the Fourier series helped to clarify periodic phenomena by representing them as the total of sinusoidal functions. A new era in the representation of functions with a finite set of discrete values was made possible by its extension to discrete sets of points. This is the fundamental idea behind the Discrete Fourier Transform (DFT), which is the process of converting a finite series of sampled data points into its frequency domain equivalent and revealing the underlying frequency components that are encoded inside.

1.2. Fourier Transform and Its Application in Image Processing

The use of the Fourier transform has become ubiquitous with the development of digital imaging, providing deep insights into the spatial frequency composition of images. An extensive range of image processing applications are made possible by the Fourier Transform (FT), which is an extension of the DFT to continuous functions and a potent tool for examining the frequency content of images.

1.3. Methodology of Fast Fourier Transform (FFT)

Fast Fourier Transform (FFT): An innovative approach that transformed Fourier transform computing is essential to the usefulness of Fourier analysis in picture processing. The FFT significantly lowers the computing complexity of the DFT, making it practical for large-scale and real-time image processing applications by taking use of symmetries and redundancies in the computation of complex exponentials.

1.4. Applications in Image Denoising and Edge Detection

The Fourier transform's uses in edge detection and denoising demonstrate its versatility in image processing. The utilization of frequency domain representation in images allows for the efficient reduction of noise using frequency filtering techniques, as well as the identification and enhancement of edges and contours via frequency gradient analysis.

We explore the many facets of the Fourier transform and its numerous image processing applications in this dissertation. We seek to reveal the revolutionary potential of Fourier analysis in interpreting the complex tapestry of digital images through an in-depth investigation of Fourier series, discrete Fourier transform, Fourier transform, and its applications in image denoising and edge detection.

2. **Mathematical background**

Before discussing how to compute the Fourier transforms on data vectors, the analytic Fourier series and Fourier transform, which are defined for continuous functions, are introduced. Naturally, the discrete and continuous formulations should coincide in the limit of data with infinitely fine resolution. The Fourier series and transform have a tight relationship with the geometry of infinite-dimensional function spaces, or Hilbert spaces, which expand on the idea of vector spaces to include functions with an unlimited number of degrees of freedom. Let us begin with a summary of function spaces.

2.1. Inner products between vectors and functions

In this section, we will use inner product and function norms. Specifically, we will use the common Hermitian inner product for functions f(x) and g(x) defined for x on a domain [a, b] :

$$b(x) = \int_a^b f(x)g(x)\ dx$$

where g bar denotes the complex conjugate.

As the sampling resolution is raised, we want the vector inner product to converge to the function inner product. The data vectors g = [g1, g2.., gn]^ T and f = [f1, f2, fn]^ T are inner products. T is explained as follows:

$$\langle f, g \rangle = \sum_{k=1}^{n} f_k g_k$$

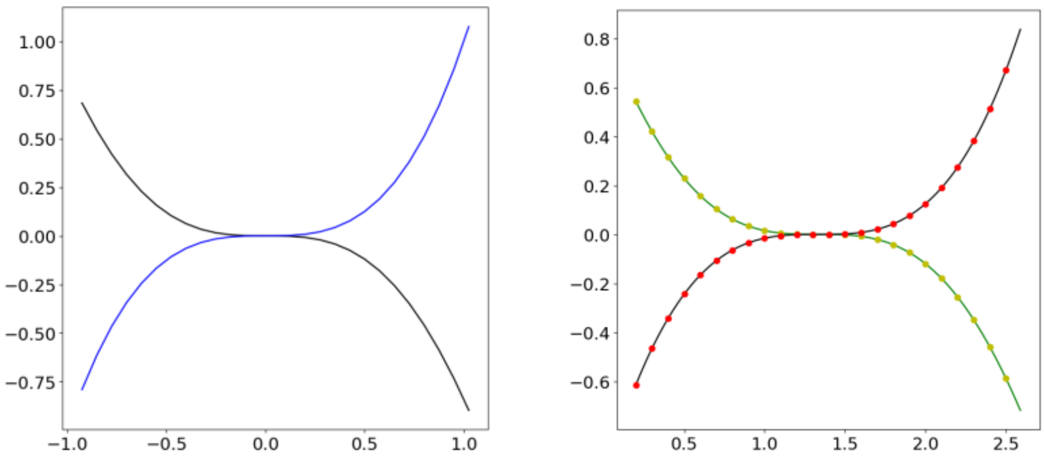


Fig 2.1: Representation of continuous function as discrete set of points

2.2. **Fourier transform**

The Fourier series for periodic functions is defined in such a way that the function repeats itself infinitely outside the definition's domain. As can be seen in Fig. 2.6, the limit of a Fourier series as the length of the domain is effectively what the Fourier transform integral represents. approaches infinity. This enables us to create a function defined on () without repeating. The Fourier series on the domain x [L, L] will be examined, and L -> infinity will then be allowed. The Fourier series on this domain is:

$$f(x) = a_0 + \sum_{k=1}^{\infty}(a_k \cos k\, x + b_k \sin k\, x) \; = \sum_{k=1}^{\infty} D_k e^{ikx}$$

with the coefficients given by:

$$\mathrm{D_k} = \frac{1}{2\mathrm{L}}\langle \mathrm{f(x)}, \psi_{\mathrm{k}}\rangle = \frac{1}{2\mathrm{L}}\int_{-\mathrm{L}}^{\mathrm{L}} \mathrm{f(x)e^{-jk\pi x/L}dx}$$

The Fourier transform pair is another name for these two integrals.

$$f(\omega) = \mathcal{F}\big(f(x)\big) = \int_{-\infty}^{\infty} f(x)e^{-i\omega x}\, dx$$

$$f(x) = \mathcal{F}^{-1}\left(\tilde{f}(\omega)\right) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \tilde{f}(\omega)e^{i\omega x}\, d\omega$$

2.3. Parseval's theorem

The energy of a signal in the spatial domain is equal to the energy of its Fourier transform in the frequency domain, according to Parseval's theorem, up to a constant factor. In other words, the L2 norm, or the total energy, of a function is preserved under the Fourier transform operation. This property is closely linked to unitarity, ensuring that two functions' inner product stays the same before and after the Fourier transform. Consequently, Parseval's theorem is valuable for approximation and truncation purposes, allowing us to control and bound errors when truncating the Fourier series or filtering out certain frequency components. This feature is very helpful in signal processing and image compression applications, where it enables efficient analysis and manipulation of signals while preserving their essential characteristics

$$\int_{-\infty}^{\infty} |f(x)|^2 dx = \frac{1}{2\pi}\int_{-\infty}^{\infty} |F(\omega)|^2 d$$

2.4. The FFT (Fast Fourier Transform) and DFT (Discrete Fourier Transform)

For continuous functions f(x), we have up to now studied the Fourier series and Fourier transform. However, approximating the Fourier transform applied to discrete data vectors is required when computing or working with real data.

After discretizing the function f(x) at regular intervals, the resultant discrete Fourier transform (DFT) is effectively a discrete version of the Fourier series for vectors of data f = [f1, f2, f3,,fn] ^T; x as displayed in the graph above.

2.4.1. (DFT) Discrete Fourier transform

It is instructive to begin with the simplest DFT formulation, even though we will always use the FFT for calculations. The discrete Fourier transform formula is:

$$\mathrm{G_k} = \sum_{\mathrm{r}=0}^{\mathrm{n}-1} \mathrm{G_r} \mathrm{e}^{-(2\mathrm{i}\pi\mathrm{rk})/\mathrm{n}}$$

Moreover, the following provides the inverse discrete Fourier transform (inverse-DFT):

$$f_k = \frac{1}{n}\sum_{j=0}^{n-1} f_j e^{\frac{i2\pi jk}{n}}$$

This means that the DFT is a linear operator, or matrix, that transforms the frequency domain f datapoints to the datapoints in the f space:

$$\{f_1, f_2, \dots, f_n\} \quad \text{DFT =>} \quad \{\hat{f}_1, \hat{f}_2, \dots, \hat{f}_n\}$$

The DFT uses the sine and cosine functions with respect to a fundamental frequency as integer multiples, to represent the data for a certain number of points.

The DFT may be calculated through matrix multiplication.

The DFT matrix is a unitary matrix, and the Fourier coefficients for the input vector are contained in the output vector f. Given the complexity of the matrix, the output has a phase as well as a magnitude, both of which can be usefully interpreted physically.

2.5. Fast Fourier transform

The utilization of the Fourier Transform Discrete (DFT) matrix in image processing often entails a significant computational load, with a complexity of O(n^2) operations. However, the introduction of the FFT (fast Fourier transform) algorithm has revolutionized image processing, enabling a plethora of applications with markedly reduced computational requirements, scaling as O(n(log(n)). This advancement has had profound implications across various domains of image processing, including compression, filtering, and feature extraction.

Consider the transmission, storage, and decoding of image data to appreciate the immense benefits of the FFT. With digital images comprising millions of pixels, the computational complexity of traditional DFT calculations can be overwhelming. However, the FFT algorithm drastically reduces this complexity, making real-time image processing feasible. Moreover, the FFT's efficiency becomes particularly evident in image compression, where many image signals exhibit high compressibility in the Fourier transform domain.

In image compression, the FFT allows for the efficient representation of images by identifying and retaining only the significant frequency components while discarding negligible ones. This property enables the storage and transmission of compressed images more efficiently, as only essential information needs to be preserved. However, the rapid computation of the compressed Fourier signal necessitates the use of FFT and its inverse (iFFT) operations.

The fundamental notion of the FFT lies in its efficient utilization of the DFT, particularly when a power of two represents the quantity of data points. As an illustration, considering an image with dimensions n=1024×1024n=1024×1024, the DFT matrix F1024F1024 can be harnessed to streamline computations, facilitating swift and efficient image processing operations. Consequently, FFT libraries have become integral components of image processing software and hardware, enabling a wide range of applications in digital imaging.

$$\hat{\mathbf{f}} = \mathbf{F}_{1024}\mathbf{f} = \begin{bmatrix} \mathbf{I}_{512} & -\mathbf{D}_{512} \\ \mathbf{I}_{512} & -\mathbf{D}_{512} \end{bmatrix} \begin{bmatrix} \mathbf{F}_{512} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_{512} \end{bmatrix} \begin{bmatrix} \mathbf{f}_{\text{even}} \\ \mathbf{f}_{\text{odd}} \end{bmatrix},$$

Where I512 is the 512 X 512 identity matrix, D512 is provided by, f-even are the even index elements of f, and f-odd are the odd index elements of f.

$$\mathbf{D}_{512} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & \omega & 0 & \cdots & 0 \\ 0 & 0 & \omega^2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \omega^{511} \end{bmatrix}.$$

One can transform F(512) into F(256), which can then be transformed into F(128), and so forth.

2.6. 2D Fourier Transforms in image processing

While our analysis focused on one-dimensional signals, The wavelet transform as well as the Fourier transform may be easily applied to signals with greater spatial dimensions, like two- and three-dimensional signals. Image processing and compression have greatly benefited from both the Fourier and wavelet transforms, which makes them an excellent case study for further research into higher-dimensional transforms.

The two-dimensional Fourier transform of the data matrix X Rn m is obtained by first applying the one-dimensional Fourier transform to each row of the matrix and then to each column of the intermediate matrix. This sequential row-wise and column-wise Fourier transform is shown in the following figure. If the Fourier transforms of the rows and columns are performed in a different order, the result stays the same.

The two-dimensional FFT is beneficial for image reduction since many of the Fourier components are small and can be discarded without affecting the quality of the image. Consequently, only a small number of large Fourier coefficients need to be stored and sent.

2.6.2. Representation of Images in Computer

Images are commonly represented in computers as two-dimensional arrays or matrices of pixel values in the field of digital image processing. Every pixel in the picture represents an element in the matrix, and each pixel's hue or intensity is represented by a number. Each pixel value in a grayscale image usually has a value between 0 and 255, which indicates how intense the grayscale color is. Each pixel in a colour image is represented by a multiple of values that each correspond to a separate color channel's intensity (e.g., red, green, blue).

2.6.3. Discrete Fourier Transform of Image Matrices

Image matrices can be used to evaluate the frequency content of images in the frequency domain by using the Discrete Fourier Transform (DFT). The DFT converts the picture's spatial domain representation into its frequency domain equivalent by handling the pixel values of the image matrix as a discrete signal. The frequency spectrum that is obtained offers information about the texture, patterns, and structures of the image by displaying the spatial frequency components that are present in it.

Example Illustration

Consider a grayscale image represented by the following 3x3 matrix:

[50 100 150]

[200 150 100]

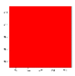
[50 25 75]

The frequency domain representation of this picture matrix can be obtained by applying the Discrete Fourier Transform to it. Different spatial frequency components may be detected in the resulting frequency spectrum, with dominant frequencies indicated by greater magnitude values and less prominent frequencies by lower magnitude values.

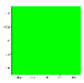
For example, the frequency spectrum may show peaks or patterns that match the edges, textures, or structures found in the original image after applying the DFT. We can learn more about the spatial properties of the image and possibly extract information that will be helpful for image processing tasks like texture analysis, edge recognition, and denoising by studying the frequency spectrum.

Example 1: # Red Green Blue pixels represented in computer

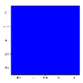
```
ar3 = np.zeros(shape=(1,1,3))
ar3[:,:,:] = [0,0,255]
plt.imshow(ar3)
```

Red 

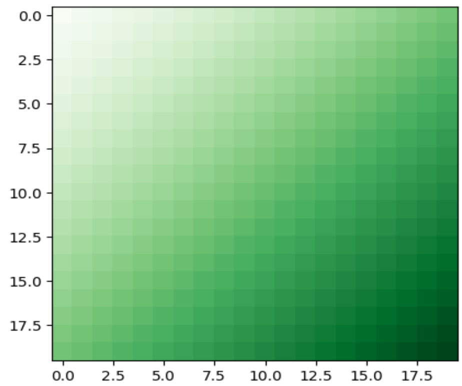
[[[255, 0, 0]]]

Green 

[[[0, 255, 0]]]

Blue 

[[[0, 0, 255]]]




```
limit = 20
x1 = np.linspace(1,100,limit)
y1 = np.linspace(1,100,limit)
x1,y1 = np.meshgrid(x1,y1)
z1 = x1+y1
plt.imshow(z1,cmap="Greens")
```

Fig 2.6.2: Representation of 20x20 pixels with varying intensity

3. Fourier series approximation for functions

In many practical scenarios, functions are represented by a discrete set of values along the x-axis. This discrete representation necessitates the use of discrete Fourier series for function approximation. In this section, we will discuss the process of approximating a discrete function f(x)f(x) using the Fourier series.

3.1. Discrete Function Representation

Let x be a discrete set of values representing the x-axis, denoted as x={x0,x1,x2,…,xN}, where N is the total number of data points. Similarly, let f(x)f(x) be a discrete set of values representing the function to be approximated, denoted as f(x)={f(x0),f(x1),f(x2),…,f(xN)}, f(x)={f(x0),f(x1),f(x2),…,f(xN)}.

3.1.1. Discrete Fourier Series Approximation

For discrete functions represented by a finite set of points, we approximate the function using a truncated Fourier series. The approximation involves finding the Fourier coefficients A0A0, Ak, and Bk that best represent the given function.

1. Determine Average Value (A_0):
   - The average value $\boldsymbol{A_0}$ is calculated as the mean of the function values f(x):

$$A_0 = \frac{1}{N}\sum_{i=0}^{N-1} f(x_i)$$

2. Compute Fourier Coefficients (A_k and B_k):
   - Fourier coefficients $\boldsymbol{A_k}$ and $\boldsymbol{B_k}$ are computed as follows:

$$A_k = \frac{2}{N}\sum_{i=0}^{N-1} f(x_i)\cos(k\omega_i x_i)$$

$$B_k = \frac{2}{N}\sum_{i=0}^{N-1} f(x_i)\sin(k\omega_i x_i)$$

   where $\omega_i = \frac{2\pi i}{N}$ is the angular frequency corresponding to the i-th data point.

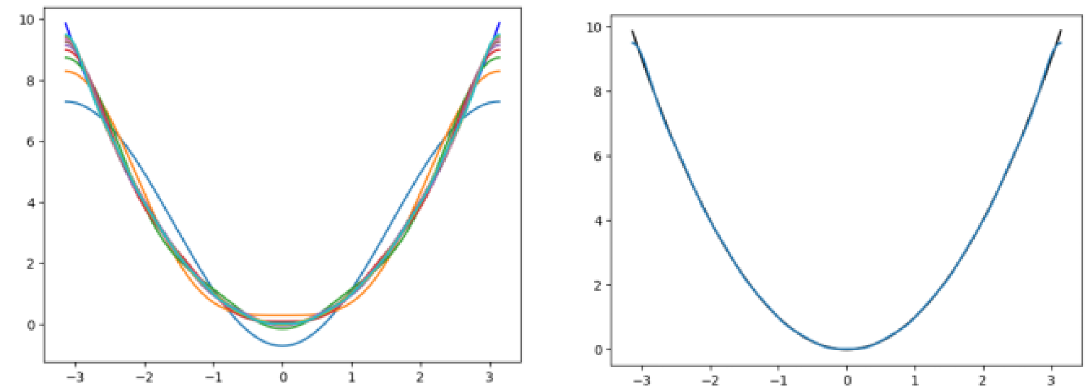
3. Truncate the Series:
   - To obtain a finite Fourier series approximation, the summation is truncated at a certain limit K, resulting in:

$$\mathrm{f(x)} \approx \frac{\mathrm{A}^0}{2} + \sum^{\mathrm{k=1k}} (\mathrm{A_k}\cos(\mathrm{k}\omega_o \mathrm{x}) + \mathrm{B_k}\sin(\mathrm{k}\omega_o \mathrm{x}))$$

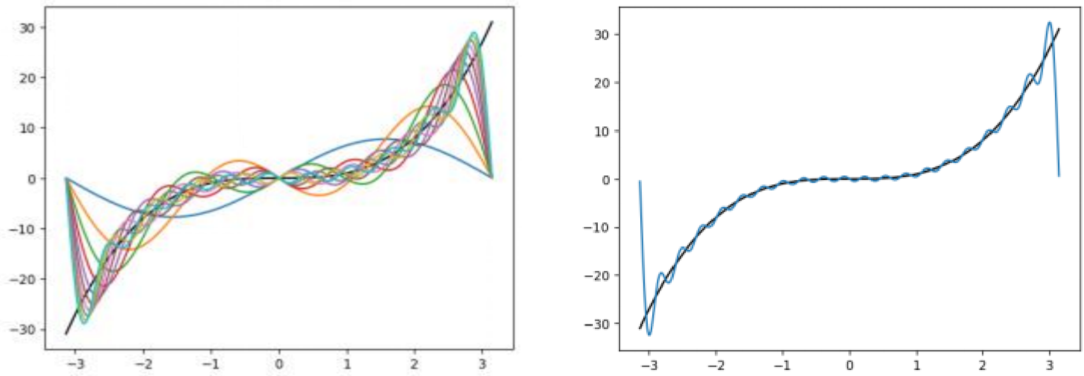
The accuracy of the Fourier series approximation depends on the number of terms included in the summation (i.e., the value of K). Increasing K results in a more accurate representation of the function f(x).

Example:

1. Fourier Series approximation x^2 of with k = 10



2. Approximating x^3using the Fourier Series with k = 20.



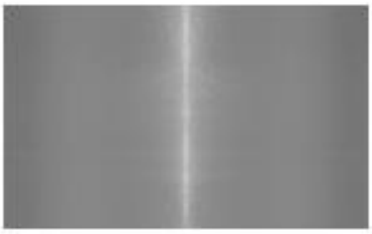
3.2. Frequency domain analysis of image using DFT

Fourier transformation of images involves applying the Discrete Fourier Transform (DFT) to the 2D matrix representation of the image to obtain its frequency domain representation. This transformation enables the analysis of the frequency content of the image and provides valuable insights into its spatial characteristics. By visualizing the magnitude spectrum, practitioners can identify patterns, textures, and structures present in the image, facilitating various image processing tasks such as filtering, enhancement, and feature extraction. The magnitude spectrum provides insights into the distribution of spatial frequencies in the image. Peaks or patterns in the spectrum correspond to prominent features, textures, and structures present in the image. For example, strong peaks may indicate the presence of edges, textures, or periodic patterns.

Example 1:

Original image: Grayscale

Fourier Transform of all rows



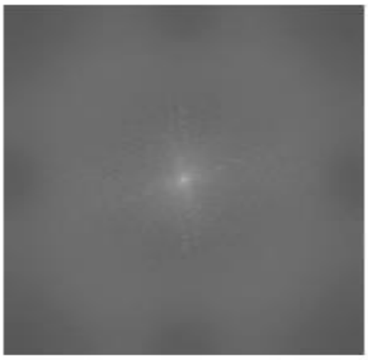
Fourier Transform of column

Example 2:

Original image: Grayscale

Row-wise Fourier Transform

Column wise Fourier



3.3 Compression of Images using Discrete Fourier Transform

Image compression using Fourier transform involves converting the image into its frequency domain representation using the Fourier transform. This transformation decomposes the image into its constituent spatial frequencies. By neglecting or quantizing the high-frequency components, which typically represent fine details and noise, and retaining only the dominant low-frequency components, significant compression can be achieved while preserving the essential features of the image. This process is justified by the Parseval's theorem, which states that the energy of a signal in the spatial domain is equal to the energy of its Fourier transform in the frequency domain. Therefore, by discarding the higher-frequency components with relatively low energy, the overall information content of the image remains largely unchanged. After the compression, the inverse Fourier transform is applied to reconstruct the compressed image, resulting in a lower-resolution version of the original image with reduced file size. This approach to image compression using Fourier transform is widely used in various applications, such as JPEG compression, where it enables efficient storage and transmission of digital images with minimal loss of visual quality.

Example:

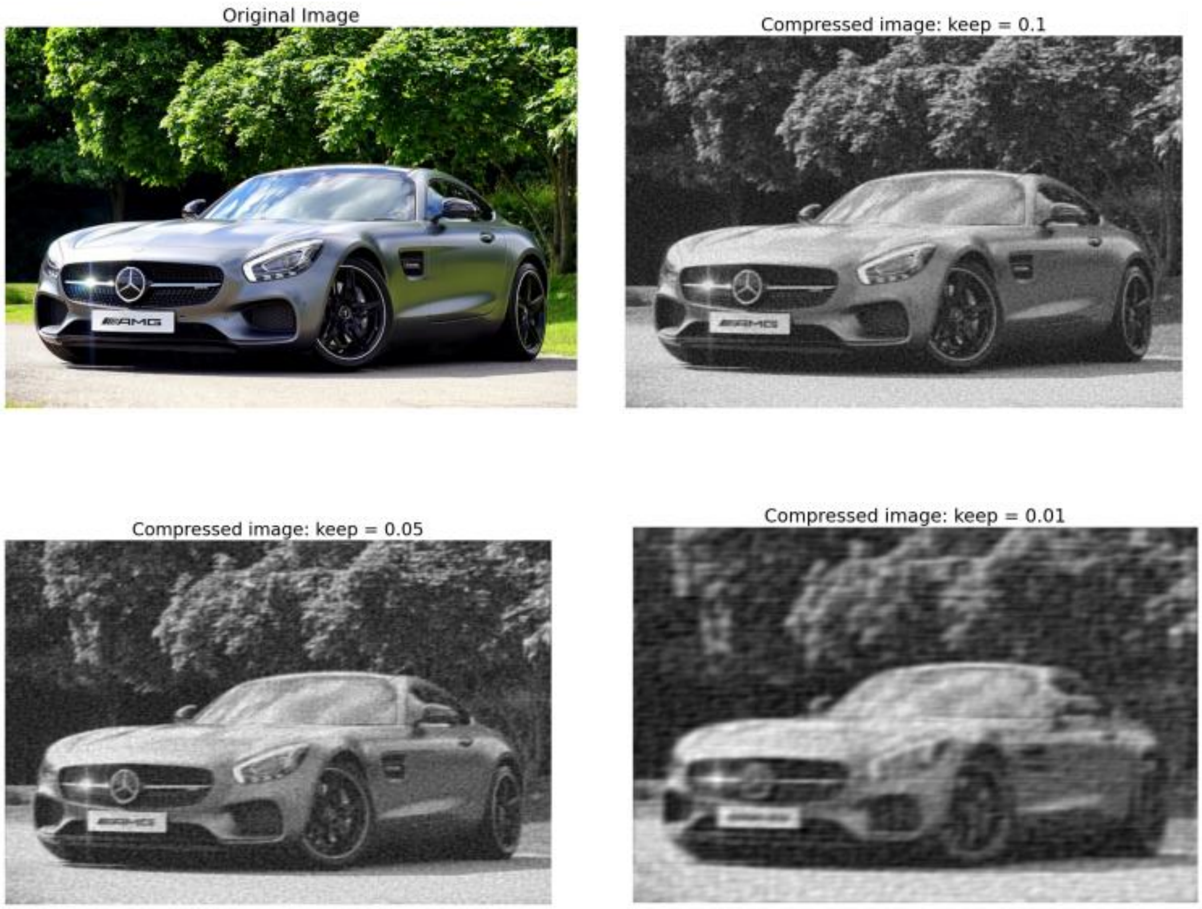


Fig: compressed image with several thresholds used to maintain the greatest Fourier coefficients at 10%, 5%, and 1%.

3.4. Denoising Data Using Fourier Transformation

In data analysis and signal processing, noise is often present in measured or observed data, which can obscure underlying patterns and affect the accuracy of analyses. Denoising, the process of removing or reducing noise from data, is crucial for enhancing signal clarity and improving the reliability of subsequent analyses. One effective approach for denoising data is through Fourier transformation, leveraging the frequency domain representation of signals.

3.4.1 Fourier Transform for Denoising

The Fourier transform decomposes a signal into its constituent frequency components, providing insights into the dominant frequencies present in the data. In the frequency domain, noise typically manifests as high-frequency components, while the signal of interest is often characterized by low-frequency components. By selectively filtering out or attenuating high-frequency noise components, it is possible to enhance the signal-to-noise ratio (SNR) and improve data quality.

3.4.2. Denoising Techniques

Several denoising techniques based on Fourier transformation can be employed to mitigate noise in data:

1. **Frequency Domain Filtering**: In this approach, a frequency domain filter is applied to the Fourier transformed data to suppress high-frequency noise components while retaining the essential signal frequencies. Common filters include low-pass, high-pass, and band-pass filters, which selectively allow certain frequency ranges to pass while attenuating others.
2. **Thresholding**: Thresholding techniques involve setting a threshold value and zeroing out or reducing Fourier coefficients below this threshold. By eliminating or reducing coefficients associated with noise, thresholding effectively suppresses noise while preserving signal information.
3. **Wavelet Denoising**: Wavelet transformation, which decomposes signals into both time and frequency domains, can also be utilized for denoising. By thresholding wavelet coefficients in the frequency domain, wavelet denoising techniques effectively suppress noise while preserving signal features.

3.4.3 Advantages of Fourier-Based Denoising

Utilizing Fourier transformation for denoising data offers several advantages:

- **Frequency Selectivity**: Fourier transformation enables selective filtering of noise based on its frequency characteristics, allowing for targeted noise suppression while preserving signal integrity.
- **Efficiency**: Fourier-based denoising techniques can be computationally efficient, particularly when implemented using FFT algorithms, making them suitable for real-time applications and large datasets.
- **Versatility**: The flexibility of Fourier-based denoising techniques allows for customization and adaptation to diverse datasets and noise characteristics, making them applicable across various domains.

Denoising data using Fourier transformation is a powerful approach for enhancing signal clarity and improving data quality. By leveraging the frequency domain representation of signals, Fourier-based denoising techniques enable selective noise suppression while preserving essential signal features. These techniques find widespread applications in diverse fields, including signal processing, image analysis, and time series analysis, facilitating more accurate and reliable data interpretation and analysis.

Example: Consider an example for input data of sum of two different frequencies of 100Hz and 240Hz represented in green color. Then we add some random noise in the data as represented in yellow color. To analyze frequencies domain we first converted take their Fourier transform as plotted if figure 2.3.b. We can see that most of the frequencies belongs to two values i.e. 100 and 240. So, we can filter noises from the data by setting some threshold and ignoring all frequencies below that threshold. Then inverse transform the data into spatial domain and we get clean data only.

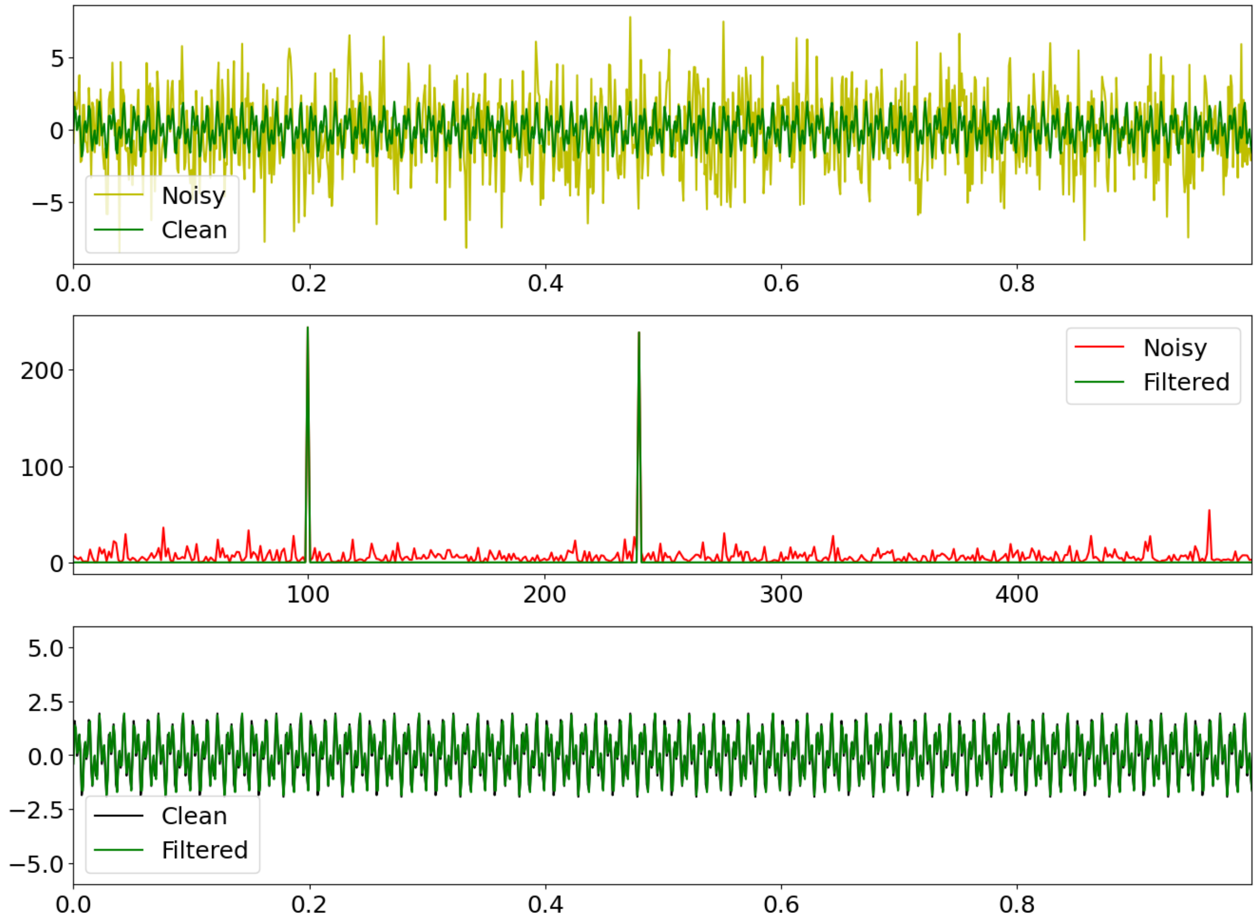


Fig 2.3: (a) Representing clear data in green added with the noisy frequencies represented in yellow (b) Fourier transform of the data representing frequencies. (c) Filtered data after removing data below some threshold and taking inverse transform.

3.5. Filtering of Images Using Fourier Transform

Filtering is a fundamental operation in image processing that aims to enhance or extract specific features from images. Fourier transform-based filtering techniques offer a powerful approach for manipulating the frequency content of images, allowing for a wide range of filtering operations.

3.5.1. Fourier Transform and Frequency Domain Representation

The Fourier transform decomposes an image into its constituent spatial frequencies, revealing information about patterns, textures, and structures present in the image. By representing an image in the frequency domain, it becomes possible to analyse and manipulate its frequency content.

3.5.2. Frequency Domain Filtering

Filtering in the frequency domain involves modifying the frequency components of an image to achieve desired effects. This is typically done by applying a frequency domain filter, which is a function that alters the magnitude and/or phase of specific frequency components.

- **Low-pass Filtering**: Low-pass filters attenuate high-frequency components while preserving low-frequency components. This is useful for smoothing or blurring images, reducing noise, and removing fine details.
- **High-pass Filtering**: High-pass filters suppress low-frequency components while accentuating high-frequency components. This is effective for sharpening images, enhancing edges, and detecting fine details.
- **Band-pass Filtering**: Band-pass filters selectively pass a range of frequencies while attenuating others. This can be used for isolating specific frequency bands corresponding to features or textures in the image.

3.5.3. Filtering Process

The process of filtering an image using Fourier transform involves the following steps:

1. **Compute the Fourier Transform**: Transform the input image from the spatial domain to the frequency domain using the Fourier transform.
2. **Apply the Frequency Domain Filter**: Multiply the Fourier transform of the image by the frequency domain filter to modify the frequency components according to the desired filtering operation.
3. **Inverse Fourier Transform**: Transform the modified frequency domain representation back to the spatial domain using the inverse Fourier transform to obtain the filtered image.

3.5.4. Advantages and Considerations

- Fourier transform-based filtering techniques offer several advantages, including the ability to analyse and manipulate the frequency content of images, flexibility in designing filters, and efficiency in processing large images.

- However, it is important to consider potential artifacts such as ringing and aliasing introduced by Fourier transform-based filtering, as well as the trade-offs between spatial and frequency domain representations in terms of computational complexity and interpretation.

4. Conclusion

In conclusion, Fourier transform-based filtering provides a powerful framework for manipulating the frequency content of images, allowing for a wide range of filtering operations such as low-pass, high-pass, and band-pass filtering. By leveraging the frequency domain representation of images, practitioners can effectively enhance, extract, or suppress specific features and textures, enabling various image processing tasks including noise reduction, edge enhancement, and feature detection.

Fourier-based denoising techniques offer versatile solutions for mitigating noise in visual data. By selectively suppressing high-frequency noise components while preserving essential signal features, Fourier transformation enables the enhancement of signal-to-noise ratios, thus improving the overall quality of visual data. These denoising techniques find wide-ranging applications in various domains, including medical imaging, remote sensing, and surveillance, where accurate interpretation and analysis of visual data are essential.

Fourier-based filtering methods, such as high-pass and low-pass filters, provide effective means for enhancing or suppressing specific frequency components in images. High-pass filters are commonly used for edge detection and feature extraction, allowing for the selective enhancement of high-frequency components associated with image edges and details. Conversely, low-pass filters are utilized for smoothing and noise reduction, attenuating high-frequency noise components while preserving low-frequency image features. These filtering techniques play a crucial role in image enhancement, facilitating tasks such as image restoration, sharpening, and segmentation, and find widespread applications across various fields, including multimedia processing, computer vision, and pattern recognition.

REFERENCES:

1. Oppenheim, A. V., and Alan S. Willsky. "Signals & systems." Pearson Education India, 2015. (Classic textbook covering Fourier series and Fourier transformation)
2. Bracewell, R. N. "The Fourier transform and its applications." McGraw-Hill Education, 2019. (Comprehensive coverage of Fourier transform theory and applications)
3. Gonzalez, Rafael C., and Richard E. Woods. "Digital image processing." Pearson Education India, 2017. (Standard textbook covering various aspects of digital image processing, including compression, denoising, and filtering)
4. Buades, Antoni, Bartomeu Coll, and Jean-Michel Morel. "A non-local algorithm for image denoising." 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)