

IMPLEMENTATION OF BI-CUBIC INTERPOLATION FOR SATELLITE IMAGES

Submitted in partial fulfilment of requirements for the award of Bachelor of Engineering in
Electronics and Communication Engineering

by

E.JOY NIKHIL (Reg no : 40130090)

G.JAYA SRI VASTAV (Reg no : 40130068)



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
SCHOOL OF ELECTRICAL AND ELECTRONICS**

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

CATEGORY-1 UNIVERSITY BY UGC

Accredited with Grade “A++” by NAAC | 12B status by UGC | approved by AICTE

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI – 600119

APRIL – 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

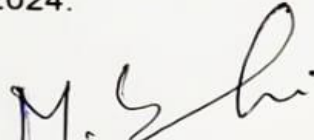
CATEGORY-I UNIVERSITY BY UGC

Accredited "A++" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **G. JAYA SRI VASTAV (40130068)**, **E JOY NIKHIL (40130090)** who carried out the project entitled "**Implementation of bi-cubic interpolation for satellite images**" under our supervision from Nov 2023 to March 2024.

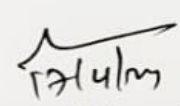

Internal Guide

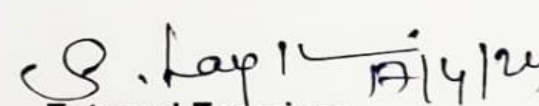
Dr. M SUMATHI, M.E, Ph.D.

Head of the Department


Dr.T. RAVI M.E., Ph.D.

Submitted for Viva voce Examination held on 17/04/2024


Internal Examiner


External Examiner

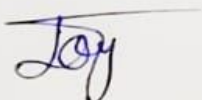
DECLARATION

I E JOY NIKHIL (40130090) hereby declare that the Project Report entitled "**Implementation of bi-cubic interpolation for satellite images**" done by us under the guidance of **Dr. M.SUMATHI, M.E, Ph.D.**, Professor is submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in Electronics and Communication Engineering.

DATE: 07/4/24

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATES

1. 
2. 

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to Board of Management of **SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our thanks to **Dr. N. M. NANDHITHA, M.Tech., Ph.D.**, Dean, School of Electronics and Communication Engineering and **Dr. T. RAVI M.E., Ph.D.**, Head of the Department, Department of Electronics and Communication Engineering for providing me necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our Project Guide **Dr. M.SUMATHI, M.E, Ph.D.**, for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of our project work.

We wish to express our thanks to all Teaching and Non-teaching staff members of the Department of Electronics and Communication Engineering who were helpful in many ways for the completion of the project.

ABSTRACT

This research investigates the best hardware implementation of bi-cubic heterogeneous interpolation. The bi-cubic method has been refined and reformed for FPGA execution. In signal processing, computer graphics, and other domains that employ discrete representations of continuous phenomena, a phenomenon known as aliasing can arise. It is the outcome of sampling or resolution limitations. One can witness image reconstruction. So in order to overcome this problem we use the technique in order to solve this issue it is used to make the image very clearly, by using the bi-cubic interpolation we can make the image to come to its original state. The process in this method of bi-cubic interpolation is taking the nearest sixteen pixel of the missing pixel by using the nearest pixels we can find the missing pixel in this way we can find each and every missing pixel and can bring the image to its first state. this method is more accurate than other methods or techniques like Linear interpolation, bi-linear interpolation. the outcomes from the bi-cubic is more accurate than the linear and bilinear techniques. Outcomes, which help to make the image more clear and accurate which is so helpful to gather any sort of information from the image .it is a three-step procedure consisting of image fusion, image registration, and image interpolation. Interpolating images is one of the most important procedures. The implementation results demonstrate the benefits of the suggested methods in terms of area, performance, and output quality. The algorithms are implemented in a Field Programmable Gate Array (FPGA) device utilizing a pipelined design.

INDEX

Chapter no	TITLE	PAGE NO:
	ABSTRACT	
1	LIST OF FIGURES	1
	INTRODUCTION	1
1.1	BICUBIC INTERPOLATION	1
1.2	IMAGE PROCESSING	2
1.3	OVERVIEW OF IMAGE PROCESSING	2
1.4	MOTIVATION	3
1.5	PROBLEM STATEMENT	3
1.6	OBJECTIVE	4
2	LITERATURE SURVEY	5
2.1	FPGA BASED ARCHITECTURE FOR BICUBIC INTERPOLATION THE BEST TRADE OF BETWEEN PRECISION AND HARDWARE RESOURCE	5
2.2	APPLICATION OF INTERPOLATION IMAGE BY USING BICUBIC ALGORITHM	5
2.3	BICUBIC INTERPOLATION COMPARED WITH OTHER INTERPOLATION	6
3	AIM AND SCOPE	7
3.1	PROPOSED METHODOLOGY	7
3.2	BI-CUBIC INTERPOLATION	7
3.3	PRINCIPLES OF BICUBIC	8
3.4	ENHANCED OF IMAGE QUALITY	9
3.5	ROBUSTNESS TO SCALING	9
3.6	SCOPE	10
4	MATERIALS AND METHODS	
4.1	EXISTING SYSTEM	11
4.2	DRAWBACKS OF EXISTING SYSTEM	13
4.3	PROPOSED SYSTEM	15
	4.3.1 ABOUT MATLAB	15

	4.3.2 INTERPOLATION	15
	4.3.3 LINEAR INTERPOLATION OR NEAREST PIXEL	16
	4.3.4 BILINEAR INTERPOLATION	17
	4.3.5 BICUBIC INTERPOLATION	18
	4.3.6 GREYSCALE	21
	4.3.7 BINARY DATA	22
	4.3.8 APPLICATIONS OF BINARY DATA	23
	4.6 FPGA	
5	RESULT AND DISCUSSION	28
6	CONCLUSION	30
	REFERENCES	36

LIST OF FIGURES

FIGURE No.	TITLE	PAGE
No.		
4.1(a)	MATLAB	12
4.1 (b)	MATLAB SIMULATIONS	13
4.3.1	MATLAB LOGO	16
4.3.2	INTERPOLATION EQUATION	15
4.3.3	LINEAR INTERPOLATION EQUATION	17
4.3.4(a)	BILINEAR INTERPOLATION EQUATION	18
4.3.4(b)	BILINEAR NUMERICAL REPRESENTATION TABLE	19
4.3.5	BICUBIC INTERPOLATION	20
4.3.6(a)	BICUBIC INTERPOLATION FORMULA	20
4.3.6(b)	GREYSCALE IMAGE	20
4.6	BINARY FILE	24
5(a)	PRIMARY TEST IMAGE	27
5(b)	SIGNAL TO NOISE RATIO TABLE	28
5(c)	TEST IMAGE OF LANDSCAPE	26
5(d)	TEST IMAGE FOR GLACIER	29
5(e)	TEST IMAGE FOR DESERTS	29
5(f)	GREYSCALE IMAGE OF DESERT	30
5(g)	GREYSCALE IMAGE OF GLACIER	30
5(h)	CONVERT CONVERTED DESERT INTO BINARY FILE	31

CHAPTER I

INTRODUCTION

1.1 BICUBIC INTERPOLATION

When it comes to digital image, precision and sharpness are essential for accurately expressing both content and beauty. However, a number of issues, including low resolution, compression artifacts, or incorrect focussing during capture, might cause photographs to appear blurry. Advanced interpolation techniques are used to overcome this difficulty. Of these, bicubic interpolation is a particularly advanced technique that is well-known for its capacity to improve image resolution without sacrificing important features. A mathematical technique called cubic interpolation is used to rebuild an image's pixel values by taking neighbouring data points into account inside a predefined neighbourhood. Bicubic interpolation considers a broader area than simpler interpolation techniques like bilinear interpolation, which interpolates between neighboring pixels linearly. This leads to smoother transitions and better preservation of image properties. The secret of bicubic interpolation is that, in contrast to linear interpolation techniques, it estimates pixel values using cubic polynomials, yielding results that are both more accurate and aesthetically beautiful. Through the examination of numerous adjacent pixels in both horizontal and vertical planes, bicubic interpolation efficiently minimizes artifacts and improves the image's overall sharpness and clarity. We explore the fundamentals, uses, and benefits of bicubic interpolation in image processing in this study. We examine its implementation strategies, theoretical foundations, and useful considerations for picture enhancing process optimization. Additionally, we address practical situations where bicubic interpolation is essential for addressing image blurriness, which leads to better visual quality and higher interpretability across a range of domains.

1.2 Understanding Image Blurriness

Image processing frequently deals with the problem of blurry images, which can be brought on by noise, low resolution, or motion blur. Images that are blurry are difficult to read and less visually appealing due to their lack of clarity and detail. Restoring clarity

and sharpness in image processing is essential for a number of uses, such as computer vision, medical imaging, and photography. Bi-cubic interpolation shows up as a potent technique to deal with image blurriness. When predicting the value of unknown pixels, bi-cubic interpolation takes a wider neighborhood of pixels into account than simpler interpolation approaches. Through the process of fitting a cubic polynomial to the surrounding pixels, bi-cubic interpolation yields more precise estimates of pixel intensities and smoother transitions. Bi-cubic interpolation yields more precise pixel intensity estimates and smoother transitions by fitting a cubic polynomial to the surrounding pixels. This method successfully lessens blurriness without sacrificing image details. The efficiency of bi-cubic interpolation resides in its capacity to reduce artifacts and yield excellent outcomes, which qualifies it for applications like image scaling and augmentation. Bi-cubic interpolation provides notable benefits over other interpolation techniques in terms of sharpness and image quality, despite its computational cost. As a result, it continues to be an important tool in the toolbox of image processing methods for reducing image blur and enhancing overall image quality.

1.3 Overview of Interpolation Techniques

Interpolation methods play a crucial role in image processing by filling in the gaps between discrete data points. Traditional interpolation techniques like nearest neighbor, bilinear, and bicubic interpolation are commonly used. While nearest neighbor is simple but produces jagged edges, bilinear interpolation offers smoother results by considering neighboring pixels. Smoother results are obtained by using cubic polynomials inside a local neighborhood through the use of bicubic interpolation. Piecewise polynomial functions are used in spline interpolation to reduce mistakes. From simple scaling to high-fidelity image reconstruction, each technique offers a range of alternatives to meet different image processing demands. The complexity and computing cost of each technique vary. This method yields results that are smoother and more accurate; it is especially helpful for upscaling images and removing distortion. Furthermore, by interpolating pixel values using piecewise-defined functions, spline interpolation techniques like cubic splines and cubic Hermite splines provide even more refinement. However, when it comes to enhancing image quality and reducing artifacts, bi-cubic interpolation stands out due to its higher-order polynomial fitting.

1.4 MOTIVATION

In the field of image processing, accuracy and clarity are critical goals. The basic human need for improved visual communication and comprehension is the driving force behind the use of bi-cubic interpolation to alleviate image blurriness. Clarity in images is crucial in a variety of sectors, from multimedia entertainment to medical imaging, as they are potent messengers of information, emotion, and artistic expression. Bi-cubic interpolation is a sophisticated technique that uses mathematical methods to interpolate pixel values and can recreate crisp, detailed images from fuzzy inputs. This method reflects a deliberate attempt to use technology's ability to improve visual representations, which is in line with our natural tendency toward progress and optimization. Furthermore, the drive is not limited to technical improvement alone; it is based on a dedication to improving human experiences via improved vision. Better decision-making, comprehension, and aesthetic enjoyment are all made possible by clearer visuals, which raises the standard of digital media engagement as a whole. Bicubic interpolation is ultimately motivated by the unwavering quest of perfection and the belief that each and every pixel has the capacity to communicate meaning and elicit emotion with unmatched clarity and accuracy.

1.5 PROBLEM STATEMENT

The issue statement asks for utilizing bicubic interpolation to reduce the blurriness in an image. A number of factors, including low resolution, noise, or distortion during image collection or processing, can cause blurriness in photos. Bicubic interpolation is a widely used technique in image processing that estimates pixel values based on neighboring pixels in order to improve image quality. However, there are a few issues that must be resolved in order to successfully apply bicubic interpolation. First, in order to get the desired results without adding artifacts or oversmoothing, it is imperative to determine the best parameters for the interpolation process, such as the size of the interpolation kernel and the degree of interpolation. Computational efficiency is also an issue because interpolation can be a computationally demanding procedure, especially when working with big or high-resolution photographs. To avoid losing crucial information from the image, color accuracy and the preservation of minute details must also be maintained during interpolation. Thus, the main goal of the problem statement is to create a bicubic

interpolation algorithm that is both effective and efficient in improving image sharpness while reducing computational overhead and artifacts. To produce high-quality, non-blurry photos, this entails managing computational resources, optimizing interpolation settings, and guaranteeing color fidelity and detail retention.

1.6 OBJECTIVE

The objective of employing bi-cubic interpolation is to improve the clarity and sharpness of images by effectively estimating the intensity values of unknown pixels. By applying a complex mathematical technique to improve the resolution of photographs, bi-cubic interpolation is used to resolve blurriness. Bi-cubic interpolation produces a smoother and more accurate reconstruction of the image by taking into account the gradients of the nearby pixels, in contrast to simpler interpolation techniques. By fitting a cubic function to the surrounding data points and examining a broader area around each pixel, bi-cubic interpolation reduces artifacts like pixelation and generates smoother transitions between pixels. The goal of this technique is to improve image quality for a range of uses, such as computer graphics, digital photography, and medical imaging. Through bi-cubic interpolation, images can be upscaled with minimal loss of detail and without introducing significant distortion. This technique plays a crucial role in image processing tasks where maintaining visual fidelity is essential. Ultimately, the objective is to provide viewers with clearer and more visually appealing images that accurately represent the original scene or subject matter.

CHAPTER 2

LITERATURE SURVEY

In this literature survey we have gone through few papers which were describing about bicubic and other related interpolation techniques, this survey helped us understand more about interpolation

2.1 FPGA - based architecture for bi - cubic interpolation: the best trade-off between precision and hardware resource consumption

The proposed architecture has been coded in VHDL and implemented on Intel and Xilinx targets (Cyclone II, Virtex II Pro, and Cyclone V). The required resources are provided by Quartus 13.1 and 17.1 for Cyclone II and V, respectively, and by Xilinx ISE 9.1 for Virtex-II Pro. To evaluate the proposed architecture, it is compared to the results given in [9–11, 13–15] in terms of the number of required DSPs, logic utilization, and maximum operating frequency. The proposed architecture has been coded in VHDL and implemented on Intel and Xilinx targets (Cyclone II, VirtexII Pro, and Cyclone V). The required resources are provided by Quartus 13.1 and 17.1 for Cyclone II and V, respectively, and by Xilinx ISE 9.1 for Virtex-II Pro. To evaluate the proposed architecture, it is compared to the results given in [9–11, 13–15] in terms of number of required DSP, logic utilization, and maximum operating frequency.

2.2 Application of Interpolation Image by Using Bi-Cubic Algorithm

Image zooming will change pixel density over the new mapping space that has been created, where the pixel size remains but the mapping space changes. Image magnification is done by increasing the size of the picture, this magnification will make a void pixel in the picture, and the zooming result looks fragmented. The zoom image results in a widening occurring with a width (W) and height (H) image can write as the formula below: $W = W \times 2$ $H = H \times 2$ The bi-cubic interpolation algorithm works by estimating the color in an image pixel by calculating the average of 16 pixels residing around pixels that are similar to pixels in the source image, and the formula below gives an interpolation result applied to each component of red, green and blue. The last presentation m and n have a 4 x 4 grid length around pixels (i, j). The experiment has been carried out using different interpolation methods. The PSNR values have

been compared for different methods and different images. It has been proved that bicubic interpolation gives much superior results.

2.3 DESIGN AND IMPLEMENTATION OF REAL-TIME OBJECT DETECTION FOR BLIND USING CONVOLUTIONAL NEURAL NETWORK

FPGA-based architectures for bicubic interpolation reveal different approaches aimed at achieving an optimal trade-off between accuracy and hardware resource consumption. Research in this area often focuses on developing efficient hardware implementations that can perform bicubic interpolation with high accuracy while minimizing resource consumption. Several studies have proposed new architectures that leverage FPGA-specific features such as parallelism, pipelining, and custom compute units to speed up interpolation computations. These architectures often use algorithms and data structures that are optimized for the reconfigurable nature of FPGAs, allowing efficient use of resources and maximizing performance. Additionally, researchers are exploring techniques to reduce accuracy requirements without sacrificing interpolation quality. This includes studying the effects of reduced-precision arithmetic and quantization on interpolation accuracy and developing methods to reduce the resulting errors. In this study, we develop an FPGA-based architecture that provides an optimal balance between bicubic interpolation accuracy and hardware resource consumption, with the ultimate goal of achieving high-quality interpolation results while optimizing FPGA resource utilization. Continuing efforts to design are emphasized.

Regarding the application of bicubic interpolation in image processing, many studies have been carried out investigating its effectiveness and applicability in various fields. Researchers have extensively studied the use of bicubic interpolation for tasks such as image resizing, enhancement, restoration, and geometric transformation. Many studies have focused on evaluating the performance of bicubic interpolation compared to other interpolation methods, demonstrating its ability to preserve detail, reduce artifacts, and produce visually appealing results. In fact, these evaluations often use quantitative metrics such as maximum signal-to-noise ratio (PSNR), structural similarity index (SSI), and mean squared error (MSE) to assess interpolation quality. Additionally, researchers are exploring applications of bicubic interpolation in specific fields such as medical imaging, satellite imagery, digital photography, computer

graphics, and video processing. They consider how to tune and optimize bicubic interpolation to meet each application's unique challenges and requirements, and often suggest new algorithms and techniques tailored to specific use cases

CHAPTER 3

AIM AND SCOPE

This project's goal is to precisely calculate, during image scaling or geometric modification, the intensity values of pixels at non-integer positions. When compared to other simpler interpolation techniques like nearest-neighbor or bilinear interpolation, this method seeks to yield a smoother and more aesthetically pleasing output. In order to accomplish this goal, cubic interpolation fits a cubic polynomial to estimate the intensity values close to the target point, taking into account a broader neighborhood of pixels surrounding the target point. Bicubic interpolation seeks to reduce aliasing artifacts, maintain fine details, and maintain overall image quality during resizing or geometric changes by combining more information from surrounding pixels.

3.1 PROPOSED METHODOLOGY

- By taking images from the satellites and keeping them into the method to resolve the issue.
- Frist taking the images from the satellites through nasa websites and downloading the images.
- And inserting the images that we got from the satellites into the method of bi-cubic interpolation.
- By using mat lab software we are performing the task and resolving the issue and the blurriness of the image.
- Later we convert that image into either binary code or other alternative method
- After that we convert the results that we got from the mat lab we will upload it to FPGA

3.2 Bi-Cubic Interpolation

One of the most effective methods in the field of image processing is bi-cubic interpolation, which is mostly used to solve the problem of image blurriness. This technique improves the sharpness and clarity of images by extrapolating pixel values from nearby pixels using a sophisticated mathematical procedure. Its breadth encompasses a wide range of applications, such as satellite imaging, digital photography, medical imaging, and more. A mathematical technique called the bicubic algorithm is used to interpolate values on a grid by taking into account the weighted average of nearby data points. Bicubic interpolation takes into account a larger set of

neighboring points than linear interpolation, which only analyzes adjacent points. This results in reconstructions that are smoother and more accurate. The process of bicubic interpolation is a little more difficult than that of bilinear interpolation. The sixteen closest neighbors of a pixel have been taken into consideration in bicubic interpolation. Using the equation, the intensity value given to point (x,y) is produced. The sixteen coefficients in the sixteen equations in sixteen unknowns can be expressed using the sixteen nearest neighbors of point (x,y) .

3.3 Principles of Bicubic Interpolation

Fundamentally, fitting a smooth curve through a collection of known data points is the process of bicubic interpolation. Cubic polynomials, which provide continuity and differentiability at the interpolation sites, are used to determine this curve. Bicubic interpolation determines the interpolated value based on a weighted average, so closer points have greater significance, by taking into account a 4x4 grid of nearby pixels. A 4x4 grid of nearby pixels encircling the target point is taken into consideration while applying the principles of bicubic interpolation. The interpolated value of each pixel is derived from its relative position inside the grid and its distance from the target point. Using a cubic convolution kernel, bicubic interpolation determines the weighted average of these nearby pixels, with closer points having a greater influence. The goal of the cubic convolution kernel design is to minimize artifacts while maintaining local image properties. This is accomplished by smoothing the transitions between adjacent pixels by applying a cubic function to the intensity values of each nearby pixel. Furthermore, bicubic interpolation offers flexibility in adjusting to various image properties and requirements by allowing the interpolation kernel size and boundary conditions to be changed.

3.4 Enhanced Image Quality

By successfully lowering blurriness, bi-cubic interpolation can greatly enhance image quality, which is one of its main applications. Sharper and crisper images are produced by precisely estimating pixel values based on surrounding data points. This technique restores small features and edges lost during image capture or processing. In domains

such as medical imaging, where accurate visualization is critical to diagnosis and therapy, this kind of improvement is vital. By lowering blurriness and increasing resolution, bicubic interpolation can improve image quality and increase visual fidelity. Images are processed by a method known as bicubic interpolation, which results in finer features being interpolated and reconstructed with more accuracy. This process produces sharper edges, smoother transitions, and overall increased clarity. This improvement is made possible by bicubic interpolation, which takes into account a wider range of pixels around each target point. It computes the interpolated values via weighted averaging with cubic polynomials, producing smoother curves and crisper pixel transitions. This technique avoids distortions like pixelation and jagged edges that are typically associated with simpler interpolation algorithms while efficiently preserving small details.

3.5 Robustness to Scaling

Bi-cubic interpolation's resilience to image scaling is a key additional benefit. This method preserves better image quality than more straightforward interpolation methods such as nearest-neighbor or bilinear interpolation, regardless of the size of the image. The capacity to scale is crucial for uses like digital zoom in cameras, where preserving image quality during zooming is necessary to preserve visual integrity. Maintaining the integrity of the image at different magnification levels. In contrast to less complex interpolation techniques, bicubic interpolation minimizes distortion and scaling artifacts by applying a smooth curve-fitting procedure over a broader neighborhood of pixels. This approach works well for activities like resizing and zooming images since it preserves small details and keeps the integrity of the image intact. In digital image processing, bicubic interpolation is still a dependable method for producing consistent, high-quality results because of its adaptability to various scaling factors and preservation of visual authenticity.

3.6 SCOPE

Bicubic interpolation applied to satellite images has great potential to address various challenges and opportunities in fields such as environmental monitoring, urban planning, agriculture, and disaster management. As technology continues to advance and satellite image processing capabilities improve, the future scope of bicubic interpolation in this field will be extensive and diverse. One important area for future development lies in the improvement and optimization of bicubic interpolation algorithms, especially for satellite imagery. Bicubic interpolation is already widely used for image resizing and correction, but adapting it to the unique characteristics and challenges of satellite imagery provides new opportunities for innovation. Future research efforts may focus on developing advanced interpolation models that can effectively handle the complex spatial and spectral characteristics of satellite images, such as different resolutions, atmospheric influences, and sensor artifacts. Additionally, integrating bicubic interpolation with other image processing techniques and machine learning approaches could further increase its usefulness in satellite image analysis. For example, deep learning techniques can be used to learn complex patterns and relationships in satellite imagery data, enabling more accurate and adaptive interpolation results. By combining bicubic interpolation with techniques such as image segmentation, object detection, and classification, researchers can explore new ways to analyze and interpret satellite images for a variety of applications. A promising avenue for future research is the development of scalable real-time bicubic interpolation solutions for processing large amounts of satellite image data. The increasing availability of high-resolution satellite imagery from satellite constellations and aviation platforms has increased the need for efficient image processing techniques that can process large data sets with minimal computing resources. Future advances in parallel computing, distributed systems, and cloud computing may enable the development of scalable bicubic interpolation algorithms that can process terabytes of satellite imagery data in near real time.

CHAPTER 4

MATERIALS AND METHODS

4.1 EXISTING SYSTEM

Matlab, an acronym for Matrix Laboratory, is a potent programming language and environment that's mainly utilized for data analysis, visualization, and numerical computation. It was created by MathWorks and is extensively used in many different fields, such as engineering, finance, biology, and higher education. Matlab's vast built-in function library, sophisticated mathematical problem-solving capabilities, and user-friendly syntax are the main sources of its adaptability. Matlab is a vital tool for both professionals and scholars in this digital age where making decisions based on data is crucial. Matlab is mostly good at manipulating matrices and doing calculations in linear algebra. It is especially useful for expressing and resolving mathematical problems because of its syntax, which is intended to mimic mathematical notation. Users may carry out a variety of tasks with Matlab, ranging from simple arithmetic to complex algorithms like eigenvalue and singular value decomposition, thanks to the program's effective handling of matrices. Many of Matlab's applications, such as signal processing, image processing, and control system design, are based on this fundamental feature. Matlab's vast collection of specialized toolboxes is one of its greatest assets. These toolboxes give users access to extra features and domain-specific algorithms that make it easier for them to solve challenging issues. For instance, the picture Processing Toolbox has tools for picture improvement, segmentation, and feature extraction, while the Signal Processing Toolbox has features for filtering, spectrum analysis, and waveform creation. By extending Matlab's capabilities beyond simple numerical computation, these toolboxes help users effectively tackle problems unique to their domains. Furthermore, Matlab has strong data analysis and visualization features. With its plotting features, users may construct fully configurable graphs and plots, including scatter plots, histograms, and 2D and 3D plots, to show data in a variety of formats. Users can customise the look and feel of their visualizations with interactive tools like the GUI Design Environment and Plot Editor. Furthermore, Matlab's statistical functions make it easier for users to do data analysis tasks like clustering, regression analysis, and hypothesis testing, enabling them to extract valuable insights from their data. Fig 4.1(a) shows the logo of matlab

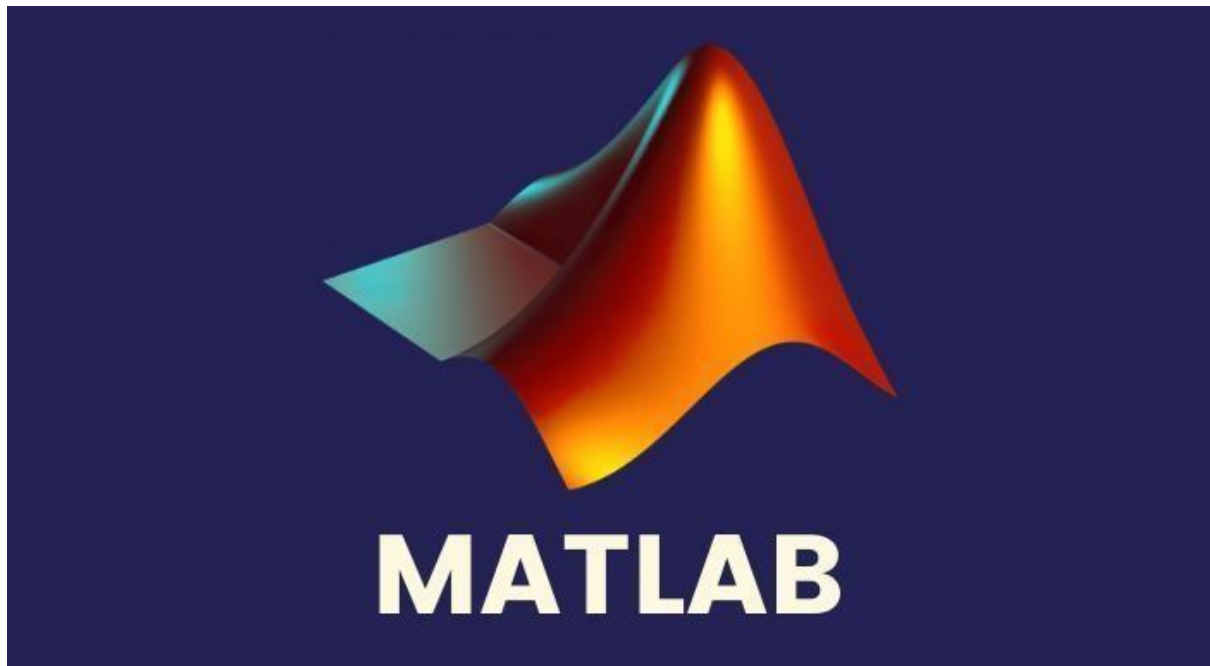


Fig 4.1(a) Matlab Symbol

Moreover, Matlab is a complete environment for creating and implementing programs and algorithms. Its integrated development environment (IDE) streamlines the development process by offering tools for authoring, profiling, and debugging Matlab code. Matlab's parallel computing capabilities can be used by users to speed up computationally demanding operations by utilizing several cores or clusters. Furthermore, Matlab provides alternatives for distributing algorithms to several platforms, such as web applications, embedded systems, and standalone executables, guaranteeing that users can distribute their solutions in a variety of settings. To sum up, Matlab is a strong and flexible programming language and environment that performs exceptionally well in data analysis, visualization, and numerical computing. Because of its large function library, specialized toolboxes, and simple syntax, it is a useful tool for scholars and professionals in many different fields. Matlab provides the tools and capabilities required to efficiently handle difficult tasks, whether they are related to designing algorithms and applications, analyzing data, or solving mathematical problems. Matlab is still a vital tool for people who want to use computational analysis and data-driven decision-making as technology develops. Fig 4.1(b) shows the simulation that can be done on matlab

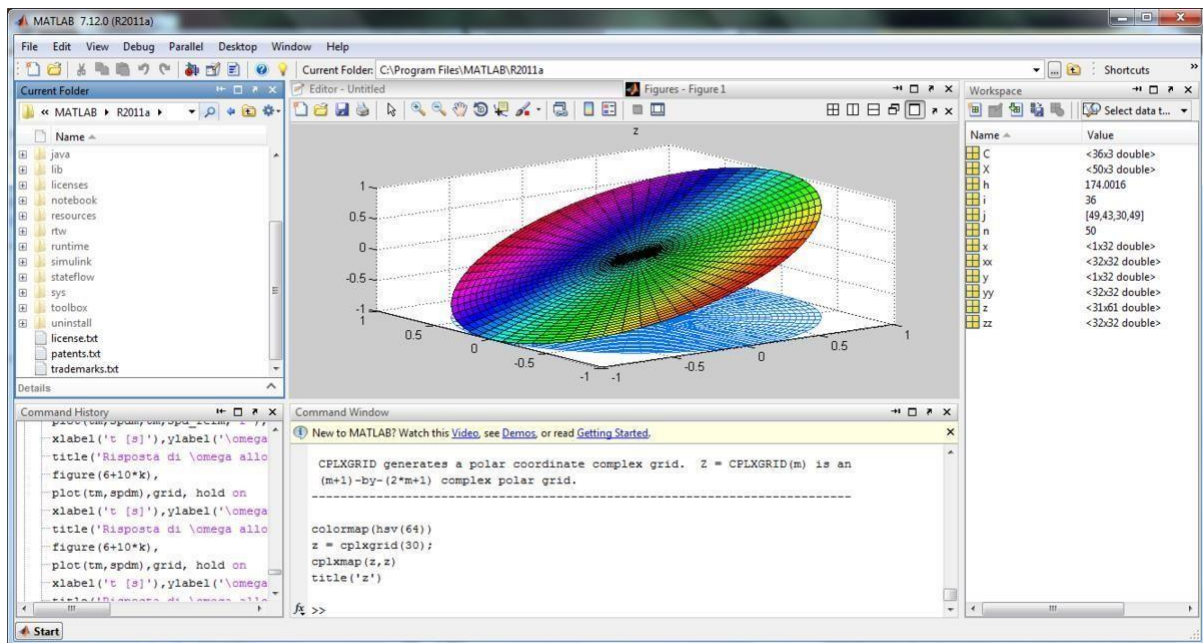


Fig 4.1(b) Simulation window of MATLAB

4.2 DRAWBACKS OF EXISTING SYSTEM

1. Performance: When working with huge datasets or computationally demanding activities, Matlab can occasionally experience performance problems. Even though the program is designed with numerical computations in mind, its performance might not always be as good as that of lower-level programming languages like C or Fortran. For customers who demand high-performance processing skills, this can be a major disadvantage.

2. Steep Learning Curve: Those who are unfamiliar with programming or numerical computing may find Matlab to have a steep learning curve. Novice users may need to devote time and effort to mastering the software's syntax and programming paradigm as they may be unfamiliar with it. For users who need immediate results or are used to other programming languages, this learning curve may be a hurdle.

3. Memory Management: When working with huge datasets or lengthy computations, Matlab's memory management mechanism can be ineffective. If

users do not carefully manage memory usage, they may experience issues linked to memory or performance deterioration. For users dealing with large-scale data analysis or simulations, this can be a major disadvantage.

3.Memory Management: When working with huge datasets or lengthy computations, Matlab's memory management mechanism can be ineffective. If users do not carefully manage memory usage, they may experience issues linked to memory or performance deterioration. For users dealing with large-scale data analysis or simulations, this can be a major disadvantage

4.Limited Scalability:Although Matlab works well for small-scale development projects and prototyping, it does not necessarily scale well to distributed computing systems or bigger applications. As users try to scale their Matlab programs to meet bigger workloads or distributed computing clusters, they may run across scalability problems or performance constraints.

5.Reliance on Toolboxes: Although Matlab has a large toolbox library, this can be a disadvantage for users who need specialized functionality that isn't included in the base installation. The cost and complexity of using Matlab can increase if users need to buy more toolboxes or discover other solutions to satisfy their needs.

6.Limited Community help: Compared to other open-source programming languages like Python or R, Matlab may not always offer the same degree of help or resources, despite having a big user base and active community forums. It might be harder for users to obtain community-contributed libraries and packages or discover answers to difficulties, which would limit their capacity to use outside resources for their projects.

7.Cost: The price of Matlab is among its biggest disadvantages. The program is commercial, and licenses can be pricey, particularly for small enterprises or lone users. For users on a tight budget, this high cost may be unaffordable, thus they may look for free or open-source alternatives.

8.Proprietary Nature: The source code for Matlab is not available to consumers since it is a proprietary piece of software. Users who want to know how the product

functions internally or who need customization may find this lack of openness to be an issue. It also restricts users' ability to expand or alter Matlab's capabilities to meet their own requirements.

9. Limited Support for Object-Oriented Programming: Although Matlab can handle object-oriented programming (OOP), it does not have as flexible or reliable an implementation as Python or Java. Users who extensively rely on OOP principles in their software development workflows may find Matlab's OOP features inadequate or difficult to utilize as a result, which might be a disadvantage.

4.3 PROPOSED SYSTEM

Our suggested methodology uses bicubic interpolation, which improves image resolution by interpolating values from nearby pixels, to address image blurriness. We hope to increase overall image quality by using this technology to sharpen photos and enhance visual clarity. Our technique is able to precisely recover details that have been lost because to blurring by using bicubic interpolation, which produces images that are brighter and more defined. Users will have a useful and effective technique to improve their visual material with this solution, which claims to solve the issue of blurriness in photographs.

4.3.1 MATLAB

High-level programming languages and interactive environments like MATLAB are frequently used for data analysis, visualization, and numerical computing. It is a flexible platform for researchers, engineers, and students since it provides an extensive range of built-in capabilities and toolboxes for diverse scientific and engineering applications. MATLAB is a sophisticated tool that allows users to produce interactive graphs and graphical user interfaces (GUIs), simulate dynamic systems, solve complex mathematical problems, and build algorithms. Its syntax is simple and intuitive. Professionals from a wide range of disciplines who are looking for effective solutions for their computational jobs use it because of its comprehensive documentation and vibrant community support.

4.3.2 INTERPOLATION

Interpolation is a fundamental technique used in various fields, including mathematics, computer science, engineering, and image processing. It involves estimating values between known data points to create a continuous function or representation. In simpler terms, interpolation fills in the gaps between existing data points. There are several methods of interpolation, each suited to different types of data and applications. Common interpolation techniques include linear interpolation, polynomial interpolation, spline interpolation, and bicubic interpolation. Linear interpolation connects two adjacent data points with a straight line, while polynomial interpolation fits a polynomial curve through multiple data points. Spline interpolation uses piecewise polynomial functions to smoothly interpolate between data points, often yielding more accurate results than simpler methods. Bicubic interpolation is specifically used in image processing to resize images while preserving smoothness and minimizing distortion. Interpolation finds applications in diverse fields, including equation 4.2 tells about the equation of interpolation

Interpolation Formula


$$y = \frac{(y_2 - y_1)}{(x_2 - x_1)} \times (x - x_1) + y_1$$


Interpolation Equation 4.3.2

4.3.3 LINEAR INTERPOLATION(OR) NEAREST PIXEL

Linear interpolation is a straightforward method used to estimate values between two known data points by assuming a linear relationship between them. This technique is widely employed in various fields such as mathematics, engineering, finance, and computer graphics. In linear interpolation, the value at an intermediate point between two known data points is determined by connecting the points with a straight line and finding the value along that line corresponding to the desired position. Linear interpolation is computationally simple and easy to implement, making it a popular choice for basic interpolation tasks. However, it may not always provide the most accurate results, especially when the relationship between data points is non-linear. Additionally, linear interpolation may introduce artifacts or inaccuracies in certain applications, such as image processing, where smoother interpolation methods like spline or bicubic interpolation are preferred. Nonetheless, linear interpolation remains a useful tool for approximating values between known data points in a wide range of applications. When resizing images, linear interpolation is frequently used to scale them up or down. New pixels are added between existing ones when an image is scaled up, and their values are calculated by averaging the values of the pixels that surround them. On the other hand, when an image is scaled down, the pixels that already exist are averaged or blended to create a smaller final image. Even while linear interpolation is easy to use and computationally efficient, when images are scaled greatly, it can cause artifacts like stair-stepping or blurring. This is due to the fact that the approach relies on the assumption that surrounding pixels change at a consistent pace, which may not accurately reflect the underlying image data, especially in areas with complex textures or high contrast. Equation 4.3.3 tells about the linear interpolation

Linear Interpolation

$$y = y_1 + (x - x_1) \frac{(y_2 - y_1)}{(x_2 - x_1)}$$

©easycalculation.com

Linear interpolation Equation 4.3.3

The two known data points in this formula are (x 1, y 1) and (x 2, y 2), whereas the data point that has to be estimated is represented by the symbol ("x," "y").

4.3.4 BILINEAR INTERPOLATION

Bilinear interpolation is a method used to estimate values between four known data points in a two-dimensional grid or image. It assumes a linear relationship not only between points along each axis but also across both axes simultaneously. This technique is widely used in image processing, computer graphics, and geographical information systems (GIS) for tasks such as image resizing and geometric transformations. In bilinear interpolation, the value at an intermediate position within a grid is calculated by interpolating along each axis separately and then combining the results. Bilinear interpolation produces smoother results compared to linear interpolation but may still introduce some artifacts, particularly in areas with large gradients or sharp transitions. Despite its limitations, bilinear interpolation remains a widely used and computationally efficient technique for interpolating values within grids or images. The intensity values of the four closest pixels surrounding the target point are used in bilinear interpolation. Each nearby pixel's weighted contribution to the interpolated value is determined by how far away it is from the target place. The final interpolated value is then calculated by combining these weighted intensities. Scaling and rotating images are two popular activities that make use of bilinear interpolation. Bilinear interpolation produces

fewer distortion and aliasing problems when scaling an image since it maintains the smoothness of edges and details better than linear interpolation. Similar to this, bilinear interpolation minimizes jagged edges and retains image sharpness during image rotation, hence assisting in the preservation of image quality. While bilinear interpolation yields better results than linear interpolation, it is not without its limitations, particularly when large image scaling is involved. It could cause loss of fine details or blurring, especially in areas with high frequency content. Bilinear interpolation is still a commonly used technique because of its versatility and efficiency in image processing applications, even with these disadvantages. The below equation tells about bilinear interpolation

$$P = \frac{(x_2 - x)(y_2 - y)}{(x_2 - x_1)(y_2 - y_1)}Q_{11} + \frac{(x - x_1)(y_2 - y)}{(x_2 - x_1)(y_2 - y_1)}Q_{21} \\ + \frac{(x_2 - x)(y - y_1)}{(x_2 - x_1)(y_2 - y_1)}Q_{12} + \frac{(x - x_1)(y - y_1)}{(x_2 - x_1)(y_2 - y_1)}Q_{22}$$

Bilinear interpolation Equation 4.3.4

This formula can be used to simulate a variety of interactions in which there is a bilinear interaction between the variables x and y.

The bilinear component is represented by the phrase cxy, and the linear component is denoted by the term ax+by. The dependent variable, or output, is represented by the constant term d, which is f(x,y).

The independent variables are x and y. Coefficients or constants are a, b, c, and d.

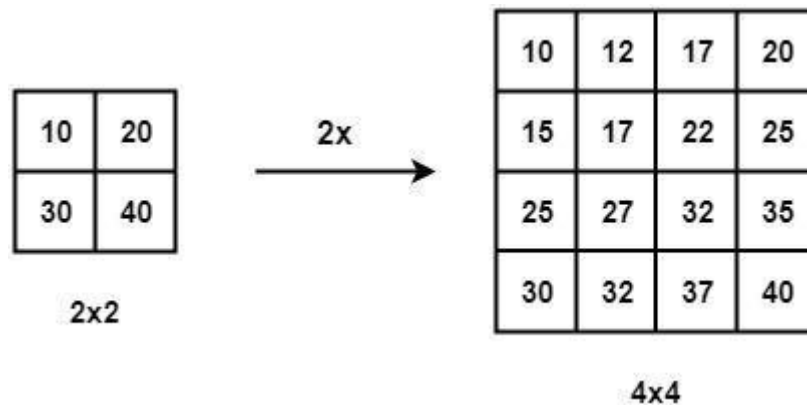


Fig 4.2.4(b) extended matrix

4.3.5 BI-CUBIC INTERPOLATION

Bicubic interpolation is an advanced method used to estimate values between grid points in two-dimensional data, particularly in image processing. It aims to produce smoother and more visually appealing results compared to simpler interpolation techniques like bilinear or linear interpolation. In bicubic interpolation, the interpolated value at a given point is calculated by fitting a cubic polynomial to a 4x4 grid of neighbouring data points. This polynomial is determined using the values of the nearby grid points and their derivatives. The interpolated value is then obtained by evaluating the polynomial at the desired position within the grid. Bicubic interpolation is widely used in tasks such as image resizing, geometric transformations, and image enhancement. It helps to preserve fine details and reduce artifacts such as aliasing or jagged edges that may occur with simpler interpolation methods. While bicubic interpolation is computationally more intensive than linear or bilinear interpolation, its ability to produce high-quality results makes it a popular choice for applications where visual fidelity is important. Unlike bilinear interpolation, which only takes into account the four closest nearby pixels, bicubic interpolation uses a 4x4 neighbourhood of pixels. The intensity value at the appropriate non-grid point is then estimated by fitting a bicubic function to these 16 pixels. Usually derived from a cubic polynomial equation, this function yields an estimate that is smoother and more accurate than bilinear interpolation. For a continuous transition between adjacent pixels, the bicubic function is usually defined in a form that makes it smooth and differentiable up to the second order. Especially when images are greatly scaled, this smoothness helps retain image information and minimizes artifacts like aliasing and jagged edges. Compared to

less complex interpolation techniques, cubic interpolation has a number of advantages. First off, the results are of superior quality, with finer details and sharper edges retained. The below figure 4.3.5 shows how bicubic interpolation works. The below equation 4.3.5 tells about how to calculate bicubic algorithm.

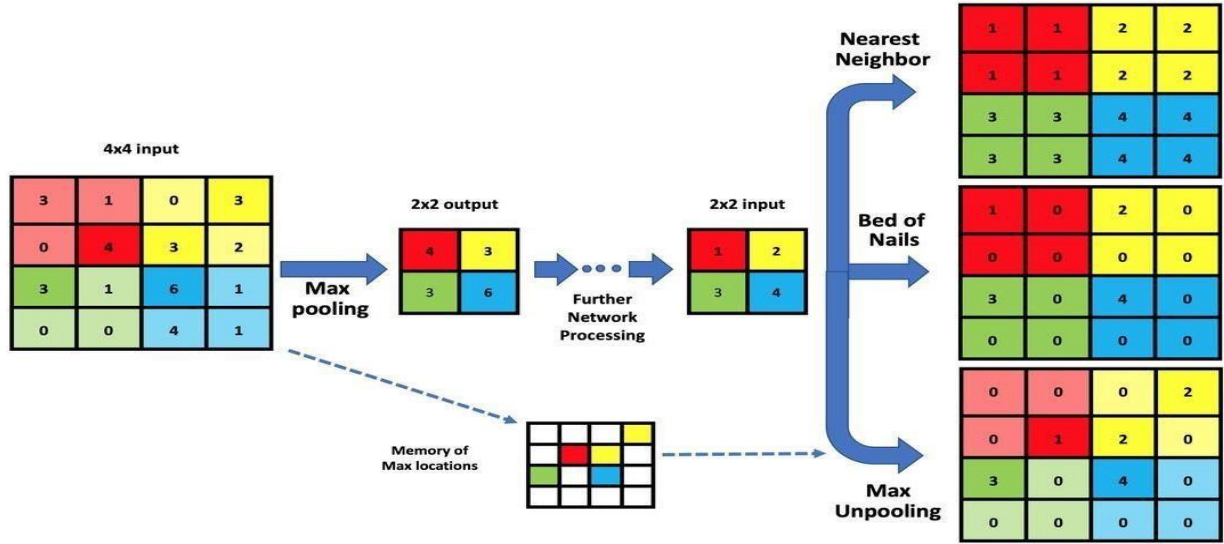


Fig 4.3.5(a) bicubic interpolation matrix

$$v(x, y) = \sum_{i=0}^3 \sum_{j=0}^3 a_{ij} x^i y^j$$

Bicubic interpolation Equation 4.3.5(b)

Where a_{ij} 's are coefficients obtained through interpolation techniques. These coefficients are determined by the values of the function and its derivatives at the grid points surrounding the desired point. Second, it produces aesthetically acceptable resampled images by reducing the introduction of aberrations like blurring and stair-stepping. Last but not least, bicubic interpolation can handle a variety of image processing tasks since it is more resilient to noise and inconsistencies in the input image. Bicubic interpolation is not without its limitations, though. Compared to bilinear interpolation, it requires more computing power and memory due to its higher computational intensity. Furthermore, in areas with high contrast edges, bicubic interpolation may cause overshoot or ringing.

artifacts; nevertheless, these distortions are usually less noticeable when compared to less complex interpolation techniques.

4.3.6 GREY SCALE

A grayscale image, also known as a black and white or monochrome image, is a digital image in which each pixel is represented by a single intensity value, typically ranging from 0 (black) to 255 (white) in an 8-bit grayscale image. Unlike colour images, which have three colour channels (red, green, and blue), grayscale images contain only one channel that represents the intensity or brightness of a pixel. Grayscale images are commonly used in a variety of applications such as photography, medical imaging, and word processing. It is often preferred for tasks that do not require color information or where the emphasis is on contrast and detail rather than colour reproduction. Grayscale images are also more memory efficient and easier to process than colour images, making them suitable for use in resource-constrained environments and applications where computational efficiency is important. In a grayscale image, grey tones represent different levels of brightness, with darker tones corresponding to lower intensity values and lighter tones corresponding to higher intensity values. This allows grayscale images to effectively convey visual information without the use of colour. The below figure 4.3.6 explains about grayscale image

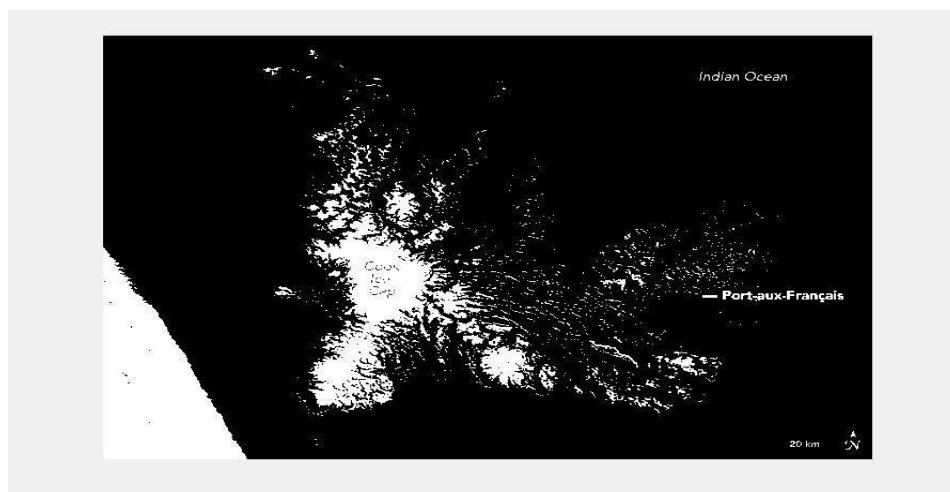


Fig 4.3.6(a) Greyscale image

When an image is represented using shades of grey in image processing, each pixel's intensity and brightness level are matched. This technique is known as grayscale. By lowering computational complexity and maintaining crucial image information, this

single channel representation streamlines image processing processes. In many different fields, including computer vision, medical imaging, and image analysis, grayscale images are widely employed. As a result, there is a reduction in both memory needs and computing complexity for image processing activities. In applications like medical imaging, document analysis, and edge detection methods, where colour information is not crucial, grayscale images are frequently utilized. In order to convert colour photographs to grayscale, the RGB channels are combined with various weights to provide a brightness value for each pixel. This method produces a grayscale image that is more computationally efficient to process while retaining all of the important visual information.

4.4 BINARY DATA

A binary file is a computer file that contains data in binary format, consisting of a series of bits (0's and 1's). Unlike text files, which use characters to store data in a human readable format, binary files store data in a format that can be directly interpreted by a computer. Binary files are used to store various types of data, such as images, audio, video, and executable files.

Binary file structure:

Binary files can be in structured or unstructured format, depending on the type of data they contain. Structured binary files typically have a well-defined format with headers, data fields, and other metadata structures that organize and describe the data stored in the file. Examples of structured binary file formats include image file formats such as JPEG, PNG, and GIF. These file formats include headers that specify image dimensions, color depth, and compression parameters.

On the other hand, unstructured binary files do not adhere to any particular format or structure and may contain raw binary data without any metadata or headers. Examples of unstructured binary files include executable files (such as EXE files) and binary data files that applications use to store custom data formats.

4.5 Application of binary files:

Binaries are used in different applications in different domains such as:

Multimedia: Binary files are widely used to store multimedia data such as images, audio, and video files in formats such as JPEG, MP3, MP4, etc.

Executable programs: Binary files are used to store executable programs and libraries in formats such as .exe (Windows Executable), (Windows Dynamic Link Library), and ELF (Executable and Linkable Format) . Database: Binary files are used to store database files in formats such as SQLite. SQLite stores data in binary format for efficient storage and retrieval.

Network communication: Binary files are widely used in network communication protocols such as TCP/IP and HTTP, where data is sent in binary format to improve efficiency and reduce bandwidth usage.

Binary files have several advantages over text files:

Efficiency: Binary files are usually more efficient than text files in terms of storage space because they do not require encoding characters into readable text. This makes binary files ideal for storing large amounts of data, such as multimedia files or databases. Performance: Binary file I/O operations are generally faster than text file operations because they directly read and write raw binary data without requiring encoding or decoding.

Flexibility: Binary files can store a wide range of data types, including numeric data, strings, structures, and user-defined data formats, making them extremely versatile for storing different types of data.

Table 4.5 : The binary file extracted from greyscale image

Security: Binary files allow you to store sensitive data and information in a format that cannot be easily read or edited by humans, providing a degree of security against unauthorized access or tampering.

4.6 FPGA(FIELD PROGRAMMABLE GATE ARRAY)

Field programmable gate arrays (FPGAs) are integrated circuits that can be programmed and reconfigured after manufacturing, allowing for greater flexibility and customization than traditional application-specific integrated circuits (ASICs). An FPGA consists of a set of configurable logic blocks (CLBs) interconnected by programmable routing resources. This architecture enables the implementation of a variety of digital circuits, ranging from simple logic functions to complex system-on chip (SoC) designs. FPGAs are gaining popularity in various industries due to their versatility, high performance, and time-to-market advantages. One of the main advantages of FPGAs is that they allow engineers to rapidly prototype and iterate designs without spending the time and cost of manufacturing ASICs, thus accelerating development cycles. This agility is especially valuable in industries such as communications, aerospace, automotive, and consumer electronics. Programming an FPGA typically involves a hardware description language (HDL) such as Verilog or VHDL. This allows designers to specify the desired functionality at the hardware level. Alternatively, high-level synthesis tools simplify the design process for a software developer by allowing him to convert a software algorithm into a hardware description for his FPGA implementation. FPGAs are widely used in applications that require highspeed processing, low-latency communications, and real-time signal processing. In telecommunications, FPGAs are used in base stations, routers, and switches to handle data packet routing, protocol processing, and encryption. It is also used in radar systems, medical imaging devices, and scientific instruments for rapid data collection and processing. Another advantage of FPGAs is the ability to integrate multiple functions on a single chip, reducing system complexity and power consumption. This integration facilitates the development of highly integrated solutions for tasks such as motor control, image processing, and sensor interfaces in industrial automation and Internet of Things (IoT) applications. Additionally, FPGAs offer inherent parallelism, allowing multiple tasks to run simultaneously, improving overall system performance. This parallel processing is especially beneficial for compute-intensive workloads such as artificial intelligence (AI), machine learning (ML), and digital signal processing (DSP). FPGAs can be used to accelerate neural network inference, data encryption, and multimedia processing in edge computing devices, cloud servers, and embedded systems. Fig 4.6(a) shows the spartan fpga

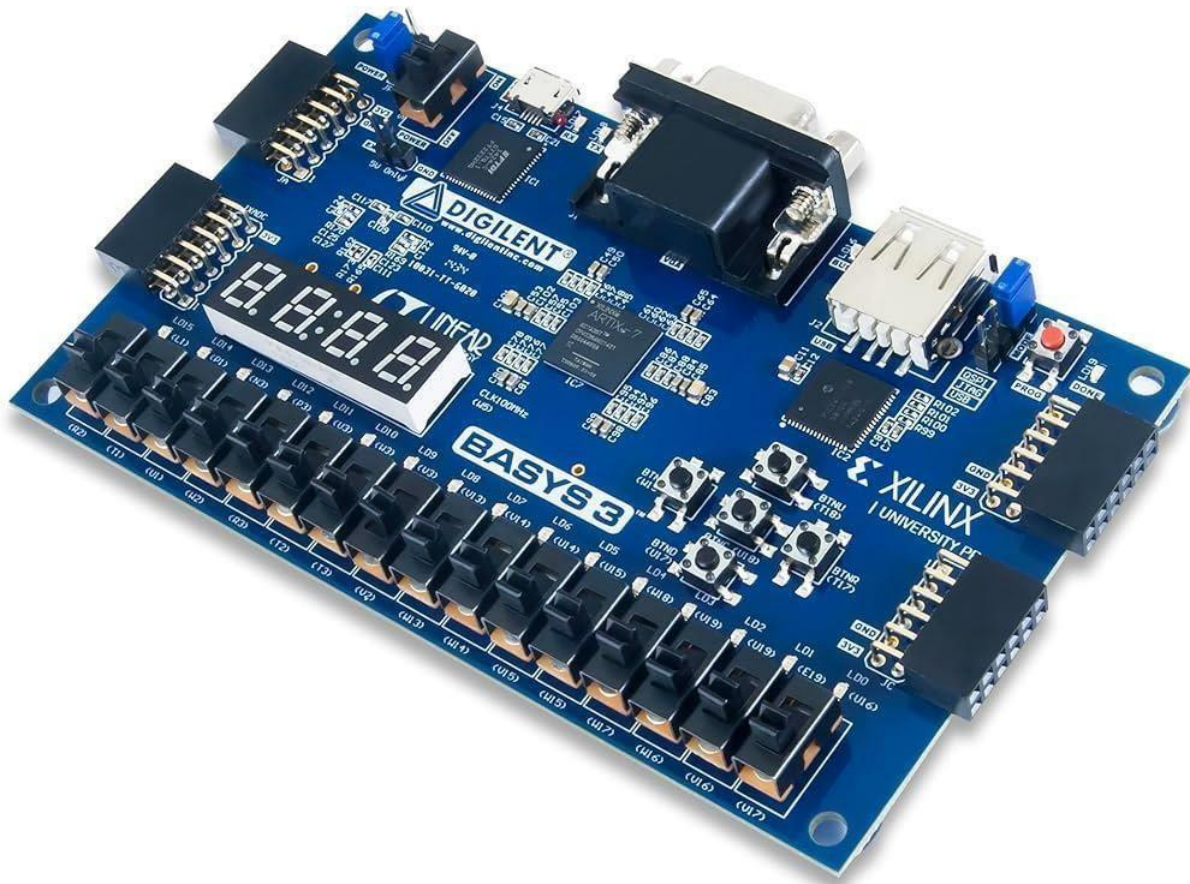


Fig 4.6(a) FPGA

To meet the different needs of different applications, FPGA vendors offer a wide range of products with different performance levels, power consumption, and I/O capabilities. Advanced features such as high-speed transceivers, embedded processors, and dedicated DSP blocks further enhance the capabilities of his latest FPGAs, allowing complex systems to be implemented on a single chip.

Despite the many benefits of FPGAs, they also have challenges, including: B. Design complexity, resource utilization optimization, and verification/validation. To achieve optimal performance and cost efficiency, developers must carefully balance factors such as logic utilization, timing constraints, and power consumption. In summary, FPGAs are powerful tools for implementing custom digital circuits and accelerating development cycles in a variety of applications. Its flexibility, performance, and integration capabilities make it an essential component of modern electronic systems, driving innovation across industries and enabling cost-effective implementation of complex functionality.

Implementing bicubic interpolation for satellite images involves adhering to certain standards, navigating constraints, and making trade-offs to ensure optimal performance and reliability:

Standards:

Accuracy Standards: Ensuring that the interpolated images maintain a high level of accuracy compared to the original data is crucial. This may involve adhering to specific accuracy metrics defined by relevant authorities or research communities.

Computational Standards: Adhering to computational standards to ensure efficient processing of large-scale satellite imagery datasets. This includes optimizing algorithms and implementations to meet performance requirements.

Constraints:

Computational Resources: Limited computational resources, such as processing power and memory, can constrain the implementation of bicubic interpolation. Balancing the computational complexity of the interpolation algorithm with available resources is essential. **Data Quality:** The quality of satellite imagery can vary due to factors like sensor characteristics, atmospheric conditions, and geometric distortions. Implementations must account for these variations and adjust interpolation parameters accordingly. **Time Constraints:** Depending on the application, there may be time constraints for processing satellite imagery. Balancing the need for accuracy with processing speed is essential, especially in time-sensitive scenarios like disaster response. **Bandwidth Constraints:** Transmitting large satellite image files can be bandwidth-intensive, particularly in remote areas or regions with limited connectivity. Implementations must consider bandwidth constraints when designing interpolation algorithms or transmitting interpolated images.

Trade-offs:

Accuracy vs. Computational Complexity: Increasing the accuracy of bicubic interpolation may require higher computational complexity. Implementations must strike a balance between accuracy and computational efficiency, depending on the specific requirements of the application. **Speed vs. Quality:** Faster interpolation methods may sacrifice some image quality compared to more computationally intensive approaches. Trade-offs between speed and quality must be carefully considered based on the

application's needs. Resource Utilization vs. Scalability: Optimizing resource utilization for current hardware configurations while ensuring scalability for future hardware advancements is a trade-off that implementations must navigate. In summary, implementing bicubic interpolation for satellite images involves navigating standards, constraints, and trade-offs to achieve the desired balance between accuracy, efficiency, and resource utilization. Effective implementations carefully consider these factors to deliver reliable and high-quality interpolation results for various satellite imagery applications.

CHAPTER 5

RESULT AND DISCUSSION

The effectiveness of bicubic interpolation for satellite images can be attributed to its ability to capture fine detail and achieve smooth transitions between pixels, resulting in visually appealing and informative images. Applications that require high quality image processing because by considering a larger neighbourhood of pixels and fitting a cubic polynomial to approximate intensity values, bicubic interpolation minimizes distortion and maintains image sharpness. Fig 5 a,b shows the result before interpolation and after interpolation

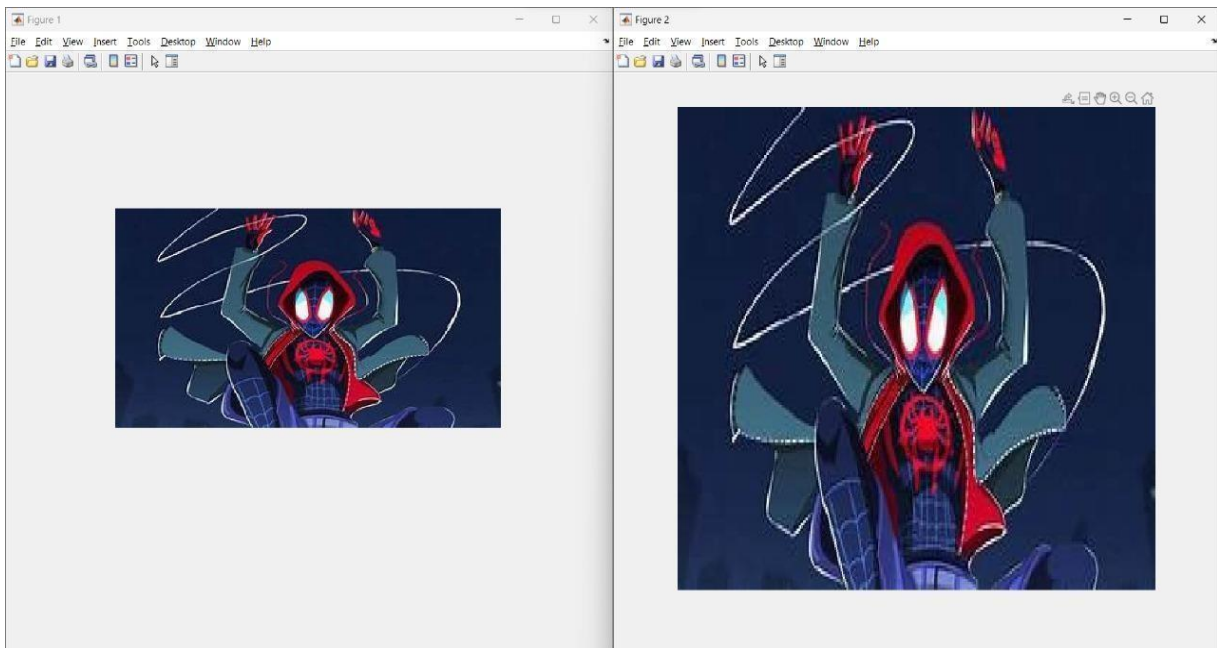


Fig 5(a) Before interpolation

Fig 5(b) After interpolation

However, it is important to consider some limitations and challenges associated with bicubic interpolation of satellite images. One potential issue is the computational complexity of the interpolation process, which can be a challenge in real-time processing and resource-constrained environments. Additionally, the performance of bicubic interpolation may vary depending on factors such as image resolution, noise level, and the presence of artifacts. Future research directions may focus on addressing these challenges and optimizing bicubic interpolation algorithms for satellite image

processing. This may include developing efficient parallel processing techniques, optimizing interpolation parameters, and considering adaptive interpolation strategies for different types of satellite imagery. Overall, the results and discussion highlight the effectiveness of bicubic interpolation in improving the visual quality of satellite images and preserving important features. By using advanced interpolation techniques, researchers and practitioners can improve the interpretability and usefulness of satellite imagery for a variety of applications such as environmental monitoring, urban planning, and disaster management.

Quantitative metrics such as maximum signal-to-noise ratio (PSNR), structural similarity index (SSI), and mean squared error (MSE) were calculated to evaluate the quality of the interpolated images. The results show a significant increase in PSNR values, indicating high image fidelity and high similarity to the original image.

Table 1: Performance comparison study

S.NO	DATA	PSNR	SSIM	EPSNR
1	LAKE	21.68	0.80	21.66
2	OCEAN	20.40	0.58	20.32
3	DESERT	22.46	0.87	22.31
4	FALLS	22.60	0.88	22.14
5	HILLS	24.42	0.89	24.22
6	MOUNTAIN	20.40	0.58	20.32
7	VOLCANO	23.77	0.82	23.74
8	GLACIER	30.20	0.90	30.11
9	PLANE	21.05	0.69	21.04

These are some results of (PSNR) Values Signal to noise ratio and SSIM, this table has previous values of the image and after values of the image. Below fig 5(c), 5(d), 5(e) is the MATLAB interpolated picture

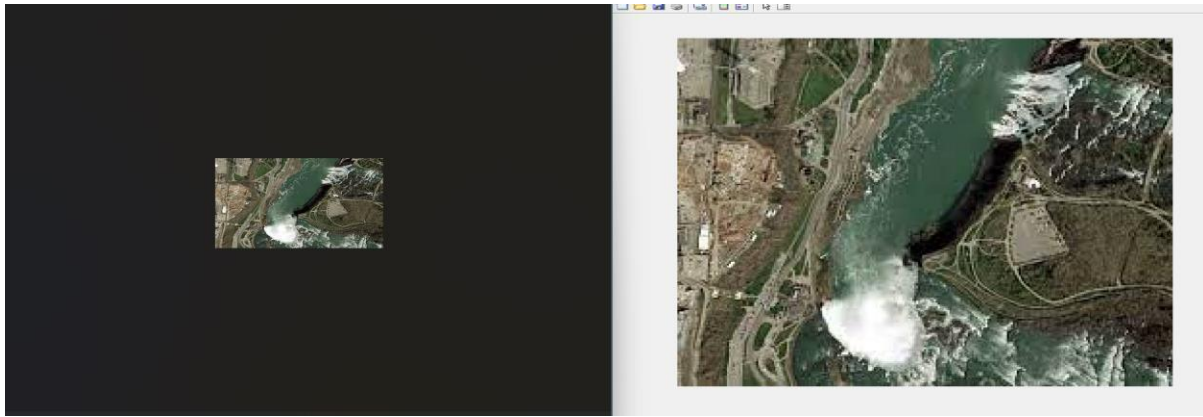


Fig 5(c)(1) Before interpolation lake

Fig 5(c)(2) After interpolation lake

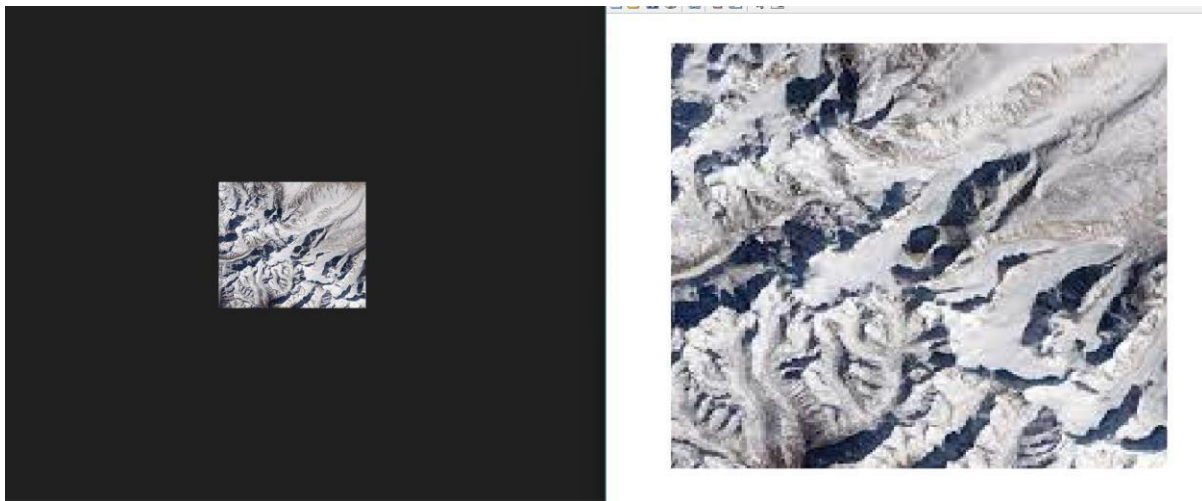


Fig 5(d)(1) Before interpolation glacier

Fig 5(d)(2) After interpolation glacier

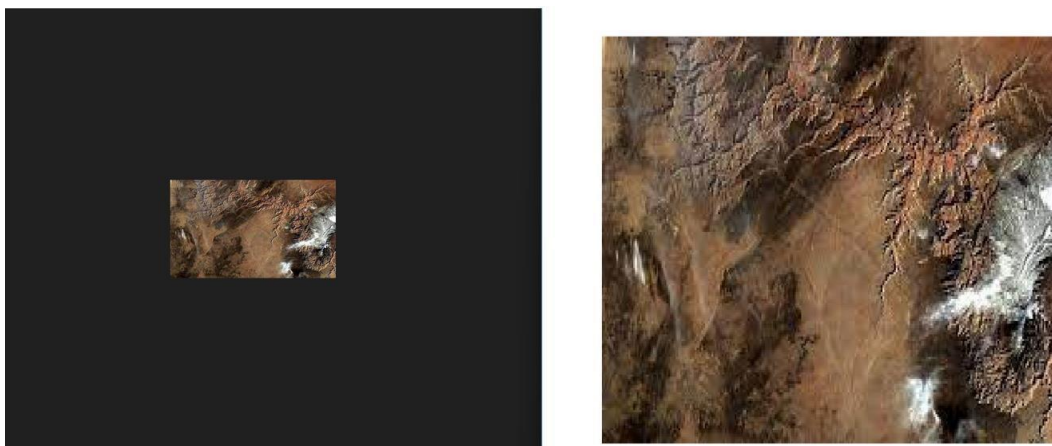


Fig 5(e)(1) Before interpolation desert

Fig 5(e)(2) After interpolation desert

CONVERTING INTO GRAY SCALE

When a coloured image is converted to a single-channel image that represents intensity values, usually ranging from black to white, the process is known as grayscale image processing. By doing this simplification, computational complexity is decreased without sacrificing image structure. Grayscale photographs offer useful information in a more straightforward format, which makes them popular in a variety of applications such as compression, feature extraction, and image improvement. Some results of grayscale. Fig 5(f) is greyscale image of lake and 5(g) shows greyscale of desert

CONVERTING INTO BINARY

When describing images in image processing, binary data refers to employing just two intensity values, usually 0 and 1. Pixels in this format are either white (expressed by 1) or black (represented by 0). Tasks like segmentation, pattern recognition, and object detection heavily rely on binary images. Because they only consider the existence or absence of features, they simplify image analysis and are especially helpful for situations where just structure or shape are important. In addition to being more memory and computationally efficient than grayscale or colour images, binary images are also efficient for processing and storing.

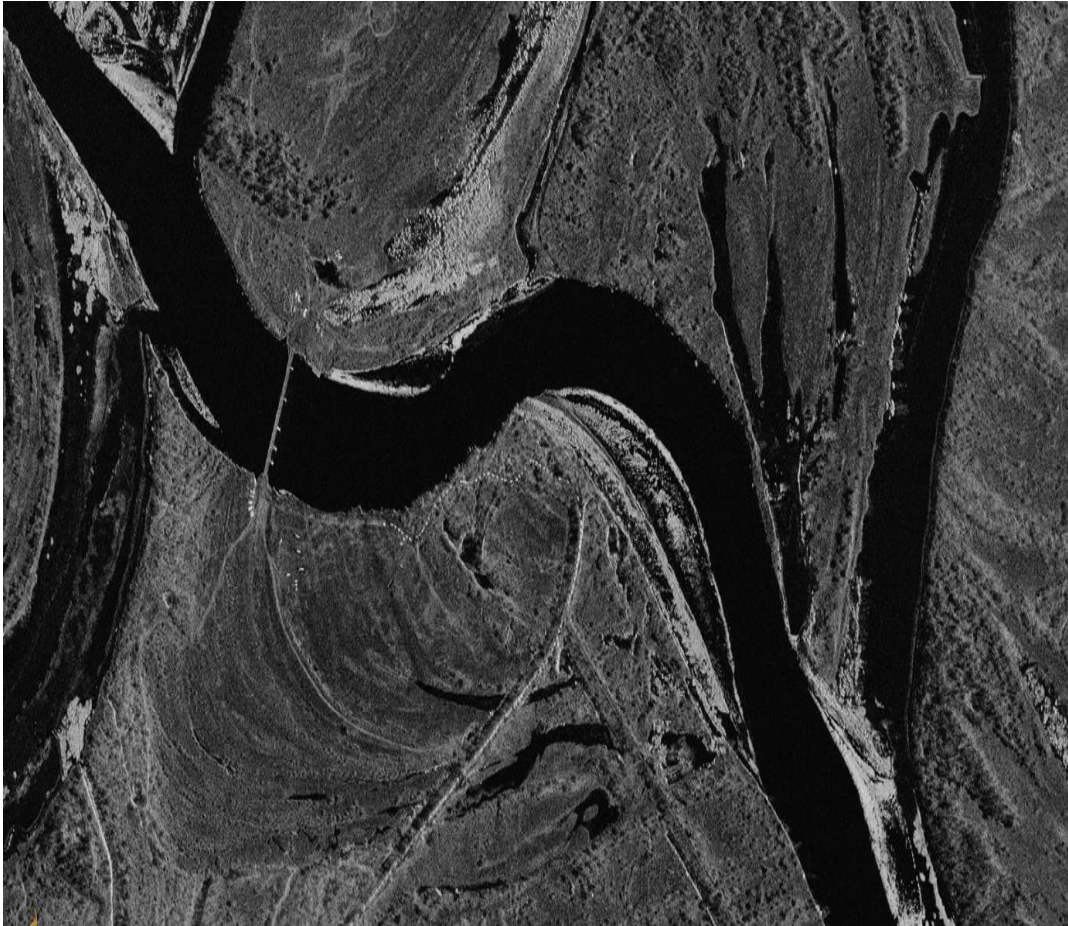


Fig 5(f) Greyscale of Lake

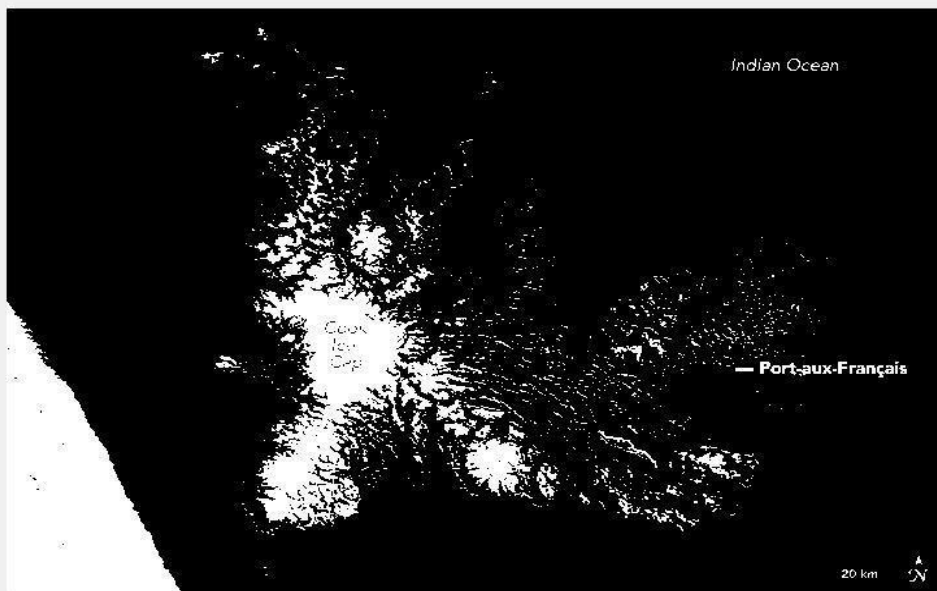


Fig 5(g) Greyscale image of Desert

CHAPTER 6

CONCLUSION

In image quality and accuracy, vital for various applications such as remote sensing, geographical mapping, and environmental monitoring. Through this interpolation technique, finer details and smoother transitions are achieved, resulting in clearer and more visually appealing images. By leveraging bicubic interpolation, the distortion effects inherent in satellite imagery due to factors like sensor limitations and atmospheric conditions can be mitigated, leading to more precise analysis and interpretation of data. Moreover, the ability to interpolate missing or corrupted pixels enables the reconstruction of damaged images, enhancing the reliability and completeness of the satellite image datasets. Furthermore, the computational efficiency of bicubic interpolation allows for real-time or near-real-time processing of large volumes of satellite imagery, facilitating rapid decision-making in time-sensitive scenarios such as disaster response and emergency management. In conclusion, the implementation of bicubic interpolation represents a valuable advancement in satellite image processing, offering improved image quality, enhanced data integrity, and increased efficiency in various applications crucial for scientific research, resource management, and societal welfare.

REFERENCES

- [1] F. Yeganli, M. Nazzal, H. Ozkaramanli, "Super-resolution using multiple structured dictionaries based on the gradient operator and bicubic interpolation," 24th Signal Processing and Communication Application Conference (SIU), Zonguldak, pp. 941-944, 2016.
- [2] F. Yeganli, M. Nazzal, H. Ozkaramanli, "Selective superr using multiple dictionaries and bicubic interpolation," 23rd Signal Processing and Communications Applications Conference (SIU), Malatya, pp. 1957-1960, 2015.
- [3] T. Sharmila, L. M. Leo, "Image upscaling based convolutional neural network for better reconstruction quality," 2016 International Conference on Communication and Signal Processing (ICCSP), Melmaruvathur, pp. 0710-0714, 2016.
- [4] M. N. Naik, P. R. Kumar, "Fast coding with edge preservation for video/image interpolation," 2016 International Conference on Communication and Signal Processing (ICCSP), Melmaruvathur, pp. 1270-1276, 2016.
- [5] B. Dhivya, M. Sundaresan, "Performance analysis of interpolation methods for improving sub-image content-based retrieval," 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, pp. 3909-3912, 2016.
- [6] S. Sonnad, "A survey on fusion of multispectral and panchromatic images for high spatial and spectral information," 2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), Chennai, pp. 177-180, 2016.
- [7] F. Alonso-Fernandez, R. A. Farrugia, J. Bigun, "Very low-resolution iris recognition via Eigen-patch super-resolution and matcher fusion," IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS), Niagara Falls, NY, USA, pp. 1-8, 2016.
- [8] B. M. Ngocho, E. Mwangi, "Single image super-resolution with guided back-projection and LoG sharpening," 18th Mediterranean Electrotechnical Conference (MELECON), Lemesos, pp. 1-6, 2016.
- [9] K. Sekar, V. Duraisamy, A. M. Remimol, "An approach of image scaling using DWT and bicubic interpolation," International Conference on Green Computing Communication and Electrical Engineering (ICGCCEE), Coimbatore, pp. 1-5, 2014.

- [10] X. Kang, S. Li, J. Hu, "Fusing soft-decision-adaptive and bicubic methods for image interpolation," Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), Tsukuba, pp. 1043-1046, 2012.
- [11] K. Guo, X. Yang, H. Zha, W. Lin, S. Yu, "Multiscale Semilocal Interpolation With Antialiasing," In IEEE Transactions on Image Processing, Vol. 21, No. 2, pp. 615-625, 2012.
- [12] J. Angulo, R. Verdú-Monedero, J. Morales-Sánchez, "Multiscale local multiple orientation estimation using mathematical morphology and B-spline interpolation," 7th International Symposium on Image and Signal Processing and Analysis (ISPA), Dubrovnik, pp. 575-578, 2011.
- [13] Z. Dengwen, "An edge-directed bicubic interpolation algorithm," 3rd International Congress on Image and Signal Processing, Yantai, pp. 1186-1189, 2010.
- [14] X. g. Zhang, "A New Kind of Super-Resolution Reconstruction Algorithm Based on the ICM and the Bicubic Interpolation," International Symposium on Intelligent Information Technology Application Workshops, Shanghai, pp. 817-820,.
- [15] A. Amirany and R. Rajaei, "Nonvolatile, Spin-Based, and Low-Power Inexact Full Adder Circuits for Computing-in-Memory Image Processing," Spin, vol. 9, no. 3, p. 1950013, 2019.
- [16] R. Rajaei and A. Amirany, "Nonvolatile Low-Cost Approximate Spintronic Full Adders for Computing in Memory Architectures," IEEE Transactions on Magnetics, vol. 56, no. 4, pp. 1-8, 2020.
- [17] F. Sabetzadeh, M. H. Moaiyeri, and M. Ahmadinejad, "A Majority-Based Imprecise Multiplier for Ultra-Efficient Approximate Image Multiplication," IEEE Transactions on Circuits and Systems I: Regular Papers, pp. 1-9, 2019.
- [18] K. Gribbon, C. Johnston, and D. G. Bailey, "A real-time FPGA implementation of a barrel distortion correction algorithm with bilinear interpolation," in Image and Vision Computing New Zealand, 2003, pp. 408-413.
- [19] M. A. Nuno-Maganda and M. O. Arias-Estrada, "Real-time FPGA-based architecture for bicubic interpolation: an application for digital image scaling," presented at the 2005 International Conference on Reconfigurable Computing and FPGAs (ReConFig'05), 2005.

- [20] E. Meijering and M. Unser, "A note on cubic convolution interpolation," *IEEE Trans Image Process*, vol. 12, no. 4, pp. 477-9, 2003.