About The Book

The processing of digital images by means of an algorithm on a digital computer is the field of digital image processing. Digital image processing, which is a subclass of digital signal processing and a discipline in its own right, provides numerous benefits over analogue image processing. It makes it possible to apply a much larger variety of algorithms to the data that is being entered and may help solve issues like the accumulation of noise and distortion as the data is being processed. The processing of digital images may be described in the form of multidimensional systems if it is taken into consideration that images are defined across more than two dimensions.

Analog and digital image processing are the two primary kinds of approaches that are used in the field of image processing. For tangible copies, such as prints and pictures, the analogue image processing method may be used. While doing work with these visual approaches, image analysts use a variety of interpretation principles from their toolkits. The digital image processing methods allow for digital images to be manipulated via the use of personal computers. When employing digital techniques, there are three main steps that all different kinds of data have to go through. These steps are known as pre-processing, augmentation, presentation, and information extraction.

In order to learn about the steps involved and the many components of Digital Image Processing, "Digital Image Processing" is a useful guide. People who read this book will have access to a wealth of helpful knowledge. Everything in this chapter is important, and the book does a great job of explaining all the fundamental ideas you'll need to know. Readers may learn a lot about computers and other digital devices while exploring the fascinating field of image processing. This book contains a wealth of information on the subject, covering a wide range of issues and providing clear explanations of each. The concepts presented in this book are presented effectively, and the writers have made the text simple to read. By reading this book from cover to cover, you will get insight into many different aspects of digital image processing. Students may prepare for their exams, write notes, and study using this book all in one convenient resource.

Price: 585 INR

9 789395 468312

AGPH BOOKS
ACADEMIC GURU PUBLISHING HOUSE

AGPH Books

FIRST EDITION

DIGITAL IMAGE PROCESSING

Ashima Kalra
Dr. Aiyah S. Noori
Mrs. Veenu
Mr. Jonnadula Narasimharao

AGPH BOOKS
ACADEMIC GURU PUBLISHING HOUSE

Digital Image Processing

by

Ashima Kalra

Dr. Aiyah S. Noori

Mrs. Veenu

&

Mr. Jonnadula Narasimharao

AG PH Books

2022

Digital Image Processing

Ashima Kalra, Dr. Aiyah S. Noori, Mrs. Veenu and Mr. Jonnadula Narasimharao

ISBN – 978-93-95468-31-2

Published by:

AGPH Books (Academic Guru Publishing House)

Bhopal, M.P. India

Contact: +91-7089366889

About Authors

Ashima Kalra Currently working as an Assistant professor in Electronics and Communication Department in Chandigarh group of colleges, Landran, (Mohali) Punjab, India. She is a Gold Medalist in B Tech in Electronics from Kurukshetra University, Kurukshetra in 2003 .Received M tech degree from Punjab Technical University, Jalandhar in 2008 and pursuing Ph.D in the field of machine learning. She has teaching experience of 19 years at Post Graduate and Under Graduate Level. Her research activities include designing model identification using neural networks, fuzzy systems, supervised learning .machine learning.She has published more than 40 papers in reputed journals and 3 book chapters in springer series. Published 6 patents and 4 text books.

Dr. Aiyah S. Noori is Lecturer at Al-Mustaqbal University College in Iraq, Department of Applied Medical Physics, She holds a Ph.D. in Physicl Science from Baghdad University, College of Science for Women with distinction. She has worked on publishing research in the field of cold plasma and its medical effects on DNA by using digital image processing to determine the rates of damage in the DNA by Commet Assay technique. It is also concerned with the practical aspect of preparing nanomaterials by extracting herbs using plasma and laser techniques, and studying the effect of these materials on the vital functions of the body using digital image processing that shows the basic algorithms of the prepared nanomaterials. She supervises many undergraduate students and Determine their research way, as well as She conducts many awareness campaigns that support sustainable development. It is mainly concerned with publishing research that links between cold plasma and texture analysis of digital images processing , as well as preparing nanomaterials by green methods.

Mrs. Veenu Bhatia is Assistant Professor in Computer Scienec Department currently working in Arya P.G. College, Panipat.

She had done MCA, M.Phil in Computer Scienec and UGC Net in Same subject. She is also pursing Ph.D. In Computer Science and Applications. She has total experience in teaching of 22 years in her subject. She had presented and published many papers. She had awarded by different awards as Award Memento on Teachers Day by Lions Club Panipat

- ❖ International Teacher Award by International Institute of Organized Research (I2OR)
- ❖ National Adroit Educator Award 2021 by Green ThinkerZ Society
- ❖ Honour as Judge in exhibitions and presentations.

- ❖ Global Innovative Educator Award 2021 by International Institute of Organized Research (I2OR)
- ❖ National Distinguished Educator Award 2021 by International Institute of Organized Research (I2OR)
- ❖ I2OR International Teaching Excellence Award 2022 by International Institute of Organized Research (I2OR) on the occasion of International Day of Education.

Mr. Jonnadula Narasimharao is currently working as an Assistant Professor in the Department of Computer Science and Engineering, CMR Technical Campus ,Medchal ,Hyderabad. He obtained his Bachelor Degree in B.E – Computer Science and Engineering from DMI College of Engineering, Anna University and Master's Degree in M.Tech – Computer Science and Engineering from TRR Engineering College , JNTU Hyderabad. He is Pursuing his doctorate in the field of Image Processing at Madhav University , Pindwara, Rajastan. His areas of Specialization include Image Processing, Deep Learning, Machine Learning, IOT, Data Mining and Networks. He has more than 15 years of Teaching Experience. He has Published his Research work in reputed international journals with high impact factor in Elsevier, Springer, web of science, Scopus. He has certified in NPTEL, Coursera courses. Added to that He has also presented papers in both international and national conferences. He has an Indian patent in his expertise areas from Computer Science and Engineering field. He is a life member of the professional body - Indian Society for Technical Education (ISTE).

Preface

The processing of digital images using a digital computer is what is meant by the term "digital image processing." In order to acquire an improved image or to extract some relevant information, we may also say that it is the use of computer algorithms.

The book "Digital Image Processing" is a resource that will aid readers in gaining an understanding of the process as well as all of the fundamental components that are included in Digital Image Processing. The readers are going to gain much from the wealth of knowledge included in this book. This book guides the reader through all of the important concepts associated with the topics, and every piece of information that is presented in this book is important. Image processing is a fascinating field of study, and through studying it, readers will also learn a great deal about computers and other digital devices. Numerous issues connected to the subject are addressed in this book, and each topic is well described.

The authors of this book have made it simple to read and simple to comprehend by using uncomplicated language, and the ideas presented in this book are explained in a clear and concise manner. If an individual reads this book all the way through, they will acquire knowledge about a variety of significant aspects that are associated with digital image processing. In addition, students may prepare for their assessments by using this book as a resource

Table of Content

CHAPTER 1 Representation

1.1. What is an image?

A digital picture is a binary representation of visual data, whereas an image is a graphical depiction of the same thing. Photographs, graphics, and even stills from videos all qualify such visuals. In this context, "image" refers to any digitally produced or replicated photograph that has been archived.

In addition to pixel density, one may talk about an image's quality in terms of either vector graphics or raster graphics. Some people use the term "bitmap" to refer to a raster picture. What we call an "image map" is essentially a data file that contains information linking various parts of a given picture to one another through hypertext.

An "image" (from the Latin "imago") is any item, such as photograph or other two-dimensional representation that represents a topic (often a physical thing) by resembling that subject. Signal processing defines a picture as a spatially distributed amplitude of colour. A writing system known as a pictorial script is one that uses pictures, rather than the abstract signs employed by alphabets, to represent different semantic concepts in place of those symbols.

Photographs and digital displays are examples of two-dimensional images, whereas statues and holograms are examples of three-dimensional images. Photos may be taken using any optical equipment, including the human eye and water, and with natural objects and phenomena like mirrors, lenses, telescopes, microscopes, and so on.

The term "image" may also refer to any flat two-dimensional representation, such as a map, graph, pie chart, painting, or banner. In this broader meaning, pictures may be created in a number of different ways, including manually (by drawing, painting, or carving), mechanically (via printing or computer graphics technology), or through a blend of the two (as in a pseudo-photograph).

A fleeting reputation cannot be built upon. This might be the image of an item in a mirror, the picture projected from a camera obscura, or the image on a cathode ray tube. A hard copy, also known as a fixed image, is a picture or other digitally-recorded image that has been permanently imprinted or otherwise affixed to a physical medium such as paper or fabric.

A mental picture is a representation of an object or scene in one's imagination or memory. Images may depict anything, from real-world objects to purely abstract ideas like graphs and functions.

1.1.1. Image layout

The layout of an image is its presentation on the page and its relationships to other components. Photos may be used as page backgrounds, in column layouts with accompanying text, or as stand-alone pictures. Changing an image's orientation and placement in a media file might help you create a more compelling tale about your company. The layout pinpoints exactly where everything that will be in the final picture.

1.1.2. Image colour

Digital pictures storing colour information are called colour images, and they consist of three monochrome bands, each of which stores a distinct hue. Each colour channel of the photos is represented by a range of greys.

In this case, the photos are color-coded in red, green, and blue (RGB images). The 24 bits/pixel used to create each colour picture breaks down to 8 bits for each of the three colour channels (RGB).

A colour image is a photograph that appears in full colour on a computer monitor or other kind of screen. On the other hand, photos that are solely shown in black and white or in grayscale are referred to as black-and-white images and grayscale images respectively. There are many different file formats that may be used to save and display a colour picture. A computerised device has to either have its own display equipment, such as a monitor, that is able to exhibit the necessary colours in order for colour pictures to be presented accurately, or it must be connected to such an apparatus. Alterations both to the picture's file format and to the device that is being used to show the image might result in colours that seem somewhat different from one device to the next.

Each pixel in a colour picture will have its colour recorded in the file that represents the image. One may think of the way in which each pixel's colour data is kept as being analogous to the way in which three- or higher-dimensional coordinates are. For instance, specifying a number for the "intensity" of red, green, and blue in a colour picture is a typical way to indicate a certain colour. Because of the wide range of colours that may be created by combining these three colours, it is typically sufficient to provide only one of the three to indicate the desired colour of a pixel. Hue-saturation-lightness (HSL) is another popular coordinate-based colour scheme, in which variable values for hue, saturation, and lightness are utilised to obtain the required colours.

Size, compression, and other variables may have a significant impact on the quality of various colour picture formats. The disc space required to keep track of every pixel's colour data might be rather large. Since each pixel in a high-quality colour photograph carries a great deal of colour information, the file size of such an image tends to be rather enormous. Subtle, low-quality photos are suitable for most applications, but they may have short inconsistencies and defects that hint at a small file size and restricted image quality. This is because small file sizes and limited image quality are both indicative of limited image quality.

Images with colour are used often by people. Most graphical user interfaces (GUIs) nowadays are shown in colour, necessitating the regular creation of such graphics. It is quite unlikely that a person would navigate the internet without coming across some type of coloured picture, whether it is in the form of an advertising or the actual content of a website. Creating, processing, and studying high-quality colour photographs is an integral part of several careers and fields of study. Due to the fact that even minute variations in the layouts and densities of pixels of various hues may have a significant impact, photographs of this kind often have extremely high file sizes.

1.2. Resolution and quantization

1.2.1. Resolution

Resolution describes how much information is included in a picture. You may use the phrase for both digital and film photography. Having a "higher resolution" suggests that there is more information included in the picture.

There are several ways in which the resolution of an image may be evaluated. Resolution measures how closely lines may be drawn to one another without becoming blurry. Resolution units may be related to physical quantities (such as lines per mm or lines per inch), the total size of an image (lines per picture height, often known as lines, TV lines, or TVL), or angular subtense. Line pairs, consisting of a dark line and an adjacent bright line, are often employed in place of individual lines; for instance, a resolution of 10 lines per millimetre implies 5 dark lines alternating with 5 light lines, or 5 line pairs per millimetre (5 LP/mm). It is common practise to express the resolution of a camera's lens or film in terms of the number of lines that may be resolved per millimetre.

Pixels per inch (PPI) is the standard unit of measurement for describing the resolution of a picture.

A higher resolution means there are more pixels per inch (PPI), which in turn means more information per pixel and a more detailed, high-quality picture.

Low-resolution images feature fewer pixels, which may be easily seen if the picture is enlarged to an extreme size (which sometimes happens when an image is stretched).

By adjusting the image's resolution, you may specify how many pixels should be included inside a square inch of the picture. For illustration's sake, a picture with a resolution of 600 ppi will have 600 pixels packed into each image of an individual. Images with a pixel density of 600 pixels per inch (ppi) will have a high level of clarity and detail. In contrast, a 72ppi picture contains far fewer individual pixels. You're probably already anticipating that it won't seem as crisp as the original 600ppi picture.

One rule of thumb for picture resolution is to capture the image with the highest possible quality setting whether scanning or taking a photograph.

1.2.1.1. Choosing the Correct Resolution for your Image

1. **Printing Resolution**
    - **a.** Professional Publications

Image resolutions of up to 600 pixels per inch (ppi) are recommended for printing on certain professional and high-end printers. Before sending in photographs, always double-check with the printer or publisher to see what kind of quality they want.

- **b.** Non-Professional

Images in a ppi range of at least 200 to 300 and preferably greater will provide the best results when printed on non-professional printers such as inkjet, laser, and other common printers. 200 ppi is sufficient for photos that just need to "look decent." It is suggested that a print resolution of 300 ppi be used for photographs. Depending on the viewing distance, images for big size poster printing might be between 150 and 300 ppi.

2. **Screen Resolution**

Screen pictures are distinct from images intended for printing in that we must consider the pixel dimensions of monitors, TVs, projectors, or display rather than PPI when creating screen images. PPI should be used for printed pictures, but the image's pixel measurements should be used to decide the size of the image and the quality of how it will look on the web or devices.

a. Web

For a long time, the consensus has been that 72 PPI is the ideal resolution for storing photos. It is a frequent fallacy, however, the resolution of an image or its PPI value is the determining element of the picture quality for online photos.

As a result of the fact that each monitor is unique and has its own unique resolution, it might be challenging to build a website that includes graphics that will appear appropriately on all various kinds of displays. With the advancement of technology, screen resolution and refresh rates have both increased. The latest Macbooks, iPhones, and iPads all use Apple's retina screens, which are quickly becoming the industry standard.

b. Projector / Powerpoint

Pictures intended for projectors should have the same pixel dimensions as the projector, much as online images. Projectors, just like computer screens, have their own unique dimensions for displaying content. For instance, the majority of projectors with a 4:3 aspect ratio have a display of 1024 × 768 pixels; hence, an image that is 1024 x768 pixels in size and has a resolution of 72 PPI would be an appropriate picture size to be presented from a projector.

1.2.2. Quantization

The process of transferring input values from an indefinitely long set of continuous values to a smaller set of finite values is what we mean when we talk about quantization. Quantization is a technique for carrying out signal modulation. A given analogue input is converted into digital signals by the process of quantization, which serves as the foundational method for lossy compression algorithms. D/A converter is built upon these algorithmic pillars. Quantizers refer to hardware implementations of the quantization method. These gadgets help in approximating the mistakes in a quantized value, which is an input function.

Image processing involves the use of a technique called quantization, which is a lossy compression method. This method involves compressing a range of values into a single quantum value. The compressibility of a stream improves as the number of discrete symbols decreases. In order to decrease the size of a digital picture file, one strategy is to minimise the amount of colours used to depict the image. Particular uses include the DCT data quantization in JPEG and the DWT data quantization in JPEG 2000.

Color quantization

Color quantization is the process of reducing the number of colours that are utilised in a picture. This is useful for displaying images on devices that only support a limited number of colours as well as for effectively compressing certain types of images. The ability to quantize colours is a standard feature of many image editors and operating systems. The closest colour approach, the median cut strategy, and the octree-based algorithm are all examples of popular current colour quantization algorithms.

It is standard practise to use dithering in conjunction with colour quantization to provide the appearance of a greater number of colours and to remove banding problems.

1.3. Bit-plane slicing

Each pixel in a digital picture has a grayscale value that is represented by one or more bytes in the image's data. An 8-bit image represents a value of 0 as 00000000 and a value of 255 as 11111111. Each byte may represent any value from 0 to 255. Because a change in that bit would dramatically alter the value that is encoded by the byte, it is referred to as the most significant bit (MSB). This bit is located on the extreme left side of the byte. Since a change in this bit does not have a major impact on the encoded grey value, it is referred to as the least significant bit, or LSB. The following equations provide the bit plane representation of an eight-bit digital image:

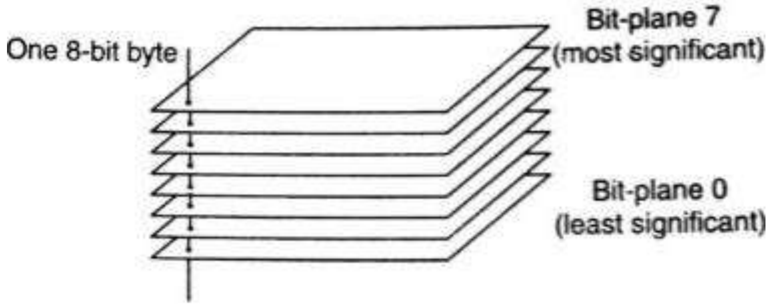


Figure 1.1 Bit plane slicing

The process of encoding an image with one or more bits of the byte being utilised for each pixel might be referred to as bit plane slicing. Only the most significant bit (MSB) of the pixel may be represented, turning the grayscale original into a binary one. Bit-plane slicing may be used for three primary purposes:

- The process of changing a grayscale picture into its binary counterpart.
- The process of representing a picture using fewer bits, which in turn causes the image to take up less space.
- Bringing more clarity to the picture by focusing on it.
- The picture that is being provided is a 3-bit image since the maximum grey level is 7. First, we take the

*https://www.ques10.com/p/5922/short-note-bit-plane-slicing/#:~:text=Bit%20plane%20slicing%20is%20a,image%20to%20a%20binary%20image.

picture and divide it into bit planes by going through a binary conversion.

110	111	110	110	111
000	000	000	001	010
001	001	001	010	011
100	101	101	100	010
110	110	110	111	111

When we split the bit planes apart, we get

1	1	1	1	1
0	0	0	0	0
0	0	0	0	0
1	1	1	1	0
1	1	1	1	1

MSB plane

1	1	1	1	1
0	0	0	0	1
0	0	0	1	1
0	0	0	0	1
1	1	1	1	1

Centre bit plane

0	1	0	0	1
0	0	0	1	0
1	1	1	0	1
0	1	1	0	0
0	0	0	1	1

LSB plane

1.4. Image formats

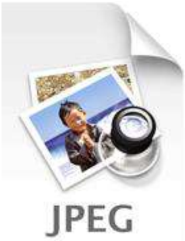
Image Format specifies the encoding scheme to be used for storing image-related information. Compressed data, uncompressed data, and vector data may all be saved. There are benefits and drawbacks of using various picture file formats. Formats like TIFF are ideal for printing, while JPG and PNG excel in the digital realm.

When should you use a JPG and when should you use a PNG? Or maybe you are just looking for information on which applications support the INDD file format.

TIF, PDF, and PSD are all image file formats, but unless you're a graphic designer, you probably haven't ever had a need to learn the differences between them.

The various file types and when it is suitable to utilise them are as follows:

1. JPEG (or JPG) - Joint Photographic Experts Group



You may find that JPEGs are the most prevalent file format online, and that's probably the sort of picture that's included in the Microsoft Word version of your company's letterhead. JPEGs are notorious for having "lossy" compression, which means that the picture quality is degraded as the file size becomes smaller.

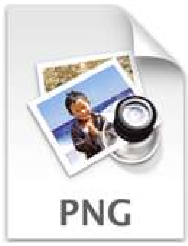
For high-resolution printing, Microsoft Office documents, the web, and more, JPEGs are an excellent choice. In order to create a project that comes out looking well, it is vital to pay attention to the resolution and the file size while working with JPEGs.

JPG vs JPEG

You may interchangeably use the.jpg and.jpeg filename extensions without any loss of quality. The file's format and behavior will remain same regardless of the name you give it.

Because early versions of Windows had a three-character restriction on filenames, the extension ".jpeg" was truncated to ".jpg," and vice versa. This is the sole reason why the same format has two different filename endings. Despite the fact that this is no longer necessary, many image editors still default of using.jpg files.

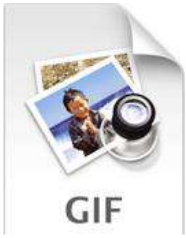
2. PNG - Portable Network Graphics



PNGs are fantastic for dynamic content like websites, but they should not be used for printed materials. PNGs are "lossless," which means they may be edited without a reduction in quality, but their resolution is still low.

The ability to store a picture with more colours on a transparent backdrop is the reason behind why PNGs are so widely utilised in web design. Because of this, the resulting picture is of considerably higher quality for use on the internet.

3. GIF - Graphics Interchange Format



The animated version of a GIF is the one that is most often seen. These animated GIFs are very popular on Tumblr sites and in banner advertisements. It seems as if we come across new pop culture GIF allusions from Giphy on a daily basis in the comments section of various social media guides. Simple GIFs may have anything from 16 to 256 colours, depending on how you define them. A smaller file size is achieved by restricting the amount of colours.

This is a typical sort of file used for online projects that need an image to load extreme rapidly as opposed to one that requires a greater degree of quality to be maintained.

4. TIFF - Tagged Image File

A TIF is a big, lossless raster file. This is a form of file that is notable for employing "lossless compression," which means that the original picture data is preserved even if the file is copied, re-saved, or compressed several times. This is a feature that sets it apart from other file types.

Even though TIFF photos may be restored to near-original quality after being altered, you should not upload them to a website in this format. Website performance will suffer since it may take a very long time to load. Its normal practise to save pictures meant for printing in TIFF format.

5. PSD - Photoshop Document

Adobe Photoshop is the gold standard when it comes to photo and image editing software, and the files it produces are known as PSDs. With "layers" in this file format, editing the picture is a breeze. The aforementioned raster file formats are created by the same application.

The fact that PSDs are only supported by Photoshop, which only supports raster pictures as opposed to vector ones, is the biggest drawback.

6. PDF - Portable Document Format

Adobe created the PDF format so that users everywhere in the world may easily share and study large amounts of data created in any program on any device. So far, they've done a good job in my opinion.

If a designer saves your vector logo in the PDF format, you will be able to examine it even if you do not have any design editing tools (as long as you have downloaded the free Acrobat Reader programme), and the designer will be able to utilise this file to make further adjustments. When it comes to sharing images online, this is the finest option available generally.

7. EPS - Encapsulated Postscript

The EPS file format is a vector format created specifically for creating high-resolution print graphics. The EPS format may be generated by the vast majority of design programmes.

The EPS extension is more of a universal file format (much like the PDF), which means that it may be used to access vector-based artwork in any design editor. This means that Adobe products are not the only ones that can read EPS files. This protects the distribution of files to designers who may not yet using Adobe products but work with software like Corel Draw or Quark.

8. AI - Adobe Illustrator Document

AI is by far the most trustworthy sort of file format for utilising photos in any kind of project, from the web to print and everything in between. It is the image format that is most favoured among designers.

Since Adobe Illustrator is the gold standard for starting from scratch when it comes to the creation of artwork, it is quite probable that this is the tool that was used to first generate your company logo. The artwork it creates is vector, the most flexible file type. All of the aforementioned file formats may be generated by it. It's the finest resource for any designer to have.

9. INDD - Adobe InDesign Document

Files produced and stored with Adobe InDesign are known as INDDs (InDesign Document). Large-scale publications, such as periodicals, magazines, and electronic books, are often designed with InDesign.

In Adobe InDesign, files from both Adobe Photoshop and Adobe Illustrator may be integrated to build content-rich designs. These designs can include complex typography, embedded graphics, page content, formatting information, and other advanced layout-related features.

10. RAW - Raw Image Formats

A RAW image has undergone the fewest transformations of any of these formats; it is often the one that is inherited by a picture for the first time. After taking a picture with your camera, the data is recorded instantly in raw format. When you transfer files to a new device and modify them in an image editor, only then will they get saved with one of the image extensions described above such .JPEG, .PNG etc.

RAW photos are significant because they capture every aspect of a photograph without subjecting it to any processing that might result in the blurring or elimination of minute visual details. However, at some point in the future, you will need to bundle them into a raster or vector file format so that they may be moved and scaled for a variety of different applications.

The accompanying photos demonstrate the wide variety of raw image file formats available, many of which are exclusive to individual cameras. An explanation of the aforementioned four raw files is as follows:

CR2: Canon developed this image extension, which stands for Canon RAW 2, specifically for use with photographs shot with one of Canon's own digital cameras. Since they are based on the industry-standard TIFF format, their quality is guaranteed from the start.

CRW: Canon was also responsible for the development of this picture extension, which came into existence before the CR2.

NEF: This file format is known as a RAW file and has a file extension that reads "Nikon Electric Format." You probably figured that Nikon cameras are responsible for its creation. If you're using a Nikon device or a Nikon Photoshop plugin, you can make significant changes to these images without having to save them as a different file format.

PEF: Pentax Digital Cameras use a RAW image file format known as Pentax Electronic Format, which is denoted by this image extension.

When it comes to working with photos, things are far more intricate than they may seem at first look. Using this manual, you should be able to choose which of the common file formats is most suited to your needs.

1.5. Image data types

24-bit colour and 8-bit colour are the most used formats for storing graphics and images.

24-bit Color Images

Each pixel in a 24-bit colour picture is represented by three bytes, generally representing the three primary colours.

The additional byte per pixel is often used to record an alpha value, which represents special effect information, making it the case that many 24-bit colour pictures are really saved as 32-bit images (e.g., transparency).

8-bit Color Image

The so-called "256 colours" that may be represented with 8 bits of colour information are widely supported by many systems.

For the purpose of storing colour information, these picture files make use of a notion called a lookup table.

Color Lookup Tables (LUTs)

A colour picker is an interface component that consists of an array of relatively big colour blocks (or a semi-continuous range of colours), which, when clicked with the mouse, allows the user to choose the colour that is indicated.

1.6. Image compression

Image compression is a sort of data compression that is done to digital photos, with the goal of reducing the costs associated with storing or transmitting such images. In order to get better results compared to those obtained using generic data compression techniques that are utilised for other digital data, algorithms may take use of visual perception and the statistical aspects that are unique to picture data.

Before beginning the processing of bigger photos or movies, image compression is a crucial first step in the area of image processing. An encoder is a piece of software that compresses photos and returns the result in a smaller file size. The mathematical transformations are an extremely important part of the process of data compression. The image-compression process may be shown as a flowchart like follows:

Figure 1.2 Flow chart of the process of the image compression[*]

We will make an effort to describe the big picture of what goes into various image compression methods. A computer's internal representation of a picture is analogous to a vector of pixels. There are a set number of bits used to represent each pixel. The color's saturation is set by these bytes (on grayscale if a black and white image and has three channels of RGB if coloured images.)

Need of Image Compression

Take a 1000x1000 pixel black and white picture where the intensity is represented by 8 bits per pixel. Therefore, the total number of bits required for each picture is 1000 *1000*8, which is 80,000,000 bits. To further illustrate, if the video has the above-mentioned types of pictures at 30 frames per second, the total bits for a 3-second movie are: 3*(30*(8, 000, 000))=720, 000, 000 bits.

---

[*]https://www.geeksforgeeks.org/what-is-image-compression/

The amount of data required to store a short 3-second movie is staggering. Therefore, we need a means of having correct representation in order to save the information about the picture in the fewest possible bits while yet maintaining the image's essential qualities. Compressing pictures is crucial for this reason.

Basic steps in image compression:

- Applying the image transform
- Quantization of the levels
- Encoding the sequences.

1.7. Colour spaces

A colour space is a predetermined layout for colour coding. In conjunction with the colour profiling that is enabled by a variety of physical devices, it enables repeatable representations of colour, regardless of whether the representation in question is analogue or digital. It is possible for a colour space to be either completely random, in which case colours are simply named and mapped onto a set of physical colour swatches, or rigorously organised, in which case colours are given discrete numbers as those found in the Pantone collection (as with the NCS System, Adobe RGB and sRGB). The term "colour space" refers to a conceptual tool that might be helpful when trying to comprehend the colour capabilities of a certain device or digital file. Color spaces reveal whether or not shadow and highlight detail and colour saturation can be preserved when rendering colours on a different device, and to what extent this is the case.

A "colour model" is a mathematical model describing the abstract way in which colours can be represented as tuples of numbers (such as in RGB or CMYK); however, a colour model without an associated mapping function to an absolute colour space is a more or less arbitrary colour system with no connection to any globally understood system of colour interpretation. The addition of a specific mapping function between a colour model and a reference colour space generates a certain "footprint" inside the reference colour space. This "footprint" is known as a gamut, and it is what defines a colour space for a particular colour model. Two absolute colour spaces based on the RGB colour paradigm are Adobe RGB and sRGB. When constructing a colour space, the CIELAB or CIEXYZ colour spaces are often used as the reference standard. These colour spaces were developed with the express purpose of including all of the colours that the typical human eye is capable of seeing.

Color models are commonly referred by their colloquial name, "colour space," which refers to a specific combination of a colour model and a mapping function. While it's true that naming a colour space will reveal the corresponding colour model, this isn't the acceptable use. For instance, the RGB colour model serves as the basis for a number of other colour spaces, but there is no such thing as the RGB colour space.

1.7.1. RGB

The abbreviation "RGB" refers to the colour space composed of red, green, and blue.

According to the RGB paradigm, every colour picture is made up of three individual pictures. Images in red, images in blue, and images in black. While one matrix is sufficient to characterise a standard grayscale picture, three are required to describe a colour image.

One color image matrix = red matrix + blue matrix + green matrix

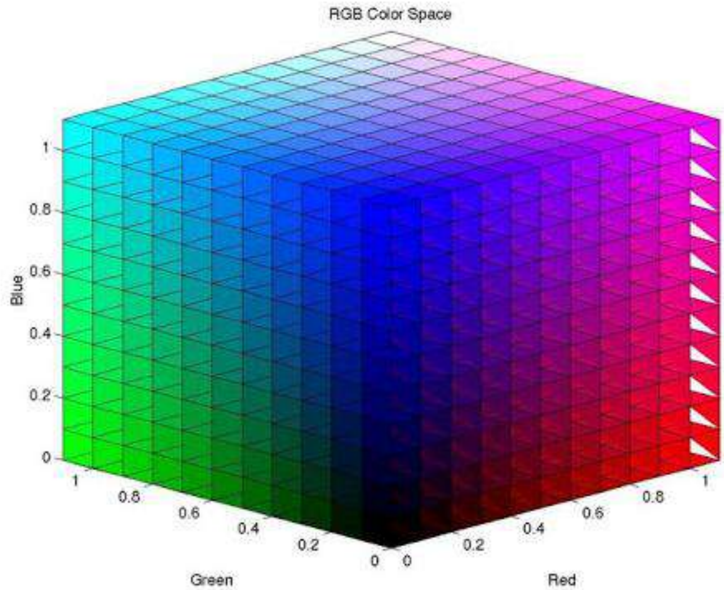
Applications of RGB

Common uses of the RGB model include:

- Cathode ray tube (CRT)
- Liquid crystal display (LCD)
- Plasma Display or LED display such as a television
- A compute monitor or a large scale screen

1.7.2. RGB to grey-scale image conversion

The average approach and the weighted method are two of the most popular ways that an RGB picture may be converted to a grayscale image. There are also a number of other methods.



Average Method

Grayscale values are calculated using the Average technique, which averages the red, green, and blue values.

```
Grayscale = (R + G + B ) / 3
```

```
Grayscale = R / 3 + G / 3 + B / 3
```

The typical approach is straightforward, but it falls short of expectations in practise. The reason for this is because the human eye has a unique response to the RGB colour space. The human eye is most sensitive to green light, with a secondary sensitivity to red and a third sensitivity to blue. This necessitates a weighted distribution with the three hues receiving different shares. Now we get to the weighted approach.

The Weighted Method

Luminosity, another name for the weighted technique, gives different amounts of importance to different colours based on their wavelength. As for the new and better formula, it goes like this:

```
Grayscale = 0.299R + 0.587G + 0.114B
```

1.7.3. Perceptual colour space

Numerous image processing tasks benefit from using a perceptual colour space. It may be used in situations where:

- A method of grayscaling a picture without changing its apparent brightness.
- Increasing the hue of the colours while keeping the apparent brightness and saturation levels the same
- Making the transitions between colours seem smooth and consistent in appearance.

Unfortunately, to the best of knowledge, while there exist colour spaces that strive to be perceptually consistent, none of them are free from substantial limitations when they are employed for image processing.

1.8. Images in MATLAB

To begin working with MATLAB, you'll need to learn how to work with arrays, which are sorted collections of real or complicated data. Images, which are ordered sequences of colours or intensities, are a logical fit for this object's representational capabilities.

In MATLAB, most pictures are stored as two-dimensional matrices, where each matrix element represents a single pixel. (The word "pixel" comes from the term "picture element," which is shorthand for a single display dot.) For instance, MATLAB would save a picture consisting of 200 rows and 300 columns of various coloured dots as a 200-by-300 matrix. This matrix would be used to represent the image.

Certain kinds of photographs, including truecolor photos, use a three-dimensional array to depict their subject matter. The red pixel intensities in a truecolor picture are represented by the first plane in the third dimension, the green pixel intensities by the second plane, and the blue pixel intensities by the third plane. Because of this standard, processing pictures in MATLAB is as straightforward as processing any other kind of numerical data, unlocking the full potential of MATLAB for image-related tasks.

1.8.1. Reading, writing and querying images

The picture data in a graphics file format is not kept as a MATLAB matrix, or even as a matrix, in its original format. Bitmap data that may be read in one continuous stream follows a header that typically contains tags with format-specific information at the beginning of most graphics files. This means you can't just use the load and save I/O commands in MATLAB to read and write images stored in a graphics file format.

In order to read and write picture data from several graphics file formats, use the appropriate MATLAB functions:

- Use `imread` to open and view images stored in various graphic file formats.
- Use `imwrite` to save a picture in a graphic file format.
- Use `imfinfo` to learn more about a picture's graphics file format.

The `imread` function can read an image from any supported graphics image file in any of the allowed bit depths. This may be done in a variety of formats. Many of the pictures you see in books are just 8 bits in size. As class uint8, they are saved when read into memory. The most important exception to this general rule is MATLAB's support for 16-bit data in PNG and TIFF pictures; if you read a 16-bit PNG or TIFF image, the data will be saved as class uint16.

The following code loads the ngc6543a.jpg image into the workspace variable RGB and then uses the image function to show the file:

```
RGB = imread('ngc6543a.jpg');
image(RGB)
```

With the `imwrite` command, image data may be written (saved). The statements

```
load clown % An image that is included with MATLAB
imwrite(X,map,'clown.bmp')
```

Make a BMP file with the clown image in it.

Writing a Graphics Image

When you save a picture using `imwrite`, the bit depth of the image will, by default, be automatically reduced to uint8. While double-precision data is useful for certain tasks, the majority of pictures used in MATLAB are 8 bits or less and may be stored in a single-precision format. Images in PNG and TIFF formats may be stored as uint16 instead of uint8, albeit this is the exception rather than the norm. You are able to change MATLAB's default behaviour by selecting uint16 as the data type for `imwrite`. This is possible since these two formats handle data with a bit depth of 16. The following code demonstrates using `imwrite` to create a 16-bit PNG file.

```
imwrite(I,'clown.png','BitDepth',16);
```

Subsetting a Graphics Image (Cropping)

It might be helpful to split up large picture files into smaller pieces or to isolate certain regions for editing. In the command line, you may provide the intrinsic coordinates of the rectangular subsection you wish to work with and then save that information to a file. If you do not know the coordinates of the corner points of the subsection, you may choose them using an interactive method, as the following example demonstrates:

```
% Read RGB image from graphics file.
im = imread('street2.jpg');

% Display image with true aspect ratio
image(im); axis image

% Use ginput to select corner points of a rectangular
% region by pointing and clicking the mouse twice
p = ginput(2);

% Get the x and y corner coordinates as integers
sp(1) = min(floor(p(1)), floor(p(2))); %xmin
sp(2) = min(floor(p(3)), floor(p(4))); %ymin
sp(3) = max(ceil(p(1)), ceil(p(2)));   %xmax
sp(4) = max(ceil(p(3)), ceil(p(4)));   %ymax

% Index into the original image to create the new image
MM = im(sp(2):sp(4), sp(1): sp(3),:);

% Display the subsetted image with appropriate axis ratio
figure; image(MM); axis image

% Write image to graphics file.
imwrite(MM,'street2_cropped.tif')
```

You may avoid using `ginput` in the previous example by manually defining sp with the picture corner coordinates should be used.

Obtaining Information about Graphics Files

Using the imfinfo function, you may learn more about image files in any of the common formats we've already covered. The information that you acquire will vary depending on the kind of file; nevertheless, it will always comprise at least the following components:

- Format of the File

- Version of the file format.
- The Time of Last Edit for a File
- Measurement of a file's size in bytes
- Size of image's width in pixels
- Height of the image in pixels
- Per-pixel bit count
- Types of images include indexed, intensity (grayscale), and RGB (truecolor).

1.8.2. Accessing pixel values

Using the impixel function, you may get the values of specific pixels in an image and have them stored in a variable. Either by supplying the coordinates of the pixels as input parameters or by selecting the pixels with a mouse in an interactive manner, you may choose which pixels to use to describe the image. The impixel command stores the pixel value in a MATLAB workspace variable.

1.8.3. Converting image types

Besides the standard colour images, Image Processing ToolboxTM also works with binary, indexed, grayscale, and truecolor images. Pixels are stored differently in each picture format. For instance, truecolor pictures show a pixel as a triplet of values for the colours red, green, and blue, while grayscale photos display a pixel as a single value for the intensity of the colour it depicts.

Floating-point, signed, and unsigned integers, and logical data types may all be used to store the pixel values of various picture kinds. Functions in the toolbox let you transform data and picture formats with ease.

You may convert a picture from one kind to another by using one of the numerous functions that are included in the toolbox. To filter a colour picture that has been saved as an indexed image, for instance, you must first convert it to truecolor format. When the filter is applied to the truecolor picture in MATLAB, the intensity values in the image are filtered in a manner that is suitable for the filter. However, MATLAB will just apply the filter to the indices in the indexed image matrix, which might provide illegitimate results if you try to filter the indexed picture.

Certain transformations may be performed with nothing but MATLAB syntax. By appending three copies of the original matrix along the third dimension, for instance, a grayscale picture may be converted to truecolor format.

```
RGB = cat(3,I,I,I);
```

The resultant truecolor picture contains the same colour matrix for each of the red, green, and blue channels, rendering the image grayscale.

In addition to these image type conversion methods, there are a number of additional functions as part of the action that they carry out, return a different image type. To mask a picture for filtering or other processes, for instance, you may utilise the binary image returned by the area of interest algorithms.

CHAPTER

2 Formation

2.1. How is an image formed?

The study of image generation takes into consideration the radiometric and geometric processes that are responsible for the production of 2D pictures of 3D objects. Analog to digital conversion and sampling are also key parts of the picture generation process in the case of digital images.

To image anything is to transfer it onto a flat surface. There is a one-to-one correspondence between the picture and the real thing. In order to form a picture, a lens will gather light that has been scattered from a lit object and focus it into a sharp point. Magnification is defined as the comparison between the picture height and the actual object height. Lens field of view is dependent on picture surface size and focal length. These mirrors have a focal length equal to one-half their centre of curvature, making them ideal for image generation due to their curved surfaces.

Formation of a digital image

Considering that taking a photo with a camera is a physical procedure. Direct solar radiation is captured and converted into usable electricity. The picture takes using a sensor array. Consequently, when the item is illuminated by sunlight, the sensors detect the quantity of light that is reflected by the object, and a continuous voltage signal is created based on the amount of data that is detected. We must digitize this information in order to use it in the production of a digital picture. This requires quantization and sampling. After being subjected to sampling and quantization, a digital picture is reduced to a two-dimensional array or matrix of integers.

2.2. The mathematics of image formation

Capturing, storing, and retrieving images from a variety of sources have all been significantly improved because of advancements in Image Processing. Image Restoration, Image Segmentation, Image Enhancement, De-Blurring, and De-noising are among the Image Processing activities that are utilised most often. Imaging methods such as angiography, magnetic resonance imaging (MRI), Arterial spin labelling (ASL), computerised tomography (CT), deep brain stimulation (DBS), electroencephalography (EEG), etc. all make good use of such images in different ways. Nonetheless, mathematics has been important in the aforementioned Image Processing applications. However, one aspect of Imaging Technology that has remained crucial despite the many innovations and fast advancements is the use of mathematics. It has been noted that there is a close mathematical relationship between image processing and its related fields. The fundamental mathematical techniques of histogram equalisation, probability and statistics, discrete cosine transforms, fourier transforms, differential equations, integration, matrix, and algebra are used in many of the image processing techniques. Matlab is one of the tools that is used the most often by academics working in the field of image processing because of its computational capabilities. SciLab, GNU Octave, SageMath, etc., are a few more widely used tools.

A specialised picture viewer is needed when working with images in mathematics. Students need to be able to comprehend the picture on a visual and numerical level. The link or connection between these two elements of digital pictures is one of the first things students need to learn and should be one of the first things they learn. So, our "Pixel Calculator" app displays digital photos as both grey-valued pictures and arrays of numbers. When the student uses a tool to magnify the picture and zooms in on it, the pixel values appear numerically overlaid on the grey values when they reach a specific degree of magnification.

The fact that a digital picture may be seen as both a mathematical object and a visual object at the same time contributes to the attractiveness of using digital images in the context of mathematics instruction. Though it is composed of numbers in a two-dimensional array, a picture may stand for nearly anything in a student's real world. This opens the door for students from all walks of life and with all sorts of interests to enter the world of mathematics in a welcoming, safe environment. Related research utilises both moving and static photographs to investigate issues in the sciences.

To further emphasize the correlation between pixel values and levels of brightness, a mechanism of adjusting pixel values is offered. To the user, it looks like a calculator that fits in their pocket. However, the value of a chosen pixel may be seen by pressing the # key, which is a special symbol. A screenshot of the Pixel Calculator user interface.

In the Pixel Calculator, the four basic arithmetic operations of addition, subtraction, multiplication, and division are put to use in unique and interesting ways. You can brighten a picture by adding to it, and darken it by subtracting. To increase contrast, multiply by a positive number, and to decrease contrast and darken by the same amount, divide by a negative number. Combinations of these procedures allow for much nuanced contrast regulation.

Many students have shown an interest in a certain group of image-altering operations such as scaling, rotation, reflection, distorting, and translation. The METIP interactive learning environment offers two distinct approaches for the specification of geometric transformations. The first approach uses formulae, which, when applied, create a graphical representation of the connection that exists between the source picture and the destination image. The second uses a geometric interface that allows for direct manipulation, with control lines serving as "handles" to shape a geometric change.

Geometric transformation formulae may mix and match a wide range of functions, and they can make use of either Cartesian or polar coordinate systems to refer to the resulting picture. We see the original picture alongside two transformation formulae that apply to it, each of which uses polar coordinates to perform its transformation.

Students may define geometric distortions using control lines since it is easy and simple to do so, but the lines can also be specified symbolically. Because of this, it is now feasible to achieve a high level of control over the transformation and to conduct quantitative research on the impacts of the transformation.

2.2.1. Linear imaging systems

Convolution and Fourier analysis are the two methods that underpin linear image processing, the same two methods that underpin ordinary digital signal processing. Convolution is the more crucial of these two operations due to the fact that the information that constitutes a picture is stored in the spatial domain rather than the frequency domain. The sharpening of object edges, the reduction of random noise, the correction of uneven lighting, the deconvolution of blur and motion, and so on are all examples of the ways in which pictures may be enhanced by linear filtering. In these processes, the original picture is convolved with the filter kernel to generate the filtered image. Image convolution presents a number of significant challenges, one of the most significant being the vast amount of computations that need to be carried out, which often results in unacceptably lengthy execution times.

In addition, convolution by separability and FFT convolution, two essential strategies for speeding up execution, are outlined.

Convolution of images operates in the same manner as convolution in a single dimension. Images, for instance, might be thought of as the sum of impulses, or scaled and shifted delta functions. Equally, the impulse responses of linear systems are used to describe them. As one may guess, the system's output picture is the same as the input image convolved with the impulse response of the system.

The picture that represents the two-dimensional delta function is made up entirely of zeros, with the exception of a single pixel located at row = 0 and column = 0 that has a value of one. For the time being, let's pretend that the row and column indexes may take on both positive and negative values, making one the island among a wide ocean of zeros. The delta function's singular nonzero point is transformed into a new two-dimensional pattern when it is introduced into a linear system. The impulse response is also known as the point spread function (PSF) in the field of image processing due to the fact that the only thing that can happen to a point is that it spreads out.

As a prime illustration of these ideas, consider the human eye. A picture, first portrayed as a pattern of light, is converted into a pattern of nerve impulses by the retina's primary layer. A neural picture is processed by the retina's second layer and sent on to the optic nerve fibres in the retina's third layer. Visualize a tiny point of light in the middle of a pitch-black backdrop as the picture being projected onto the retina. So, the eye receives a stimulus in the form of a visual impulse. If we make the assumption that the system is linear, we can figure out the picture processing that is going on in the retina by looking at the image that is produced by the optic nerve. In other words, we're looking for the processing's point spread function.

Convolution by Separability

As long as the PSF can be split, this method may be used for quick convolution. If the PSF can be decomposed into two one-dimensional signals, such as a vertical and a horizontal projection, we say that it is separable. Separate images, such as the square point-spread-function (PSF) are examples of this kind of image. Each pixel's value is determined by multiplying its corresponding horizontal projection point by its corresponding vertical projection point. Specifically, this is how the numbers look:

$$x[r,c] = vert[r] \times horz[c]$$

The original two-dimensional picture is denoted by x[r, c], whereas the resulting one-dimensional projections are denoted by vert[r] and horz[c]. Obviously, this is not true of the vast majority of pictures online. In this case, the pillbox is not detachable. To be sure, the number of pictures that may be broken apart is endless. This may be grasped by creating completely random horizontal and vertical projections and locating the corresponding images. Profiles with two exponential terms. Then, using Equation, we can locate the picture that best represents these profiles. When the picture is presented, it takes on the form of a diamond as the distance from the origin rises, gradually becomes smaller and smaller until it finally disappears.

The pillbox or other circularly symmetric PSF is excellent for most image processing jobs. It is preferable to make the identical adjustments in all directions to the digital picture, despite the fact that they are often stored and processed in a rectangular format of rows and columns. The issue this poses is whether or not a PSF exists that is both circularly symmetric and divisible. As for the distribution, the answer is "yes," however there is just the Gaussian distribution. In the case of a two-dimensional Gaussian picture, the projections are likewise Gaussians. The standard deviation of the image Gaussian and the projection Gaussian are the same.

FFT Convolution

Inconvenient though it may be, the Fourier transform is the most efficient method for convolving a picture with a big filter kernel. For instance, the FFT is around 20 times quicker than traditional convolution when applied to the task of convolving a 512 x 512 picture with a 50 x 50 PSF. The transition to two dimensions is really straightforward.

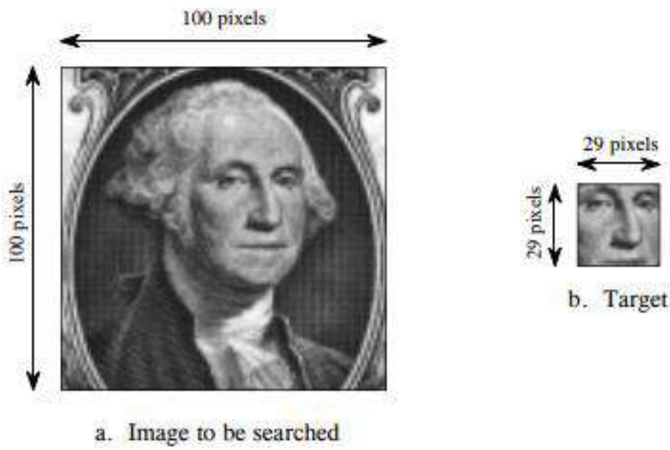
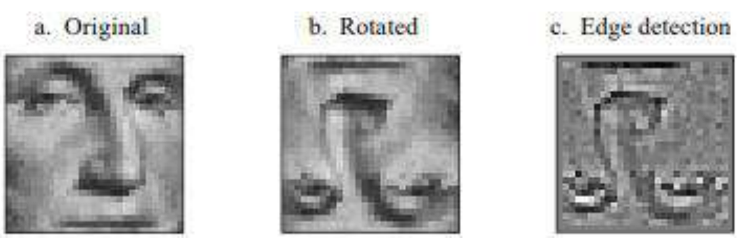


Figure 2.1 Target detection

We'll show you how FFT convolution works by using it as an example; it's a technique for finding a certain pattern in a picture. Let's pretend we set out to create a method of evaluating banknotes worth one dollar, whether for the purpose of ensuring the quality of the printed product, sniffing out counterfeits, or checking the legitimacy of a purchase made at a vending machine. A picture of the banknote of 100 × 100 pixels is obtained, with the focus placed on the likeness of George Washington, as illustrated in Figure 2.1. The purpose is to look for a certain pattern inside the picture; in this case, a face within the 29x29 pixel

*https://www.analog.com/media/en/technical-documentation/dsp-book/dsp_book_Ch24.pdf area. In other words, given a picture and a known pattern, how can we most efficiently pinpoint where the pattern exists in the image? Correlation (a matching filter) is the answer to this issue, and it may be achieved via convolution.

There are two tweaks that must be made to the target picture before the real convolution can take place and produce a PSF. The components of these are shown in Fig. 2.2. The signal of interest, shown in (a), is the one we want to identify. Image (b) has been rotated by 180 degrees, which is the same as flipping it left-to-right and then upside-down. Because of the reversal that takes place during convolution, the target signal must be inverted in order to do correlation using this method.



[*] *Figure 2.2 Development of a correlation filter kernel*

The second change is an optimization technique for the method. It is more effective to try to detect the margins of the face in the borders of the original picture as opposed of attempting to identify the face itself in the original image.

---

[*] https://www.analog.com/media/en/technical-documentation/dsp-book/dsp_book_Ch24.pdf

This is because the correlation peak is now more pronounced than it was with the initial characteristics since the edges are sharper. Taking this extra step is optional but highly recommended. Before performing the correlation, the original picture and the target signal both have a 3x3 edge detection filter applied to them. This is the simplest possible implementation of the technique. The associative feature of convolution shows that this is equivalent of applying the edge detection filter twice to the target signal while preserving the original picture. In most cases, a single application of the 3x3 kernel for edge detection is all that is necessary. In Figure 2.2, point (b) becomes point (c) as a result of this modification. Due to this, (c) the PSF is suitable for use in the convolution.

2.1.1 The Dirac delta or impulse function

The signal of an impulse consists entirely of zeroes with the exception of a single nonzero value. Thus, impulse decomposition allows for a sample-by-sample analysis of signals. The basic notion of digital image processing (DIP) was also introduced. How the input signal is broken down into simple additive components, how each of these components is then processed by a linear system, and how the output components are then synthesized. The generated signal is the same as if the original signal had been sent into the system without any division or combining. Although there are several decompositions available, the impulse decomposition and the Fourier decomposition are the workhorses of signal processing. In the context of impulse decomposition, the process may be represented as a convolution in mathematics. Signals in a continuous time domain may also be convolved with, however the corresponding math is more involved.

The first one is the δ[n] delta function, named after the Greek letter (delta). The delta function is a normalised impulse in which the value of the first sample, at index zero, is one and the values of all subsequent samples are zero. Because of this, the delta function is sometimes referred to as the "unit impulse."

Input of a delta function (unit impulse) produces an output called the impulse response. Impulse responses will be different amongst systems if there are significant differences between them. Like the input and output signals, which are often denoted by x[n] and y[n], respectively, the impulse response is typically represented by the symbol h[n]. One may easily replace this to a more accurate label; for instance, a filter's impulse response could be called f[n].

In mathematical terms, every jolt can be written as a delta function with certain shifts and scaling. Let's say a[n] is a signal and it consists entirely of zeros with the exception of the eighth sample, which has a value of -3. For comparison, consider a delta function with a rightward shift of 8 samples and a multiplier of -3. Put another way: a[n]'-3 δ[n-8]. This notation is used practically in all DSP equations, therefore familiarity with it is essential.

What will be the output of a system if it receives an impulse as its input, such as the value -3 δ[n-8]? Homogeneity and shift invariance come into play here. When the input is scaled and shifted, the output is also scaled and shifted in exactly the same way. If δ[n] produces the output h[n], then it follows that -3 [n-8] produces the result -3h[n-8]. The output is the impulse response modified by the same shift and scaling as the delta function applied to the input. Knowing the impulse response of a system allows you to predict how it will respond to a given stimulus.

2.2.2. The point-spread function

How an imaging system reacts to a point source or object is defined by its point spread function (PSF). The PSF is the impulse response of a focused optical system, however the phrase "system's impulse response" may be used to describe the PSF in a more generic sense. In many situations, the PSF may be understood as the enlarged blob in an image that stands in for a single point. In terms of the functionality it provides, it is the imaging system's spatial domain counterpart of the optical transfer function. It is an important notion in Fourier optics, as well as in astronomical imaging, medical imaging, electron microscopy, and other imaging methods, such as 3D microscopy (such as in confocal laser scanning microscopy), and fluorescence microscopy.

The quality of an imaging system may be evaluated based on the degree to which the point object is stretched out, also known as blurring. The process of image production in non-coherent imaging systems, such fluorescence microscopes, telescopes, and optical microscopes, may be explained using linear system theory and is linear in terms of picture intensity. To put it another way, if we take pictures of A and B at the same time, we end up with a picture that is the same as the sum of those pictures taken separately. Basically, because photons don't interact with one another, photographing subject A won't change how the image subject B, and vice versa. The image of a complex object is the convolution of the real object and the point spread function (PSF) in a space-invariant system, where the PSF is the same in all directions in the imaging space. From diffraction integrals, the PSF may be calculated.

Since optical non-coherent imaging systems have the advantage of linearity, i.e.

$$\mathrm{Image}(\mathrm{Object}_1 + \mathrm{Object}_2) = \mathrm{Image}(\mathrm{Object}_1) + \mathrm{Image}(\mathrm{Object}_2)$$

In order to calculate the image of an object in a microscope or telescope, the object-plane field must first be expressed as a weighted sum over 2D impulse functions, and the image plane field must then be expressed as a weighted sum over the images of these impulse functions. We call this the superposition principle, and it holds true for all linear systems. In certain disciplines of mathematics and physics, they may be referred to as Green's functions or impulse response functions. Point spread functions relate to the images of the individual object-plane impulse functions, which represent the fact that a mathematical point of light in the object plane is spread out to produce a finite area in the image plane.

The picture is calculated by adding the PSFs of the individual points that make up the object after it has been segmented into points of varied intensities. Because the point spread function (PSF) is often totally defined by the imaging system, it is possible to characterize the whole picture simply knowing the optical parameters of the system. A convolution equation is often used to describe this imaging procedure. In order to use deconvolution to return an image to its original state, understanding the point spread function (PSF) of the measurement equipment is crucial in fields like astronomy and microscope image processing. When dealing with laser beams, the PSF may be mathematically represented utilizing the ideas of Gaussian beams. For example, deconvolution of the modelled point spread function (PSF) and the picture enhances feature visibility and eliminates imaging noise.

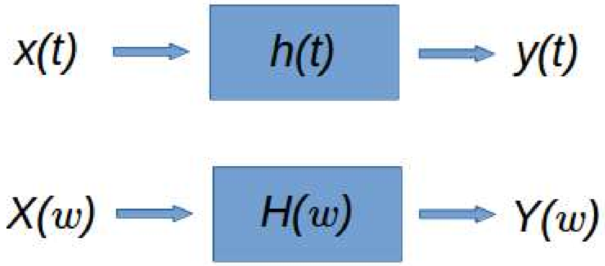
2.2.3. Linear shift-invariant systems and the convolution integral

System with linear temporal invariance (LTI) and linear shift invariance (LSI). Both the linear time invariant system and the shift-invariant system are essentially the same thing, referring to either an analogue or a discrete-time system, respectively.

The result is always identical to the superposition of the outputs that we obtained when we applied each input since the system is linear and every time we describe an input as the sum of simpler signals. The shift-invariant is identical to the time-invariant, meaning that if we delay the input, the output will be the signal's original input that wasn't delayed. There is no time-dependent change in the system regardless of the delay we choose.

The LSI/LTI system's user-friendliness may be attributed to two main features. Since the system is linear and invariant, it can be easily manipulated; as the saying goes, "the output of the system is just the convolution of the input to the system with the system's impulse response."

Two characteristics that help to define LTI/LSI systems are the impulse response and the frequency response. They provide two distinct approaches to determine what the output of the system will be in response to a particular input signal.



The input signal x(t) is transformed into the desired output signal y(t) by the system h(t). Let's take a closer look at the crucial two characteristics:

Linear:
We can accomplish superposition. The input of a linear system is the sum of the signals, thus the name. As a result, the system may handle each signal independently and then combine the results:

If the output of x1(t) translates to that of y1(t) and that of x2(t) to that of y2(t), then for all values of a1 and a2,

$$h\{a_1 x_1(t) + a_2 x_2(t)\} = a_1 y_1(t) + a_2 y_2(t)$$

Time-invariant
The features of the system remain constant throughout time. It is also the case that it does not make a difference where the beginning point of the coordinate system is situated while speaking about spatial invariance.

If we introduce a lag into the input, that lag will be replicated in the final product. If there is a mapping between an input signal x(t) and an output signal y(t), then for all possible values of τ,

$$h\{x(t-\tau)\} = y(t-\tau)$$

Due to these properties, it is possible to describe the functioning of the system by utilising its impulse and frequency responses. They provide two distinct ways of looking at the system, each of which has its uses.*

2.1.2 Convolution: its importance and meaning

Convolution is a mathematical operation in mathematics (more specifically, functional analysis) where two functions (f and g) are multiplied together to get a third function (f * g) that explains how the form of one function is altered by the other. Both the final function and its calculation are known by the same name: convolution. It is the integral of the product of the two functions, with one of them flipped and shifted. The convolution function is the result of evaluating the integral for all possible values of shift.

Some features of convolution are similar to cross-correlation: for real-valued functions, of a continuous or discrete variable, it differs from cross-correlation ($f * g$) only in that either $f(x)$ or $g(x)$ is reflected about the y-axis; thus it is a cross-correlation of $g(-x)$ and $f(x)$, or $f(-x)$ and $g(x)$. For complex-valued functions, the cross-correlation operator is the adjoint of the convolution operator.

The fields of probability, statistics, acoustics, spectroscopy, signal processing, image processing, geophysics, engineering, physics, computer vision, and differential

*https://www.bogotobogo.com/OpenCV/Impulse_response_frequency_response_linear_time_invariant_LTI_linear_shift_invariant_LSI_Convolution.php equations are only some of the applications that may be found using convolution.

Functions on Euclidean space and other groups may be used to define the convolution. The discrete-time Fourier transform, among other periodic functions, may be defined on a sphere and convolved via periodic convolution. For functions on the set of integers, one may define a discrete convolution.

Applications of convolution generalizations may be found in signal processing, where they are used in the design and implementation of finite impulse response filters, as well as in the fields of numerical analysis and numerical linear algebra.

Deconvolution refers to the process of carrying out the computation that is the inverse of the convolution operation.

2.3. The engineering of image formation

Capturing equipment, such as cameras, perform image creation, which is analog-to-digital conversion of a picture using 2D Sampling and Quantization algorithms. The 3D world is often presented to us in a 2D format.

The analogue picture was also formed in this manner. It is essentially the process of converting the three-dimensional world that constitutes our analogue picture into the two-dimensional world that constitutes our digital image.

Sampling and quantizing the analogue signals often requires a digitizer or frame grabber.

Imaging:

Imaging is the process of transforming a physical thing in the real world into a flat, two-dimensional digital picture. In order to do this, it is necessary for every point on the 3D object to be in perfect alignment with the picture plane. Light is reflected from everything we can see, and this allows us to record the whole scene on the picture plane.

Image quality relies on a number of elements, including the lens and space in which the photo was taken.

Color and Pixelation:

A frame grabber, functioning similarly to a sensor, is located at the picture plane in digital imaging. Its purpose is to collect light and concentrate it on the item, however the reflected light from the 3D object causes the continuous picture to become fragmented. When light is directed to a sensor, it creates an electrical current.

It all comes down to how much light is sampled and quantized in order to make an electrical signal, which in turn determines whether or not each resulting pixel will be coloured.

A computer picture may be created from these individual pixels. The quality of a picture is determined by the number of these pixels. The higher the density, the sharper and more detailed the resulting picture.

Forming a Digital Image:

It is necessary to have a process that continuously converts data into a digital format in order to be able to construct or produce a picture that is digital in its nature. The following are the two primary procedures:

- **Sampling (2D):** Sampling is the digital image's equivalent of a physical resolution scale. The quality of the digitised picture is in direct proportion to the sample rate. In image processing, the size of the sampled picture is quantified by a numerical number. It has something to do with the values of the image's coordinates.
- **Quantization:** The quantization of a digital picture is the total number of greyscale values it contains. Quantization describes the process by which the picture function's continuous values are transformed into their discrete digital representation. It's connected to how bright or dark a picture is.
- When trying to gain the fine shading features of a picture, a typical human being will eventually acquire a high degree of quantization levels. In general, the more quantization levels there are, the more distinct the picture will be.

2.3.1. The camera

Put simply, a camera is an optical device used to record images. At its most fundamental, a camera is just a sealed box (the camera body) with a tiny opening (the aperture) in it that lets light in and creates a picture on a light-sensitive sensor (usually a digital sensor or photographic film). To regulate how light reaches the camera's photosensitive element, cameras use a wide range of techniques. Light entering a camera is concentrated by lenses. To adjust the size of the opening, just turn the ring. The exposure period of a photosensitive surface is controlled by a shutter mechanism.

In the field of photography, the still-image camera is the primary tool. Photographs, digital pictures, and photographic prints are all methods that may be used to create copies of previously captured images. Film, videography, and cinematography are all related creative disciplines that use moving-image cameras.

The earliest instrument for projecting an image onto a flat surface, known in Latin as a camera obscura, is where the term "camera" originates (literally translated to "dark chamber"). The camera obscura was the precursor of the modern photographic camera. In 1825, Joseph Nicéphore Niépce took the first image that could be kept forever.

Cameras typically only record images in the visible light spectrum, but there are other cameras that can record images in the infrared and other invisible parts of the electromagnetic spectrum.

Light is allowed to enter an enclosed box through a converging or convex lens, and a picture is then captured on a light-sensitive media. This fundamental design is used in every single camera. The amount of light that enters the camera is controlled by a shutter.

The scene to be recorded may be seen in the viewfinder of most cameras, and the camera's focus, aperture, and shutter speed can be adjusted in a number of ways.

2.3.2. The digitization processes

Information is "digitised" when it is transformed into a digital format that can be read by computers. By creating a sequence of integers that define a distinct collection of points or samples, an object, picture, sound, text, or signal (often an analogue signal) may be represented. As a consequence, the object is represented as a digital picture and the signal is represented as a digital form. Digitizing simply implies "the conversion of analogue source material into a numerical format," thus the numbers might be decimal or any other system. However, in practise, the digitised data is in the form of binary numbers, which aids processing by digital computers and other processes.

The ability to digitise "information of any sort in any format" is vital to the efficiency and interoperability of data processing, storage, and transfer. Digital data has the ability to be more readily shared and retrieved and may be transmitted endlessly without generation loss so long as it is moved to new, stable forms as necessary, but analogue data is often more stable. Because of this opportunity, there has been a surge in the number of initiatives aimed at digitizing and making more widely available the works of cultural institutions.

In certain cases, people confuse digital preservation with digitalization. Digitization is frequently the first and most important stage in digital preservation, although the two processes are distinct. The digitization of artefacts is an important preservation method for libraries, archives, museums, and other memory institutions, as well as a means to expand the availability of their collections to the public. Information professionals face new issues as a result of this, and the range of possible solutions is as broad as the organisations that need to address them. The data on certain audio and video cassettes, for example, might be lost forever if they are not digitised before their expiration date due to technology obsolescence and medium degradation.

Time, money, cultural heritage worries, and providing an equal forum for historically underrepresented perspectives are just some of the problems and repercussions of digitalization. The majority of organisations that are digitising their operations have come up with their own methods for overcoming these obstacles.

There has been a lack of consistency in the outcomes of large-scale digitization initiatives throughout the course of their existence; yet, several organizations have achieved their goals, even if not in the manner that is often associated with Google Books.

Due to the rapid pace at which technology may evolve, it can be challenging to maintain up-to-date digital norms. Attending professional conferences and participating in relevant organisations and task forces are two great ways for experts in a certain industry to stay abreast of developments and contribute to ongoing debates.

2.3.3. Noise

Image noise, a sort of electrical noise, is the random fluctuation of an image's brightness or colour information. The image sensor and electronic circuitry of a scanner or digital camera are both capable of producing it. Film grain and the inevitable shot noise of a perfect photon detector are two more sources of image noise. Noise in captured images is an unwelcome by-product of the imaging process that detracts from the quality of the final output.

The term "noise" originally referred to as "unwanted signals," as in the variations in electrical signals that were received by AM radios that resulted in audible and distracting acoustic noise ("static"). Noise is often used to describe undesirable electrical disturbances.

Noise in photographs may vary from being undetectable in a well-lit digital snapshot for being the dominant feature in optical and radio astronomical images, from which only a little amount of information can be extracted by complex processing. It is undesirable for there to be such a high amount of noise in a picture since it would be hard to identify the topic of the image.

Types

Gaussian noise

The majority of Gaussian noise in digital photos is produced as the picture is being captured. Both the ambient light and the sensor's internal temperature contribute to noise, and the electrical circuits that are linked to the sensor provide additional noise.

A standard model of picture noise is Gaussian, additive, independent at each pixel, and independent of the signal intensity. This kind of noise is created mostly by Johnson–Nyquist noise, also known as thermal noise. Other sources of noise, such as that which originates from the reset noise of capacitors, also contribute to image noise ("kTC noise"). The "read noise" of an image sensor, or the consistently high noise level in shadowy regions, is mostly contributed by the amplifier. When using a colour camera, additional noise may appear in the blue channel if the blue channel is amplified more than the other two channels (green and red). However, shot noise takes over as the dominant form of image sensor noise during longer exposures; it is neither Gaussian nor signal-independent. Further, a wide variety of Gaussian de-noising techniques are available.

Salt-and-pepper noise

Salt-and-pepper noise, sometimes known as "impulsive" noise, is another name for "fat-tail" noise. In a picture with salt-and-pepper noise, black pixels will be found in white areas and white pixels will be found in black areas. Mistakes in the analog-to-digital conversion process, bit errors during transmission, and other similar phenomena may all contribute to this form of background noise. Dark frame removal, median filtering, combined median and mean filtering, and interpolating around dark/bright pixels may effectively get rid of it.

A similar, but non-random, display is produced by dead pixels in an LCD monitor.

Shot noise

It is common for statistical quantum fluctuations, or variations in the number of photons perceived at a particular exposure level, to account for the majority of the noise in the brighter areas of a picture captured by an image sensor. Photon shot noise is the name given to this kind of background radiation. There is no correlation between the noises occurring at different pixels, and the root-mean-square value of shot noise is equal to the square root of the picture intensity. There is a Poisson distribution for shot noise, which is fairly close to the Gaussian distribution except at very high intensities.

In addition to the noise caused by photons being fired, there is also the possibility of extra shot noise being caused by the dark leakage current in the image sensor; this noise is frequently referred to as "dark shot noise" or "dark-current shot noise." "Hot pixels" inside the picture sensor have the highest levels of dark current. By subtracting (using "dark frame subtraction") the dark charge of normal and hot pixels, we are left with simply the shot noise, or random component, of the leakage. Noise is more than simply shot noise and hot pixels show as salt-and-pepper noise if dark-frame subtraction is not performed or if the exposure duration is long enough for the hot pixel charge to surpass the linear charge capacity.

Quantization noise (uniform noise)

Quantization noise is the artefact of reducing the grayscale of a perceived picture to a finite set of levels. It seems to be spread out about evenly. Even while it is possible for it to be signal dependent, it will be signal independent if there are other noise sources that are significant enough to create dithering or if dithering is intentionally introduced to the signal.

Film grain

Film grit is a kind of signal-dependent noise whose statistical distribution is very close to that of shot noise. The amount of dark silver grains in an area will follow a binomial distribution if film grains are randomly distributed (same number per area) and each grain has an equal and independent likelihood of evolving into a dark silver grain after absorbing photons. For low-probability regions, this distribution will seem very similar to the well-known Poisson distribution of shot noise. For many purposes, the simplicity and robustness of the Gaussian distribution make it the model of choice.

Film grit is often thought of as a roughly isotropic (non-oriented) noise source. The randomness of the silver halide grain dispersion in the film only exacerbates the impression.

Anisotropic noise

Some sources of noise have a discernible orientation in the pictures they appear in. For example, image sensors are occasionally vulnerable to row noise or column noise.

Periodic noise

Periodic noise is sometimes introduced into a picture by electrical interference that occurs as the image is being captured. Periodic noise modifies a picture such that it seems to have a repeating pattern superimposed over it. This noise appears as single spikes in the frequency domain. Using notch filters in the frequency domain may significantly reduce this noise.

Noise reduction against detail preservation is an application-dependent trade-off. For instance, low pass filtering might be a good choice if you don't care about the castle's fine features. If it is believed that the minute features of the castle are significant, a workable approach may consist of completely removing the image's border from the picture.

CHAPTER 3

Pixels

3.1. What is a pixel?

Pixels are the smallest displayable portion of an electronic picture or graphic.

In computer graphics, a pixel is the fundamental building block. Everything you see on a computer screen is made up of pixels. This includes images, videos, and text.

One alternative name for a pixel is a picture element (pix = picture, el = element).

On the display screen of a computer monitor, a pixel may be represented by either a dot or a square. Geometric coordinates are used to construct pixels, the fundamental elements of any digital picture or display.

The display resolution is a measurement of the number of pixels on the screen as well as their quantity, size, and colour combination. This measurement is dependent on the graphics card as well as the display monitor. As an example, a computer with a display resolution of 1280 x 768 can generate no more than 98,3040 pixels on a display screen.

More pixels per inch of monitor screen provide better visual output, and this is determined in part by the pixel resolution spread. A photograph with a resolution of 1920 x 1080, for instance, has a pixel count of 2,073,600, making it 2.1 megabytes in size.

In terms of actual size, a pixel may be any number of different sizes, depending on the screen's native resolution. If the display is set to its highest possible resolution, it will be the same size as the dot pitch, and if the resolution is set lower, it will be bigger since each pixel will require more dots. As a result, it's possible to make out individual pixels, resulting in the "pixelated" appearance of blocks and chunks.

Each pixel is placed in a regular two-dimensional grid, however other sampling arrangements are also possible. For instance, in liquid crystal display (LCD) panels, the three primary colours are sampled at various points of a staggered grid, but digital colour cameras utilise a grid that is more regular.

Pixels on computer displays are typically square, with uniform sampling pitch throughout the vertical and horizontal axes. A pixel's shape is rectangular in other systems, such as the anamorphic widescreen format of the 601 digital video standard.

Most modern high-end display devices can project millions of distinct colours, each of which has its own unique logical address, size of eight bits or more, and a logical address. Each pixel's colour is calculated by carefully balancing the three primary colours that make up the RGB colour space.

It's possible to declare each pixel's colour with a varying amount of bytes, depending on the colour scheme in use. For instance, the number of colours available in 8-bit colour system is severely constrained due to the fact that only one byte is dedicated every pixel.

Three bytes are allotted, one for each colour of the RGB scale, in the conventional 24-bit colour systems used for practically all PC monitors and smartphone displays. This results in a total of 16,777,216 colour combinations. Each of the three primary colours, red, green, and blue, receives 10 bits in a 30-bit deep colour system, resulting in a staggering 1.0731015 possible colour combinations.

But because the human eye can't tell the difference between more than ten million colours, greater colour variations can cause problems with colour banding rather than adding more information.

3.2. Operations upon pixels

Point operations, filtering, and image transformations are just a few of the methods often employed by scientists and engineers to modify or analyse digital pictures. The subfields of computer vision and pattern recognition often intersect with digital image processing (IP), and the methods of segmentation, classification, and difference analysis are used in the processing of images in these subfields. They are all based on the same foundational IP procedures.

In the process of image processing known as point operations, each pixel in the output picture is solely reliant upon the matching pixel in the image that was fed into the system. Generally speaking, a point operation is any arithmetic or logical action done on a single picture or between two images of the same size that are compared on a pixel-by-pixel basis.

3.2.1. Arithmetic operations on images

In image arithmetic, two or more pictures are processed using a logical or mathematical operation. This means that the value of a pixel in the output picture is determined only by the values of the corresponding pixels in the input images, as the operators are applied on a pixel-by-pixel basis. Therefore, the pictures should all have the same proportions. In spite of being the most basic kind of image processing, image arithmetic has several practical uses. With arithmetic operators, you may accomplish a lot in a short amount of time since the procedure is straightforward.

The input pictures may undergo arithmetic operations such as addition, subtraction, and bitwise arithmetic (AND, OR, NOT, XOR). These manipulations may assist improve the quality of the supplied photographs. Image arithmetic is crucial for investigating the characteristics of the provided images. The operated pictures may then be utilised as an upgraded input image, and many additional operations can be done to the image to perform tasks like as clarifying, thresholding, dilation, and so on.

Addition of Image:

Using the **cv2.add()** function, we can combine two images. This is a straightforward method of adding up the pixel values in the two photos.

```
Syntax: cv2.add(img1, img2)
```

However, it is not optimal to add the pixels. We thus use **cv2.addweighted()**. Keep in mind that the width and height of each picture should be proportional.

Syntax: *cv2.addWeighted(img1, wt1, img2, wt2, gammaValue)*
Parameters:
img1: *First Input Image array(Single-channel, 8-bit or floating-point)*
wt1: *Weight of the first input image elements to be applied to the final image*
img2: *Second Input Image array(Single-channel, 8-bit or floating-point)*
wt2: *Weight of the second input image elements to be applied to the final image*
gammaValue: *Measurement of light*

(a) Input Image1

(b) Input Image2

Figure 3.1 Images used as Input*

Code is:

```
# Python program to illustrate
# arithmetic operation of
# addition of two images

# organizing imports
```

*https://www.geeksforgeeks.org/arithmetic-operations-on-images-using-opencv-set-1-addition-and-subtraction/

```python
import cv2
import numpy as np

# path to input images are specified and
# images are loaded with imread command
image1 = cv2.imread('input1.jpg')
image2 = cv2.imread('input2.jpg')

# cv2.addWeighted is applied over the
# image inputs with applied parameters
weightedSum = cv2.addWeighted(image1, 0.5, image2, 0.4, 0)

# the window showing output image
# with the weighted sum
cv2.imshow('Weighted Image', weightedSum)

# De-allocate any associated memory usage
if cv2.waitKey(0) & 0xff == 27:
    cv2.destroyAllWindows()
```

*Figure 3.2 Output

*https://www.geeksforgeeks.org/arithmetic-operations-on-images-using-opencv-set-1-addition-and-subtraction/

Subtraction of Image:

With the aid of cv2.subtract(), we may combine two pictures by subtracting their pixel values, much like adding. All of the pictures need to be the same size and depth.

```
Syntax:  cv2.subtract(src1, src2)
```

```
# Python program to illustrate
# arithmetic operation of
# subtraction of pixels of two images

# organizing imports
import cv2
import numpy as np

# path to input images are specified and
# images are loaded with imread command
image1 = cv2.imread('input1.jpg')
image2 = cv2.imread('input2.jpg')

# cv2.subtract is applied over the
# image inputs with applied parameters
sub = cv2.subtract(image1, image2)

# the window showing output image
# with the subtracted image
cv2.imshow('Subtracted Image', sub)

# De-allocate any associated memory usage
if cv2.waitKey(0) & 0xff == 27:
    cv2.destroyAllWindows()
```

3.2.2. Logical operations on images

Logical operators are often used to combine two (mostly Logical operators are often used in the process of combining two pictures, the majority of which are binary. Typically, the logical operator is used in a bitwise fashion on integer pictures.

Boolean algebra, a mathematical tool for manipulating the truth values of ideas in an abstract fashion without worrying about what the concepts really imply, is the mathematical basis for most logical operators. A concept's truth value in Boolean value can only be true or false. You may model things like in Boolean algebra:

The cube is both big and red.

by something like:

A AND *B*

where A represents for "The block is red," B for "The block is big," and C for "Other." Each of these words, therefore, might be true or incorrect depending on the context in which it is spoken. In addition, the whole composite phrase has a truth value; specifically, it is true if both of the subphrases that it is composed of are true, and it is false in any other circumstance. Figure 3.3 shows a truth table that may be used to express the AND combination rule (and its complement, NAND) using the familiar method of representing true by 1 and false by 0.

A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

AND

A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0

NAND

Figure 3.3 Truth-tables for AND and NAND

The left table lists all conceivable permutations of A and B's truth values, along with the corresponding A AND B truth value. Other logical propositions may have truth-tables based on the same principles.

Operators: NAND, OR, NOR, XOR, XNOR and NOT.

Applying this logic to the realm of image processing, where each pixel in a binary picture is either 0 or 1, the pixel values may be read as truth values. By adhering to this practise, logical operations may be performed on pictures by directly applying the truth-table combination rules to the pixel values of a given pair of input images (or a single input image in the case of NOT). In most cases, the output picture is also a binary image of the same size as the input images and is generated by comparing matching pixels from the input images. Logically combining a single input picture

*https://homepages.inf.ed.ac.uk/rbf/HIPR2/logic.htm with a constant logical value is conceivable, much like with other image arithmetic operations; in this instance, each pixel in the input image is compared to the same constant to get the matching output pixel. For specific instances of these operations, please refer to the explanations of the various logical operators.

Images with integer pixel values may also be processed using logical operations. In this enhancement, logical operations are often performed in a bitwise way on binary representations of those numbers, with the output pixel value being the result of comparing corresponding bits with corresponding bits. Let's say we're working with 8-bit integers and need to know how to XOR the numbers 47 and 255 together. In binary, 47 is represented by 00101111, and 255 by 11111111. Bitwise XORing them together yields the binary value 11010000, which is equivalent to the decimal value 208.

The bitwise operation of logical operators is not universal. Some, for example, may interpret zero as false and any non-zero value as true before using standard 1-bit logical operations to create the output picture. The result may be a binary picture consisting only of ones and zeroes, or it could be a grayscale image created by multiplying the binary output image (made up of ones and zeroes) with one of the input images.

3.2.3. Thresholding

Thresholding is the quickest and easiest way to divide a picture into separate sections when working with digital photographs. Thresholding is a method for producing binary pictures from their grayscale counterparts.

In its most basic form, thresholding converts an image's pixels to either black or white depending on whether or not their intensity, measured in terms of the image's intensity vector I_{ij}, is below or above a predetermined threshold value, or threshold T. As a consequence, the bright snow on the right becomes fully white and the dark tree on the right becomes completely black in this illustration.

The threshold T may be chosen at the discretion of the user in certain circumstances, but in many others, the user will prefer that it be determined mechanically by the algorithm. In these situations, the threshold should be the "best" threshold in the sense that the partition of the pixels above and below the threshold should match as nearly as possible the real partition between the two classes of objects represented by those pixels. In other words, the "best" threshold is the threshold that most closely approximates the actual partition (e.g., pixels below the threshold should correspond to the background and those above to some objects of interest in the image).

Although Otsu's method is the most well-known and widely-used automated thresholding technique, there are many more approaches that may be used. Following is a list drawing on the research classifies thresholding techniques into broad groupings according to the kind of data the algorithm processes. However, it's important to keep in mind that any such classification would always be imprecise, as certain approaches may legitimately be placed under more than one (for instance, Otsu's approach can be thought of as both a histogram-shape and a clustering algorithm).

- **Histogram shape-based methods:** Methods that are based on the form of the histogram, such as analysing the peaks, valleys, and curvatures of the smoothed version of the histogram. These techniques rely heavily on assumptions about the probability distribution of picture intensity (i.e., the shape of the histogram),
- **Clustering-based methods:** Grayscale samples may be grouped into a foreground and a background using a clustering-based technique.
- **Entropy-based methods:** Algorithms based on entropy-based approaches take into account the cross-entropy between the original and binarized picture, the entropy of the foreground and background areas, etc.
- **Object Attribute-based methods:** Methods that are based on Object Attributes look for a degree of resemblance between the grayscale and the binary representations of the picture, such as fuzzy form similarity or edge coincidence, among other things.

- **Spatial methods:** High-order probability distributions and/or pixel correlations are used in spatial approaches.

3.3. Point-based operations on images

Grayscale range and distribution may be adjusted with ease using point operations. With the help of a predetermined transformation function, each pixel in a picture may be "point operated" onto a different one.

```
g(x, y) = T(f(x, y))
```

- g (x, y) is the output image
- T is an operator of intensity transformation
- f (x, y) is the input image

Basic Intensity Transformation Functions

Utilizing a neighbourhood size of 1 x 1 is the most basic approach of improving images. In this scenario, the pixel that is outputted, denoted by 's,' is simply dependent on the pixel that is read and denoted by ('r,') and the point operation function may be reduced as follows:

```
s = T(r)
```

If we define *T* as the point operator of a certain gray-level mapping connection between the input and output images, we get the following.

- s,r: indicate the grey level of the pixel that was entered as well as the pixel that was produced.

There are a variety of transformation functions that are applicable to the various contexts.

Linear

There are a few different kinds of linear transformations, the two most common being identity and inverse.

An identity transformation produces an identical copy of the given picture as its output.

```
s = r
```

This is the negative transformation:

```
s = L - 1 - r = 256 - 1 - r = 255 - r
```

L equals the largest grayscale value in the picture. When examining breast tissue in a digital mammography, for instance, the negative transformation is useful for bringing out white or grey information contained in dark parts of the picture.

3.1.1 Logarithmic transform

a) General Log Transform

The following is the generic log transformation equation:

```
s = c * log(1 + r)
```

Note:

- s,r: denote the gray level of the input pixel and the output pixel.
- 'c' is a constant; to map from [0,255] to [0,255], c = 255/LOG(256)
- the base of a common logarithm is 10

The log transformation inverts the sign of the intensity scale, such that lower values become greater ones. It expands the range across which a small range of dark greys is represented. In most cases, dark photos benefit most from the log modification.

b) Inverse Log Transform

Simply put, the inverse log transform is the inverse of the logarithmic transform. It expands a small range of very dark greys to a significantly brighter one. The values of pixels with lighter levels are expanded by the inverse log transformation, while the values of pixels with darker levels are compressed.

```
s = power(10, r * c)-1
```

Note:

- s,r: denote the gray level of the input pixel and the output pixel.
- 'c' is a constant; to map from [0,255] to [0,255], c =LOG(256)/255

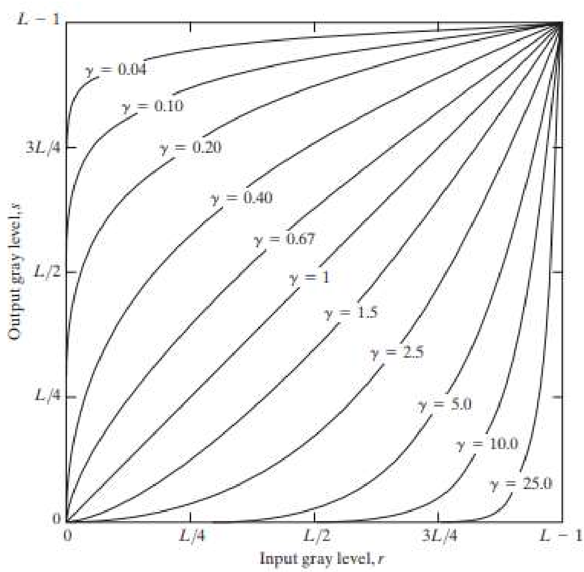
3.3.1. Power-law (gamma) transform

The gamma transform is a popular non-linear transformation for grayscale images. It is also known as the exponential or power transformation. The gamma transformation has the following mathematical expression:

s = c * power(r, γ), where

- s,r: denote the gray level of the input pixel and the output pixel;
- 'c' is a constant
- 'γ' is also a constant and is the gamma coefficient.

Plots of the equation [formula] for various values of γ (c = 1 in all cases)



All curves have been adjusted to match the shown interval. The 'r' intensity level at the input is plotted on the x-axis, while the’s’ intensity level at the output is shown on the y-axis.

Typically, the intensity value is first converted from the range of 0 to 255 to the range of 0 to 1 before the conversion is performed. First, the original range is restored, then the gamma conversion is performed.

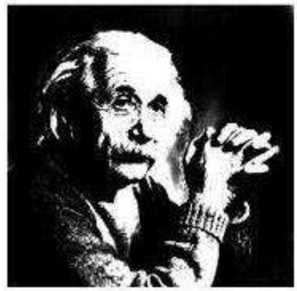
Different values of gamma may provide a wider variety of transformation curves in gamma transformation than in log transformation.

These are the photographs that were improved as a result of utilising a variety of y gamma values.

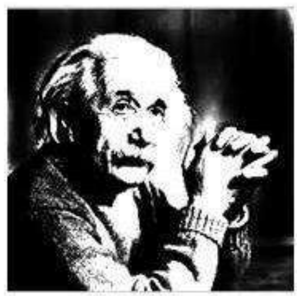
Depending on the value of γ, the gamma transformation may favourably improve the contrast of either the dark or the bright area.

- When $\gamma > 1$, the contrast of the light gray area is enhanced. Take $\gamma = 25$ for example, the pixels with the range of 0.8-1 (at the scale of 256, it corresponds to 240-255) are mapped to the range of 0-1
- When $\gamma < 1$, the contrast of the dark gray area is enhanced
- When $\gamma = 1$, this transformation is linear, that is, the original image is not changed

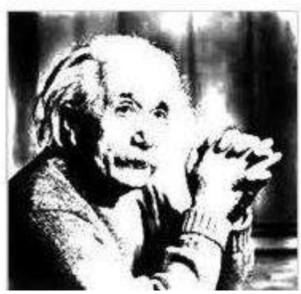
Gamma = 10



Gamma = 8



Gamma = 6



[*]*Figure 3.4 Different γ (gamma) values*

[*]https://www.dynamsoft.com/blog/insights/image-processing/image-processing-101-point-operations/#:~:text=Point%20operations%20are%20often%20used,with%20a%20predefined%20transformation%20function.

3.4. Pixel distributions: histograms

The histogram is a graphical representation of a digital picture used in digital image processing. Number of pixels representing each tone value is plotted to create a graph. Histograms for captured images are a common feature on modern digital cameras. The photographers use them to check the range of captured tones.

The horizontal axis of a graph represents the range of tones, while the vertical axis represents the number of pixels making up that range. On the horizontal axis, the left side indicates black and dark regions, the vertical axis reflects the size of the area, and the centre symbolises a colour that is somewhere in the middle between black and grey.

Applications of Histograms

1. Histograms are employed in software for straightforward computations in digital image processing.
2. It is used to picture analysis. The in-depth analysis of the histogram may be used to anticipate an image's characteristics.
3. By knowing the specifics of the image's histogram, the brightness may be changed.
4. By knowing the specifics of a histogram's x-axis, the contrast of a picture may be changed as necessary.

5. For image equalisation, it is used. A high contrast picture is created by expanding the grey level intensities along the x-axis.
6. The usage of histograms in thresholding enhances the look of the picture.
7. The kind of transformation used in the technique may be determined if we know the input and output histograms of a picture.

3.4.1. Histograms for threshold selection

In the field of image processing, an automated image thresholding technique known as the balanced histogram thresholding method (BHT) is a fairly straightforward approach. This is another histogram-based thresholding approach, similar to Otsu's Method and the Iterative Selection Thresholding Method. This strategy begins with the presumption that the picture may be broken down into two primary categories: the backdrop and the foreground. The BHT approach investigates several threshold levels in an effort to choose the one that best separates the histogram into its two distinct groups.

This technique involves weighing the histogram, determining which of the two sides is heavier, and then removing weight from the side that was previously the heavier side in order to make it the lighter side. It does the same action again and again until the sides of the weighing scale are brought together.

When introducing the concept of automated picture thresholding, this technique is an excellent option for a first effort due to the fact that it is so straightforward.

3.4.2. Adaptive thresholding

In basic thresholding, the threshold value is global, i.e., it is equal for all the pixels in the picture. Adaptive thresholding is a technique that calculates the threshold value for smaller areas. As a result, the threshold value will be different for each region since the threshold value is calculated for smaller regions.

The adaptive Threshold () function of the Imgproc class in OpenCV may be used to conduct an adaptive threshold operation on an image. The syntax of this procedure is as follows.

```
adaptiveThreshold(src, dst, maxValue,
adaptiveMethod, thresholdType, blockSize, C)
```

The following values are accepted by this procedure.−

- **src** − An object of the type Mat that represents the input image's source.
- **dst** − The final result (output) picture is represented as a Mat object.
- **Max Value** − a double-type variable that stores the value that will be provided if the pixel value exceeds the threshold value.

- **Adaptive Method** – a type-specific integer variable that represents the adaptive approach to be used. One of the following two values will represent this.
    - **ADAPTIVE_THRESH_MEAN_C** – the neighborhood's mean is used to calculate the threshold value.
    - **ADAPTIVE_THRESH_GAUSSIAN_C** – When the weights are a Gaussian window, the threshold value is the sum of the values in the surrounding area.
- **Threshold Type** – A variable of type integer that represents the kind of threshold that will be used in the process.
- **Block Size** – A variable of the integer type that represents the size of the pixel neighborhood that is used in the calculation of the threshold value.
- **C** – A double-typed variable for the constant shared by the two approaches (subtracted from the mean or weighted mean).

3.4.3. Contrast stretching

The enhancement methods are used to give a picture more contrast, which is one of their primary goals. Increasing the dynamic range of the scene's lighting is one of the most common ways to improve the quality of a photograph. Contrast stretching describes this method. Simple image enhancement method known as contrast stretching, also known as normalising, aims to increase a picture's contrast by "extending" the range of intensity values it already contains to cover a desired range of values, which is the entire range of pixel values that the image type in question is capable of displaying. This is accomplished by expanding the range of values that may be represented by each pixel in the image. When applied to a picture, contrast stretching modifies the range and distribution of the digits used to represent each pixel. This is often done to draw attention to parts of a picture that a human observer would miss.

By'stretching' the range of intensity values contained in an image to span a desired range of values, such as the full range of pixel values that the image type in question allows, contrast stretching (often referred to as normalisation) is a simple image enhancement technique that aims to improve contrast. In contrast to the more complex histogram equalisation, this method can simply use a linear scaling function on the image's pixel values. Thus, the 'improvement' is softer than before. (The majority of implementations take a grayscale picture as input and output a grayscale image.)

Contrast stretching working

The higher and lower pixel value limitations across which the picture is to be normalised must be specified prior to doing the stretching. In many cases, the lowest and maximum allowable pixel values for the picture type in question will serve as these boundaries. For grayscale photographs with 8 bits of resolution, for instance, the bottom limit may be 0 and the top limit might be 255. Use the letters *a* and *b* to denote the bottom and upper bounds.

The simplest normalising method then looks over the picture to locate the lowest and highest pixel values. Label them as *c* and *d*. Then, the following function is applied as a scale to each pixel *P*:

$$P_{out} = (P_{in} - c)\left(\frac{b-a}{d-c}\right) + a$$

A number below 0 is assigned 0, whereas a value around 255 is assigned 255.

The difficulty with this is that an outlying pixel with an extremely high or low value might have a major impact on the value of *c* or *d*, potentially leading to highly inaccurate scaling. As a result, one strategy that is more reliable is to first create a histogram of the picture, and then choose *c* and *d,* say 5th and 95th percentiles in the histogram (this means that 5% of the pixels in the histogram will have values that are lower than c, and 5% of the pixels will have values that are higher than *d*). As a result, the impact of extreme values on the overall scale is reduced.

Using the peak of the intensity histogram to determine the most frequent intensity level in a picture is another typical method of dealing with outliers. This peak is then used to set a cutoff fraction, the smallest fraction of the peak magnitude beyond which data is disregarded. Next, a scan is performed, starting at 0, on the intensity histogram, and ending at the first intensity value whose contents are greater than the cutoff fraction. *C* is so defined. The histogram of intensity is similarly scanned downward from 255 to the first intensity value that contains data over the cutoff fraction. So this is the definition of d.

Color pictures are supported by certain systems. In this situation, we must extend all channels with the same offset and scale to maintain the right colour ratios.

3.4.4. Histogram equalization

Adjusting the contrast of a picture by looking at its histogram is called histogram equalisation, and it is a technique used in image processing.

In order to improve visibility, histogram equalisation is used. It's not given that the disparity here will grow over time. Histogram equalisation may perform poorly in particular situations. This results in a lessening of contrast.

This technique, when applied to a large number of photos, will often result in an increase in the overall contrast of those images, particularly when the image is represented by a limited range of intensity values. This change allows for a more uniform distribution of intensities throughout the histogram, making full use of the available intensity range. In this way, regions with poor local contrast may improve. This is achieved by the process of histogram equalisation, which works by effectively spreading out the densely crowded intensity values that are utilised to diminish visual contrast.

The technique works well when the foreground and backdrop of a picture have the same brightness or darkness. In particular, the procedure may lead to improved views of the underlying bone structure in x-ray pictures, as well as improved detail in photos that are either over- or under-exposed. The method's main benefit is that it is a simple approach that can be easily adjusted based on the input picture and an invertible operator. This means that in principle, the original histogram may be restored if the histogram equalisation function is known. This is not a really computationally difficult calculation. The method's lack of selectivity is a drawback. It might make the noise more noticeable while reducing the signal's effectiveness.

When the spatial correlation of a signal is more significant than its intensity, as in the separation of DNA pieces of quantized length, the weak signal-to-noise ratio often hinders visual detections in scientific imaging.

While histogram equalisation might have unintended results in photography, it can be highly helpful for scientific photos like thermal, satellite, or x-ray images—the same kind of images to which one can apply fake colour. It's worth noting that applying histogram equalisation to photographs with a low colour depth might result in unintended consequences (such as noticeable visual gradient). For instance, if it were applied to an 8-bit picture that was presented using an 8-bit grayscale palette, it would further lower the colour depth of the image (the amount of distinct shades of grey). Photos having a greater colour depth than palette size, such as continuous data or 16-bit grayscale images, will benefit the most from histogram equalisation.

Histogram equalisation may be seen of as a picture change or a palette shift. In this case, the expression for the operation is *P(M(I)*, where I is the starting picture, M is the histogram equalisation mapping operation, and P is a colour selection. Histogram equalisation may be accomplished as a palette change or mapping change if a new palette is defined as *P'=P(M)*, with image *I* remaining unmodified. However, if palette P is left unaltered and the image is changed to *I'=M(I)*, then the implementation is carried out by a modification to the image itself. Changing the palette is often preferred since it does not overwrite any data.

Newer variants of the technique employ a collection of histograms (termed subhistograms) to highlight regional differences rather than global ones. Methods like adaptive histogram equalisation, contrast limiting adaptive histogram equalisation (CLAHE), multipeak histogram equalisation (MPHE), and multipurpose beta optimised bihistogram equalisation (MBOBHE) are all examples of this kind of equalisation technique. To boost contrast without introducing HE algorithm abnormalities like brightness mean-shift and feature loss is the primary focus of these techniques, with MBOBHE being a particularly promising example.

It would seem that biological neural networks do a signal transformation that is analogous to histogram equalisation. This is done in order to optimise the output firing rate of the neuron as a function of the statistics that are input. In particular, this has been shown in the retina of the fly.

Equalizing a histogram is a subset of the broader category of histogram remapping techniques. These techniques aim to increase visual quality and make images simpler to interpret (e.g., retinex)

A colour image's histogram displays the distribution of pixels across different colour channels. Separately applying histogram equalisation to the Red, Green, and Blue channels would result in drastic shifts in the overall colour balance of the picture, that’s why it is impossible to do so. If the picture is transformed to a different colour space, such as HSL/HSV colour space, then the technique may be applied to the luminance or value channel without affecting the hue or saturation of the image.

Adaptive Histogram Equalization

The Adaptive Histogram equalisation varies from traditional histogram equalisation in that it computes many histograms, each corresponding to a different area of the picture, and utilises them to disperse the image's brightness values. It is consequently appropriate for boosting the local contrast and strengthening the delineation of edges in each section of an image.

Contrastive Limited Adaptive Equalization

When compared to adaptive histogram equalisation (AHE), contrast-limited AHE (CLAHE) is distinct due to its focus on restricting contrast. In the case of CLAHE, the contrast limiting technique is executed on each neighbourhood from which a transformation function is produced. This is done in order to get the best possible results. Noise amplification by adaptive histogram equalisation is a problem, hence CLAHE was created to fix it.

3.4.5. Histogram matching

Histogram matching, also known as histogram specification, is a technique used in image processing to alter an image's histogram such that it more closely resembles a given histogram. When the given histogram is normally distributed, a particular instance of the well-known histogram equalisation technique arises.

The histogram matching method of relative detector calibration may be used to achieve this equilibrium in detector responses. When two photos have the same local lighting (such as shadows) over the same place but were captured using different sensors, atmospheric conditions, or global illumination, this method may be used to normalise the photographs.

Let's say *X* is a grayscale picture that serves as the input. It has a probability density function denoted by $p_r(r)$, where r represents a value on the grayscale and $p_r(r)$ indicates the likelihood of that value. This likelihood may be easily calculated using the image's histogram by:

$$p_r(r_j) = \frac{n_j}{n}$$

For a given number of pixels, n, the frequency of the grayscale value r_j is denoted by n_j, where n is the total number of pixels.

Consider an output-desired probability density function, $p_z(z)$. It is necessary to perform a transformation on $p_r(r)$ in order to turn it into $p_z(z)$.

It is simple to transfer each probability density function (pdf) to its cumulative distribution function by

$$S(r_k) = \sum_{j=0}^{k} p_r(r_j), \qquad k = 0, 1, 2, 3, \ldots, L-1$$

$$G(z_k) = \sum_{j-0}^{k} p_z(z_j), \qquad k = 0, 1, 2, 3, \ldots, L-1$$

The total number of grayscale levels is denoted by L. (256 for a standard image).

The goal is to find the z-value in the target probability distribution function (pdf) that corresponds to each r-value in X. I.e. $S(r_j) = G(z_i)$ or $z = G^{-1}(S(r))$.

CHAPTER

4

Enhancement

4.1. Why perform enhancement?

Image enhancement is the process of increasing the overall quality of the image as well as the amount of information included in the raw data before it is processed. Methods like FCC, spatial filtering, density slices, and contrast enhancement are often used. When increasing contrast or extending an image, a linear transformation is used to increase the grayscale. Spatial filtering increases the naturally existing linear characteristics including fault, shear zones, and lineaments. Density slicing is a method of visually representing characteristics by dividing the continuous gray-tone range into discrete density intervals, each of which is represented by a unique colour or symbol.

Given that additional scattering occurs mostly in the blue wavelength, false colour composites (FCCs) are often utilised in remote sensing in place of actual colours. Because it provides consumers with the most consistent data possible about Earth's objects, the FCC has been accepted as a standard. In normal FCC, vegetation appears red because vegetation is particularly reflective in NIR and the colour applied is red. Since infrared (IR) is absorbed by water, bodies of water seem black if they are transparent or very deep. Water bodies reflect light in the green wavelength, which causes the colour blue to be produced regardless of the turbidity or shallowness of the water. This results in the appearance of different hues of blue.

In order to increase the quality of a picture, it is standard practise to apply image enhancement algorithms to remotely sensed data, resulting in a new improved image. The improved picture is often simpler to comprehend than the original image.

Multiple bands of the electromagnetic spectrum are concurrently scanned to create RS picture of the same scene. Bandwidth refers to the range of wavelengths across which a certain spectral measurement was made, and represents the average radiance observed in that band. The range of grey levels (GL) in a picture has a direct correlation to the contrast of the image; generally speaking, the bigger the range, the greater the contrast, and vice versa. When enhancing contrast, both linear and non-linear methods are applied.

Given that almost all digital photos are altered in some manner, it is instructive to look back at the standards that the authors set for the standard aesthetic improvement and presentation of aerial photographs:

- Turning the camera such that the horizon is level in very skewed views and the direction of shadows is down and to the right in upright shots.
- Limited cropping of the picture after rotation or to accentuate aspects of interest.
- Saturation of the dark and light ends of the image's brightness is the only kind of contrast modification possible.
- Limited modification of specific colour bands or colour balance.
- We can see more detail in the items in the image with a limited amount of sharpening, like an unsharp mask.
- Marking of certain features or locations in an image.
- Mosaicking or sewing together many photographs of the same scene shot at or at the same time, as specified in the image's description or explanation.

4.2. Enhancement via image filtering

Image enhancement refers to the technique of applying processing to an image in order to improve specific aspects of the image. Image enhancement is fundamentally enhancing the interpretability or perception of information in pictures for human viewers and giving better input for other automated image processing processes. The primary purpose of image enhancement is to change aspects of a picture in order to make it more appropriate for a certain observer to use in conjunction with a particular endeavour. It's a procedure in which some aspect(s) of a picture is altered. The choice of qualities and the method they are updated are particular to a certain activity. A lot of subjectivity will be introduced into the decision-making process about the techniques of picture augmentation due to the presence of observer-specific characteristics such as the human visual system and the observer's level of expertise. To improve an image when removing the image's noise enhancement of the dark picture and highlight the edges of the items in an image. For certain specialised purposes, the final product is more suited than the original picture. Methods of processing are heavily problem-focused. For instance, the most effective ways for enhancing X-ray pictures may not be the most effective techniques for enhancing microscopic images.

Image processing contains both theory and procedures that may fill many volumes. The key idea used by the majority of the described approaches is that each pixel in the final picture is derived from the immediate vicinity of its corresponding pixel in the input image. Only a handful of the enhancement techniques, however, are global in the sense that they make use of each and every pixel of the input picture while generating the final product. The two ideas that are discussed which are considered to be the most significant are the correlation, which is the process of matching an image neighbourhood with a pattern or mask, and convolution, which is a single approach that may apply many different effective filtering processes.

There is a wealth of literature on the subject of filtering digital waveforms in one dimension or pictures in two dimensions. Digital image filtering is based on the principle of post-processing images using common methods borrowed from signal processing theory. One may compare these effects to those achieved by using different filters in conventional photography. When attached to the lens of a camera, optical filters amplify or reduce the intensity of certain qualities of the picture captured on film. For example, photographers may employ a red filter to differentiate plants from a backdrop of mist or haze, and most professional photographers use a polarising filter for glare removal. In contrast to optical filters, which do their magic in the analogue domain (and are so also called analogue filters), the filters we use to process digital pictures are all digital.

The term "sliding neighbourhood processing" refers to a typical technique for filtering pictures. In this method, a "mask" is slid over the input image, and at each point, an output pixel is generated using some formula that combines the pixels inside the current neighbourhood.

Convolution filters are used for blurring and sharpening of changing picture when paired with the Histogram equalisation for improving medical image by modulating the contrast of the image. After that, the image will be fine-tuned to look its best. So, employ morphological for segmentation. Using the triple filter to fine-tune the threshold of morphological and normalising value.

4.3. Pixel neighbourhoods

A pixel's neighbourhood is the set of neighbouring pixels. The neighbourhood of a pixel is necessary for operations such as morphology, edge detection, median filter, etc.

Many computer vision techniques enable the programmer to pick an arbitrary neighbourhood. In most cases, these algorithms produce a new picture by deriving the value of each new pixel as a function not only of the value of the pixel that it corresponds to in the old image, but also of the values of the old pixels that are next to it. The neighbourhood surrounding a pixel is also commonly dubbed a "window" or "peephole" around that pixel. One kind of neighbourhood is formed by the non-zero elements in a "convolution kernel." Another kind of neighbourhood is a morphological structural feature.

A pixel with the coordinates p that is located at (x, y) has four neighbours that are horizontal and vertical, and their coordinates are as follows:

$$(x + 1, y), (x - 1, y), (x, y + 1), (x, y - 1)$$

This group of pixels, which is referred to as p's 4-neighbors, is represented by the symbol N_4 (p). Each pixel is a unit distance from (x, y), and some of the neighbours of p reside outside the digital picture if (x, y) is on the boundary of the image.

Coordinates for p's four diagonal neighbours are given.

$(x+1, y+1), (x+1, y-1), (x-1, y+1), (x-1, y-1)$

And are denoted by $N_D(p)$. These points, together with the 4-neighbors, are called the 8-neighbors of p, denoted by $N_8(p)$.As before, some of the points in $N_D(p)$ and $N_8(p)$ fall outside the image if (x, y) is on the border of the image.

4.4. Filter kernels and the mechanics of linear filtering

Nonlinear spatial filtering

A nonlinear (or non-linear) filter in signal processing is one whose output does not scale linearly with its input. That is, if the filter generates output signals R and S in response to two independent input signals r and s, but does not always generate output of αR + βS when the input is a linear combination of αr and βs.

Nonlinear filters may be implemented in both the continuous- and discrete-domain settings. The former category includes, for example, any electrical device whose current output voltage *R(t)* at each instant is equal to the square of the input voltage *r(t)*; or which is the input clipped to a set range *[a,b]*, precisely *R(t) = max(a, min(b, r(t))*. An essential example of the latter kind is the running-median filter, which is designed in such a way that each output sample *Ri* is equal to the median of the most recent three input samples *ri, ri-1, ri-2*. Nonlinear filters may be shift invariant, much as linear filters.

In particular, non-linear filters are useful for suppressing non-additive forms of noise. Spike noise, which affects a negligible fraction of samples but may add up to a significant amount overall, is often filtered out using the median filter. As a matter of fact, non-linear filters (analog-to-digital converters) are essential to all digital signal processing since they convert analogue signals to binary numbers and are used in all radio receivers to down convert kilohertz to gigahertz transmissions to the audible range.

Nonlinear filters are more challenging to use and construct than linear ones because the most powerful mathematical methods of signal analysis (such as the impulse response and the frequency response) cannot be used for them. Since the ideal non-linear filter would be very difficult to build and implement, linear filters are often used to clean up signals that have been distorted or otherwise degraded as a result of nonlinear processes.

As we have seen, nonlinear filters behave in a very different way from linear filters. The most distinctive feature of nonlinear filters is that their responses do not conform to the previously described criteria, especially those pertaining to scaling and shift invariance. The outcomes of using a nonlinear filter might also differ in unexpected ways.

Applications

Noise removal

During transmission or processing, signals often get damaged, and one of the most common goals in filter design is the restoration of the original signal, which is a process that is generally referred to as "noise removal." Additive noise, the simplest kind of corruption, occurs when the intended signal *S* is combined with an undesirable signal *N* that has no known link to *S*. To the extent permitted by Shannon's theorem, a Kalman filter will decrease *N* and restore *S* if the noise *N* has a simple statistical description, such as Gaussian noise. More specifically, linear bandpass filters may effectively partition *S* and *N* if and only if their frequencies do not overlap.

However, a non-linear filter will be required for optimal signal recovery when dealing with practically any other kind of noise. It may be sufficient, for instance, to transform the input to a logarithmic scale, apply a linear filter, and then transform the resulting signal back to a linear scale if the noise is multiplicative rather than additive. In this particular illustration, the first and third stages do not follow a linear progression.

When some "nonlinear" aspects of the signal are more essential than the total information contents, non-linear filters may also be helpful. To maintain the integrity of a scanned drawing's linework or the crispness of a photograph's silhouette is a common goal in digital image processing. Those details will likely be muddled by a linear noise-removal filter; a non-linear filter could provide better results (even if the blurry image may be more "correct" in the information-theoretic sense).

The time domain is used by several nonlinear noise-removal filters. Most of the time, they look at the input digital signal in a small window around each sample and use a statistical inference model (either implicitly or explicitly) to estimate the most probable value for the original signal at that instant. Filtering issue for a stochastic process refers to the estimating and control theory challenge of designing such filters.

The following are some examples of nonlinear filters:

- phase-locked loops
- detectors
- mixers
- median filters
- ranklets

Among several crucial image processing operations, nonlinear filters play a prominent role. It is usual practise to incorporate a number of nonlinear filters in the pipeline that is used for real-time image processing. These filters are used to create, shape, detect, and change picture information. Furthermore, utilising adaptive filter rule generation, each of these filter types may be customised to function in one manner under certain conditions and in another way under another set of circumstances. The objectives might range from simple feature abstraction to more complex noise cancellation. Most image processing systems use some kind of filtering to refine input picture data. The most common kind of filter construction is the nonlinear filter. For instance, if a picture has a modest level of noise but a very large magnitude, then a median filter is likely to be the most suitable choice.

Order Statistics Filter: Filters work by arranging the pixels in the region of the picture they cover in a certain order. The ranking result is used to replace the value of the central pixel. The edges are better retained in this filtering.

Types of Order statistics filter:

(i) Minimum filter: The minimal filter is the one at the 0th percentile. The minimum value inside the window is substituted for the centre value.

(ii) Maximum filter: The maximal filter is the one with a 100th percentile. The biggest value inside the window replaces the centre value.

(iii) Median filter: Consideration is given to each and every pixel included in the photograph. First, the pixels in the immediate vicinity are sorted, and then the median value from that set is used to replace the pixel's original value.

(iv) Sharpening Spatial Filter: Derivative filter is another name for it. When compared to its smoothing counterpart, the sharpening spatial filter is intended to increase contrast. The feature's primary function is to eliminate blurring and emphasize edges. The first and second derivatives form the basis of this method.

First order derivative:

- A flat segment must have a value of zero.
- In order to begin a grey level step, you must have a non-zero value.
- It can't be 0 along the ramps.

First order derivative in 1-D is provided by:

```
f' = f(x+1) - f(x)
```

Second order derivative:

- Must be 0 in areas that are flat.
- Both the beginning and ending points of a ramp must be set to zero.
- Along ramps, it must be zero.

Second order derivative in 1-D is given by:

```
f'' = f(x+1) + f(x-1) - 2f(x)
```

4.5. Filtering for noise removal

The study of anatomical structure and the image processing of MRI medical pictures have both benefited greatly from the use of noise reduction methods, which have evolved into an integral part of the medical imaging application. Multiple de-noising algorithms, including the Weiner filter, Gaussian filter, median filter, etc., have been created to report these problems. Only three of the filters indicated above have been utilised effectively in medical imaging. Salt and pepper, speckle, Gaussian, and Poisson noise are the most prevalent types of noise seen in medical MRI images. In order to make a comparison, medical imaging such as MRI scans in both grayscale and RGB are used. The effectiveness of these methods is evaluated using a number of different noise characterizations, including salt-and-pepper, Poisson, speckle, blurred, and Gaussian Noise. The assessment of these techniques is performed based on the measurements of the picture file size, the histogram, and the clarity scale of the photographs. The experimental findings reveal that the median filter is superior for eliminating salt-and-pepper noise and Poisson Noise from grayscale pictures, while the Weiner filter is superior for removing Speckle and Gaussian Noise and the Gaussian filter for the Blurred Noise.

Gaussian noise, Poisson noise, Blurred noise, Speckle noise, and salt-and-pepper noise are only few of the types of noise that may be generated by a number of different external elements and components of a transmission system. In medical imaging applications, the process of eliminating noise has become a significant aspect, and the filters Median filter, Gaussian filter, and Weiner filter are the most widely used filters. These filters produce the best outcome for the respective noises they are designed to remove.

The smoothing of pictures, which is necessary in order to get rid of the noise, has become a vital need, and in order to do this, the best filters or the standard filters are used in the majority of image processing programmes. A successful picture de-noising model will be able to eliminate noise while keeping the image's edges unaltered. In general, linear models are employed because of their speed; however, their disadvantage is that they are unable to retain the edges in an effective way. There are two different models that may be used for the process of de-noising, which are known as linear models and non-liner models.

To make sense of this information, filters are used, and the optimal filter is determined by analysing the MRI pictures' histogram, size, and clarity.

De-noising a picture is a crucial part of the image processing workflow, both on its own and as an integral part of other workflows. The process of removing noise from a picture may be accomplished in a number of ways. Various algorithms are used to solve it. As a result, noises are identified with the help of nearby information and are eliminated using the best filtering methods without compromising the picture quality, therefore enhancing the smoothness of the image that was collected for analysis.

4.5.1. Mean filtering

Image smoothing, or minimising the amount of intensity change from pixel to pixel, may be accomplished with relative ease by using a technique called mean filtering. The process is often used in the art of picture noise reduction.

How It Works

Simply said, the principle behind mean filtering is to swap out the original values of each pixel in a picture with the average (or "mean") value of its surrounding pixels. As a result, out-of-context pixel values are wiped out. To most people, a mean filter is a convolution filter. Just like previous convolutions, this one relies on a kernel to determine what kind of area should be sampled for the mean. The common usage of a 3x3 kernel, although bigger kernels (e.g., 5x5 squares) may be used for more extreme smoothing. (It's important to keep in mind that a tiny kernel may be applied several times to get a result that is close but not identical to that of a big kernel.)

The average filtering operation is performed by computing the basic convolution of an image with this kernel.

The four different varieties of mean filters. They are:

(1) Arithmetic mean filter

The mean-centered filter is the simplest kind of filter. Let us denote by S_{xy} the collection of coordinates inside an m-by-n-pixel rectangular sub-image window that is centred at the given location (x, y). The average value of the corrupted picture g(x, y) inside the region indicated by S_{xy} is calculated via the arithmetic mean filtering procedure. At every coordinate (x, y), the value of the restored image f is equal to the geometric mean of the pixels in the area delimited by S_{xy}.

$$\hat{f}(x, y) = \frac{1}{mn} \sum_{(s,t)\in S_{xy}} g(s, t).$$

All the coefficients in the convolution mask need to be set to 1/mn for this operation to be realised.

(2) Geometric mean filter

The expression represents an image that has been recovered by using a geometric mean filter.

$$\hat{f}(x, y) = \left[\prod_{(s,t)\in S_{xy}} g(s, t) \right]^{\frac{1}{mn}}.$$

In this case, the value of each pixel that has been restored is determined by the product of the pixels that are included inside the sub-image window, which is then increased to the power of 1/mn. While both the arithmetic and geometric mean filters smooth the image, the geometric mean filter often results in less information loss.

(3) Harmonic mean filter

It can be shown that the formula describes the harmonic mean filtering process.

$$\hat{f}(x, y) = \frac{mn}{\sum_{(s,t)\in S_{xy}} \frac{1}{g(s, t)}}.$$

However, the harmonic mean filter does not fare as well with pepper noise as it does with salt noise. It works well with various forms of noise, such as Gaussian noise.

(4) Contra harmonic mean filter

An image restoration based on the equation is produced by the contra harmonic mean filtering process.

$$\hat{f}(x,y) = \frac{\sum_{(s,t)\in S_{xy}} g(s,t)^{Q+1}}{\sum_{(s,t)\in S_{xy}} g(s,t)^{Q}}$$

where Q represents what is known as the filter's "order." This filter works very well at minimising or completely removing the impact of salt-and-pepper sounds. In the presence of positive values of Q, the filter gets rid of pepper noise. It gets rid of salt noise for negative values of Q. It can't do both at the same time. Take note that when Q = 0, the contra harmonic filter reduces to the arithmetic mean filter, and when Q = -1, it reduces to the harmonic mean filter.

4.1.1 **Median filtering**

When it comes to digital image processing, the Median filter is the most well-known order-statistic filter. Because of its effective de-noising power and accurate mathematical representation, the median filter is a method that is widely used for the elimination of impulsive noise. When applied, the Median Filter takes the average intensity of the pixels around a given pixel and uses it as the new pixel's value. The Median Filter takes the results of neighbouring pixels and averages them using a filtering window of a predetermined size. Since median filters are applied uniformly throughout a whole image, they have a tendency to affect both noisy and noise-free pixels. This means that tainted pixels might theoretically replace otherwise valid ones at any time. Because of this, de-noising often results in blurred and distorted features, resulting in the loss of any fine details in the original image.

Median filtering is a common image of digital noise suppression. It is a non-linear filtering technique. Noise reduction like this is a common pre-processing procedure that yields better results in subsequent steps (for example, edge detection on an image). The median filter is a technique that is used extensively in digital image processing due to the fact that, under some circumstances, it may maintain edges while simultaneously reducing noise. Additionally, this technique has uses in signal processing.

When applying a median filter to a signal, it is common practise to iterate over the signal one element at a time, replacing each element with the middle element of its nearby elements. The "window" is the pattern of neighbours that is slid across the whole signal one entry at a time. The window must include all entries within a certain radius or elliptical area for two-dimensional (or higher-dimensional) data, in contrast to one-dimensional signals, where the window is often simply the first few preceding and following entries (i.e. the median filter is not a separable filter).

4.5.2. Rank filtering

When applied to photos, rank filters place the pixel with the grey level that is K^{th} highest inside a window of M pixels ordered by value. The exceptional situations $k = 1$, $k = M$ *(MIN and MAX filter), and* $k = (M + 1)/2$ *(medium filter)*, which have previously been employed in image processing, are explored in a systematic manner in relation with all rank filters. This is done so in order to better understand how these filters function. It is possible to express some of these qualities analytically. They share a common language with grayscale monotonic transformations. For one-dimensional functions—including line-like picture structures—the output functions of monotonic input functions may be determined exactly. It has been shown that using the *MIN* and *MAX* filters in a cyclical fashion yields the same result as applying them once, even if the cycles are much longer. After applying the rank filters to a collection of test pictures, it becomes clear that their effect on the spectrum cannot be simply described using a transfer or autocorrelation function. It is not possible to characterise the median filter's smoothing in terms of a low-pass filter, but via the mean local variance reduction that occurs while using the filter. Using both synthetic and real-world data, we show that rank filters maintain edges while smoothing the picture less than linear filters.

Rank filters are a special kind of non-linear filter that calculate the filtered value based on the local gray-level rank of the input data. The local gray-level histogram in the vicinity of a pixel is the foundation on which this collection of filters is built (defined by a 2D structuring element). The classical median filter is obtained by selecting the value in the centre of the histogram as the filtered value.

There are a variety of applications for rank filters, including the following:

- Image processing techniques that improve image quality include: smoothing, sharpening, etc.
- Preparation of an image for display, including filtering out unwanted details and boosting the ones that are there.
- Extraction of features, such as borders or single points
- Image editing techniques such as blurring, sharpening, or removing unwanted elements

4.5.3. Gaussian filtering

Speckle noise is a common kind of noise seen in digital photographs and MRI scans, and it may have both internal and external causes. Speckle Noise in MRI brain scans and ultra sound scans may be removed using a Gaussian filter. In this method, the value of the surrounding or nearby pixels is averaged and used to replace the noisy pixel that is already present in the image. This method is based on the Gaussian distribution.

Gaussian filters are low pass filters used to blur parts of an image and reduce noise (high frequency components). To get the desired effect, the filter is applied to each pixel in the Region of Interest by first passing through an Odd sized Symmetric Kernel (the DIP version of a Matrix). Due to the fact that the pixels closer to the centre of the kernel have greater weightage towards the final value than the pixels closer to the periphery, the kernel is not as sensitive to sudden colour changes (edges). One way to think about a Gaussian Filter is as a function approximation of the Gaussian distribution.

4.6. Filtering for edge detection

We may also state that abrupt shifts or discontinuities in an image are what we refer to when we mention the word "edge." Edges are defined as sharp discontinuities in an image.

When an image is broken down into its component parts, the edges are where the majority of the shape information is contained. Therefore, the first step is to identify the edges present in an image. Next, the appropriate filters are applied, and finally, the parts of the image that include edges are enhanced. This process ultimately results in an image that is sharper and more distinct.

After proper edge detection is implemented, edges are commonly employed for measurements since they are one of the most essential elements of a structure.

Dragonfly's edge detection filters may be used to highlight the changes and sharpness of an image's edges.

4.6.1. Derivative filters for discontinuities

Quantifying the rate of change in pixel brightness information provided in a digital image is made possible by derivative filters. The information about the rates of change in brightness obtained by applying a derivative filter to a digital image may be used to improve contrast, identify borders and boundaries, and quantify feature orientation.

When you first launch the lesson, an image of a specimen (taken with a microscope) will display in the window to the left, labelled Specimen Image. After the name of each specimen is an acronym for the contrast mechanism that was used to create the image. The following abbreviations are often employed: (FL) for fluorescence; (BF) for brightfield; (DF) for darkfield; and (POL) for polarised light. The behaviour of collected specimens in the image processing lesson will vary depending on whatever optical microscope method was used to capture them.

The output image is shown in the Output Image window, which is situated to the right of the Specimen Image window. This window shows the specimen image after a derivative filter has been applied to it. To follow the instructions choose an image to work with, choose a Specimen section, and then pick a derivative filter in the Sobel Operation section. Visitors are encouraged to investigate how the different Sobel processes change the final image.

Based on the selection of a 3 x 3 kernel mask, the Sobel derivative filter's convolution process may generate a derivative in any of eight directions. Microscope digital photos benefit greatly from these convolutions when it comes to sharpening the image's edges. After the application of the necessary improvement algorithms, edges in a microscopic structure may often be exploited for measuring purposes. Edges are frequently one of the most essential characteristics found in a microscopic structure.

Convolution of the specimen image with the first of the two kernel masks stated above corresponds to an operation equivalent to a horizontal derivative filtering operation, while convolution with the second of these kernel masks amounts to an action equivalent to a vertical derivative filtering operation. If the brightness value of a pixel at the coordinates (x, y) in an image is given by B(x, y), then the finite partial derivatives of B in the horizontal and vertical directions show the relative change in brightness in each direction and may be notated as:

$$\frac{\delta B}{\delta x}$$

and

$$\frac{\delta B}{\delta y}$$

Grayscale pictures are the images that are formed as a derivative via the process of convolution using Sobel and similar kernel masks. These images store high-frequency spatial information in the direction of interest as abrupt changes in brightness between light and dark. The tutorial's Sobel filters are shown through the Horizontal Edges and Vertical Edges selections in the Sobel Operation drop-down menu. In order to replicate differential interference contrast (DIC) pictures, microscopists often use derivative filters at 45 degrees. The following is a list of some examples of these filters, and you can access them all using the pull-down menu labelled "Sobel Operation."

In order to acquire a measure of their magnitude that is not reliant on the orientation in which it is seen, it is possible to combine the Sobel derivatives in two orthogonal directions by taking the square root of the sum of their squares. The Sobel operator was developed to quantify this concept. The Sobel operator is one of the most often used techniques for enhancing boundaries because of the quality of the results it produces.

With the Sobel operator, you can also determine the directional component of a gradient or edge for every pixel in the gradient or edge. In order to achieve this goal, one must first calculate the arc tangent of the ratio of the brightness value partial derivatives, as shown in the following equation:

$$\mathbf{Angle} = \mathbf{ArcTan}\left(\frac{\delta B/\delta y}{\delta B/\delta x}\right)$$

Every pixel in the image may be assigned a direction when angle measure is scaled to the display's grayscale range. The visual impression of this image might be overpowering, but not particularly instructive, since each pixel is presented according to its orientation alone. When the direction information is scaled according to the associated magnitude information in order to generate an image that displays both the edges and their orientation, a more suitable representation may be achieved. This will result in an image that shows both the edges and their orientation. This is same as selecting Edge Direction (Intensity) from the Sobel Operator drop-down menu, which is used in the lesson. Additionally, a color-coded representation of the magnitude and direction information may be obtained by using the HSI colour space. The HSI hue component may be used to convey information about the direction of rotation, whereas the HSI intensity component can be used to convey information about the size of an object. The Edge Direction (Hue) selection in the tutorial's Sobel Operator pull-down menu maps to this depiction. By thresholding across a range of hues (not covered in the tutorial), the hue image may be converted to binary, and from there, a pixel count for each colour (direction) can be acquired for edge analysis.

4.6.2. First-order edge detection

The majority of edge detection algorithms are based on the idea that an edge may be found everywhere there is either a break in the intensity function or a very sharp gradient in the intensity of the image. With this presumption in mind, it should be possible to detect the edge of the image by taking the derivative of the intensity value throughout the image and looking for locations where the derivative is at its highest. Pixel values change quickly with distance in the x and y axes, and the gradient is a vector whose components reflect this rate of change. To find the gradient's individual components, we may use the following formulas.

$$\frac{\partial f(x,y)}{\partial x} = \Delta x = \frac{f(x+dx,y) - f(x,y)}{dx}$$

$$\frac{\partial f(x,y)}{\partial y} = \Delta y = \frac{f(x,y+dy) - f(x,y)}{dy}$$

Where dx and dy represent the horizontal and vertical distances travelled. Distances dx and dy may be thought of in terms of pixels in a discrete picture. When pixel spacing (dx, dy) is 1, the x, y coordinates of a pixel are: (i, j) Therefore, the value of *(Δx and Δy)* may be computed by using equations.

$$\Delta x = f(i+1,j) - f(i,j)$$

$$\Delta y = f(i,j+1) - f(i,j)$$

Calculating the change in the gradient at the coordinates (i j) is one way to determine whether or not a gradient discontinuity is present. This may be accomplished by determining the magnitude measure that follows, and the gradient direction θ is indicated by the equation.

$$\emptyset = \tan^{-1}\left[\frac{\Delta y}{\Delta}\right]$$

An example of the gradient approach is the Sobel operator. A discrete differentiation operator, it approximates the gradient of the image intensity function.

$$\Delta x = \begin{bmatrix} -1 & 1 \\ 0 & 0 \end{bmatrix}, \quad \Delta y = \begin{bmatrix} -1 & 0 \\ 1 & 0 \end{bmatrix}$$

Using a bigger mask size has the added benefit of allowing for more local averaging inside the neighbourhood of the mask, which in turn reduces mistakes caused by noise. The fact that the operators are centred and, as a result, are able to offer an estimate that is based on a centre pixel is one of the benefits of utilising a mask of an odd size (i,j). The Sobel edge operator is a prime example of this category of edge operators. Specifically, the masks for the Sobel edge operators are as follows:

$$\Delta x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad \Delta y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

A gradient of an image's intensity is computed for each pixel, indicating the direction of the greatest potential rise from light to dark and the rate of change in that direction. The result demonstrates how "abruptly" or "smoothly" the image changes at that point, and hence how probable it is that a portion of the image represents an edge, in addition to indicating how the edge is likely to be orientated.

Practically speaking, the magnitude estimate (the chance of an edge) is more trustworthy and straightforward to comprehend than the direction calculation. At each pixel in an image, the gradient of a two-variable function (the image intensity function) is a two-dimensional vector whose components are defined by the horizontal and vertical derivatives of the function. The gradient vector points in the direction of the maximum potential rise in intensity at each image point, and the length of the gradient vector corresponds to the rate of change in that particular direction. This means that the Sobel operator yields a zero-vector at every image point inside an area of constant image intensity, and a vector that points across the edge, from darker to brighter values, for any image point that sits on the edge.

4.6.3. Linearly separable filtering

In image processing, a detachable filter is represented by the product of two simpler filters. The standard practise is to split a convolution in two dimensions into two 1-dimensional filters.

The ability to apply a separable filter to an image may make a technique that was previously considered "theoretical and too costly" feasible within the same computing restrictions. For alternative "interactive" (or offline) methods, the ability to utilise a separate filter might be the deciding factor in making them really real-time.

Let's imagine we need to apply certain filters to an image in order to bring out some details, hide others, or identify edges and other characteristics. Computing a 2D image filter of size MxN would need MxN separate, sequential memory accesses (often referred to as "taps") and MxN multiply-add operations. This may quickly become unfeasible when dealing with massive filters, since the cost grows quadratically with the filter's spatial breadth. Separate filters may save the day in this situation.

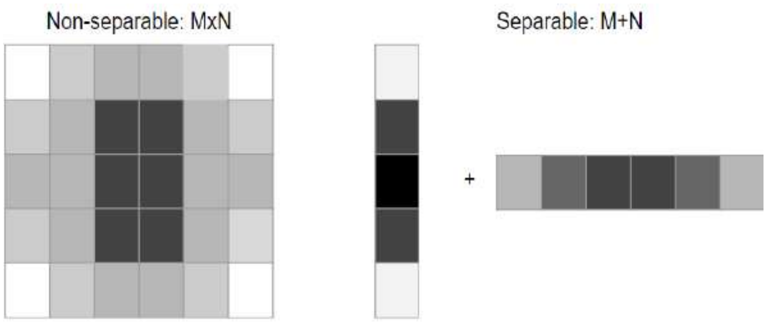


Figure 4.1 Separable and Non-Separable filters[*]

When a filter is separable, it may be broken down into a pair of orthogonal 1D filters (usually horizontal, and then vertical). The first pass employs M taps, whereas the second pass employs N taps, for a grand total of M+N filtering operations. This necessitates the storage of the intermediate findings, either in the computer's memory or locally (line buffering, tiled local memory optimizations). Scaling is

---

[*] https://bartwronski.com/2020/02/03/separate-your-filters-svd-and-low-rank-approximation-of-image-filters/

linear rather than quadratic, but this comes at the expense of having to store intermediate findings and coordinate the passes. `Therefore, employing separable filters is going to be much quicker than the naïve, non-separable technique for any filter size more than ~4 x 4 (depending on the hardware, implementation, etc.).

4.6.4. Second-order edge detection

When evaluating an image's second derivative, the Laplacian is a 2-dimensional metric. The Laplacian of an image is often employed for edge identification (0 crossing edge detectors) because it draws attention to areas of rapid intensity change. The Laplacian is often used to further lower an image's susceptibility to noise after it has been smoothed using a filter that roughly resembles a Gaussian smoothing filter. The standard input for the operator is a grayscale image, and the expected output is a binary image. The zero crossing detector examines an image in search of locations in the Laplacian at which the value of the Laplacian crosses through zero, or points at which the Laplacian takes on a different sign. These spots tend to appear along the edges of pictures, which are defined as sites where there is a sudden shift in the intensity of the image. However, they may also appear at locations that are more difficult to link with edges. The zero crossing detector is not an edge detector but a feature detector. Due to the fact that zero crossings are always found on closed contours, zero crossing detectors often provide a binary image with single-pixel-thick lines indicating the locations of zero crossings. As demonstrated in the equations, the Laplacian operator is the derivative of an image.

$$\Delta^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

For X-direction, $\frac{\partial^2 f}{\partial x^2} = f(x+1, y) + f(x-1, y) - 2f(x, y)$

For Y-direction, $\frac{\partial^2 f}{\partial y^2} = f(x, y+1) + f(x, y-1) - 2f(x, y)$

By making these three substitutions in the equations, we get the new equation, which is as follows:

$$\Delta^2 f(x,y) = f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1) - 4f(x, y)$$

If we utilise the value of Mask as an explanation for the blow, then we get the following equation:

0	1	0
1	-4	1
0	1	0

4.7. Edge enhancement

Edge enhancement is a kind of image processing filter that boosts the contrast around the edges of an image or video in an effort to make the picture or video clearer (apparent sharpness).

The filter works by locating sharp edge boundaries in the image, such as the boundary between a subject and a backdrop of a colour that contrasts with it, and boosting the image contrast in the region that is immediately around the edge. For example, the filter may identify the edge between a subject and a background of a colour that contrasts with it. This causes overshoot and undershoot, which are essentially faint bright and dark highlights on each side of any edges in the image, respectively, and makes the edge seem more defined from a normal viewing distance.

The method is widely used in the video industry and can be seen in almost all TV shows and home video releases.

Sharpness controls on newer TVs are an application of edge enhancement. Also, it's often used in computer printers to improve the quality of printed text and images. Edge enhancement is another common feature of digital cameras, although it is often not customizable.

The procedure of enhancing edges may be carried out in either analogue or digital form. Example applications of analogue edge enhancement include current cathode ray tube (CRT) TVs and other forms of all-analog visual equipment.

Several factors determine the final result of an image's edge enhancement; the most used approach is unsharp masking, which has the following settings:

- *Amount*. This determines how much of an enhancement is made to the contrast in the region where edges are identified.

- *Radius* or *aperture*. That determines how much of the region around the edge will be modified by the improvement and how large the edges themselves will be. When the radius is decreased, the enhanced region surrounding the edge becomes smaller and only the sharpest, finest edges are affected.
- *Threshold*. This modifies the edge detection mechanism's sensitivity when it is possible. When the threshold is decreased, more subtle colour transitions are recognised as edges. A threshold that is too low may lead to certain tiny bits of surface textures, film grain, or noise being wrongly detected as being an edge. This can happen when the threshold is set too low.

In some circumstances, edge enhancement may be done in either the horizontal or the vertical direction alone, or it can be applied in both directions in varying degrees. Edge enhancement, which may be applied to pictures originating from analogue video, may benefit from this.

Effects of edge enhancement

Whereas other types of image sharpening may improve the image of fine detail in otherwise uniform regions of an image, such as texture or grain, edge enhancement can only improve the appearance of gradients and sharp edges. The advantage of this is that flaws in the image reproduction, such as grain or noise, as well as flaws in the subject, such as naturally occurring defects on a person's skin, are not made more evident by the procedure. It's possible that the picture's natural image may suffer as a result, as the general degree of sharpness has improved but the amount of detail in smooth, flat parts has not.

Edge enhancement, like other methods of image sharpening, can only improve an image's apparent sharpness or acutance; it cannot improve the actual sharpness of an image. Some image information is lost due to filtering since the augmentation is not fully reversible. In addition to the loss of information introduced by the first sharpening operation, further sharpening procedures on the resultant image introduce artefacts like ringing. An example of this may be observed when an image, such as the picture on a DVD video, which has already had edge enhancement done to it, has more edge enhancement added by the DVD player it is played on, and perhaps also by the television it is shown on. The first edge enhancement filter, in its most basic form, generates brand new edges on each side of the current edges, which are then improved in future steps.

4.7.1. Laplacian edge sharpening

An image sharpening effect is used on digital images to make them seem crisper. Sharpening improves an image's edge definition. Images with low edge quality are the most boring ones. Background and edges are also similar. On the other hand, a sharpened picture is one in which the edges are distinct. At the periphery, brightness and contrast are known to shift. If the difference is noticeable, we say that the picture is in focus. All elements in the foreground and background are easily visible.

Image sharpening using the smoothing technique

Laplacian Filter

- It acts as a filter or a mask for derivatives and is of the second order.
- Images in both the horizontal and vertical planes are detected simultaneously.
- This method may be used to identify edges in addition to horizontal and vertical planes without any further processing.
- All of this filter's values add up to zero.

Example:

```
% MatLab program for edge sharpening.
% Read the image in variable 'a'
a=imread("cameraman.jpg");

% Defined the laplacian filter.
Lap=[0 1 0; 1 -4 1; 0 1 0];

% Convolve the image read
% in 'a' with Laplacian mask.
a1=conv2(a,Lap,'same');

% After convolution the intensity
% Values go beyond the range.
% Normalise the range of intensity.
a2=uint8(a1);

% Display the sharpened image.
imtool(abs(a-a2),[])
```

```
% Define strong laplacian filter
lap=[-1 -1 -1; -1 8 -1; -1 -1 -1];

% Apply filter on original image
a3=conv2(a,lap,'same');

% Normalise the resultant image.
a4=uint8(a3);

% Display the sharpened image.
imtool(abs(a+a4),[])
```

Explanation of code:

- MatLab program explanation for edge sharpening. a=imread("cameraman.jpg"); This line reads the image in variable a.
- Lap=[0 1 0; 1 -4 1; 0 1 0]; This line defines the Lapalacian filter.
- a1=conv2(a Lap,' same'); This line convolves the image with the Laplacian filter.
- After convolution, values of some pixels go beyond the range [0 255]. Hence the next line is used.
- a2=uint8(a1); This line normalizes the pixel range.
- Sharpened image = Original image – Edge detected image if the central pixel of Laplacian filter is a negative value.
- imtool(abs(a-a2),[]) This line displays the sharpened image.
- lap=[-1 -1 -1; -1 8 -1; -1 -1 -1]; This line defines the strong Laplacian filter, with positive central pixel value.
- a3=conv2(a lap,' same'); This line convolves the original image with this filter.

- a4=uint8(a3); This line normalizes the range of pixel values.
- imtool(abs(a+a4),[]) This line displays the sharpened image.

4.7.2. The unsharp mask filter

Sharpening a digital picture often has the effect of bringing out features that were obscured in the original. By excluding low-frequency spatial information from the original picture, the unsharp mask filter technique serves as a powerful sharpening tool that enhances the definition of tiny detail.

The unsharp mask filter method is used to enhance the clarity of many different types of digital images, and this interactive course delves into the specifics of how it works. The left-hand pane, named Specimen Picture, will initially display a randomly picked specimen image (taken under the microscope). There is a shorthand for the contrast method used to produce the picture after the name of each specimen. Fluorescence (FL), brightfield (BF), darkfield (DF), phase contrast (PC), differential interference contrast (DIC), Hoffman modulation contrast (HMC), and polarised light (POL) are some of the contrast modalities employed. The behaviour of collected specimens in the image processing lesson will vary depending on whatever optical microscope method was used to capture them.

The Filtered Picture window, located next to the Specimen Image window, shows the modified version of the original image after the unsharp mask filter method has been applied. Choose a picture from the drop-down menu labelled "Choose A Specimen," and then play about with the Standard Deviation and Weighting Value controls until the picture seems more focused and detailed.

Subtracting an unsharp mask from the specimen picture is a key step in the method for the unsharp mask filter. An unsharp mask is created by applying a Gaussian low-pass spatial filter on the specimen picture, resulting in a blurred version of the original. For simplicity, this filter may be thought of as a convolution operation on an image using a two-dimensional Gaussian function (g(x,y)) as the kernel mask, as specified by the following equation:

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

The range of frequencies discarded by the Gaussian filter is proportional to the size of the kernel mask, which in turn is a function of the parameter σ. The pixel value of σ is controlled throughout the tutorial using the Standard Deviation slider. The removal of a bigger number of spatial frequencies from the unsharp mask picture is caused by the Gaussian filter when the size of the kernel mask is increased. After that, the unsharp mask is subtracted from the original picture using the following formula:

$$F(x, y) = \frac{c}{2c - 1} I(x, y) - \frac{(1 - c)}{2c - 1} U(x, y)$$

In the equation, the value of the filtered image's pixel at coordinates (x, y) is represented by the function F(x, y), while the values of the corresponding pixels in the original and unsharp mask (blurred) images are represented by the functions I(x, y) and U(x, y), respectively. The difference equation's weights for the original and blurred images are determined by the constant c. In the lesson, the Weighting Value slider may be used to change the value of c anywhere between 1 (the position corresponding to the filtering level of 0 percent) and 5/9 (0.556), which corresponds to the filtering level of 400 percent. The Standard Deviation slider sets the standard deviation (in pixels) of the Gaussian function used to create the kernel mask.

The operation of an unsharp mask filter is shown by the equation that was provided earlier, which shows that the original picture is subtracted from correctly weighted regions of the unsharp mask. High-frequency spatial detail is improved by this subtraction procedure, whereas low-frequency spatial information is reduced. The reason for this is because the Gaussian filter does not eliminate the high-frequency spatial information from the original picture that was removed from the unsharp mask. Furthermore, the Gaussian filter (to the unsharp mask) completely removes low-frequency spatial information from the source picture. This explains why a sharper result is often achieved by raising the size of the Gaussian filter mask before applying the unsharp mask filter.

Since most sharpening filters do not provide any tweakable settings for the user, the unsharp mask filter stands out as a big benefit. The unsharp mask filter, like other sharpening filters, improves the sharpness of edges and the clarity of small details in digital images. Shading distortion, which manifests itself in images most often as subtly shifting background intensities, may be fixed using sharpening filters since these filters also reduce low frequency detail. The sharpening filter has the unintended consequence of making the filtered picture noisier. That’s why the unsharp mask filter has to be used with caution, and it's important to strike a good compromise between sharpening and noise growth.

CHAPTER

5

Fourier transforms and frequency-domain processing

5.1. Frequency space: a friendly introduction

A digital picture is transformed from the spatial domain into the frequency domain in the frequency domain. Application-specific picture enhancement using frequency-domain image filtering. The frequency domain technique known as the Fourier transform is utilised to do this translation from the spatial domain to the frequency domain. A low pass filter is used to soften a picture, whereas a high pass filter is used to bring out fine details. Following the application of both filters, it is subjected to analysis for the ideal filter, as well as the Butterworth filter and the Gaussian filter.

The Fourier transform characterises the space known as the frequency domain. The use of the Fourier transform in the field of image processing is extensive. Indicating the potential distribution of signal energy over a variety of frequencies, frequency domain analysis is utilised.

The method of Fourier transformation is useful in the field of picture processing. Its purpose is to separate a picture into its individual sine and cosine waves. The input picture is in the spatial domain, while the result is in the frequency domain, also known as the Fourier transform. The Fourier transform has several uses, including picture filtering and compression. Processing and reconstructing images, etc.

5.2. Frequency space: the fundamental idea

A frequency distribution transformation is performed first. Then, our black box system will execute whatever processing it needs to complete, and in this particular instance, the output of the black box will not be a picture but rather a transformation. In the spatial domain, it is perceived as an image after undergoing an inverse transformation.

It may be conceptualized visually as

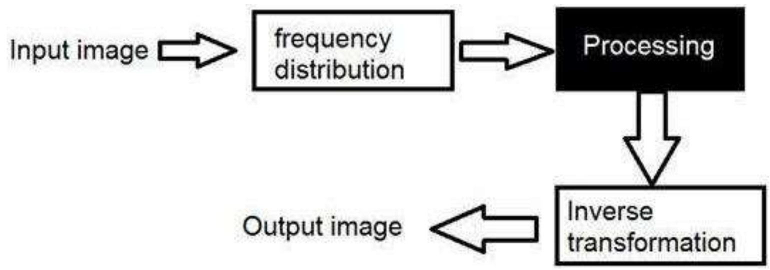


**Figure 5.1 Frequency Domain*

*https://www.tutorialspoint.com/dip/introduction_to_frequency_domain.htm

Transformation

Mathematical operations called transforms may be used to translate a signal from the time domain to the frequency domain. A wide variety of transformations may do this. Below is a list of some of them.

- Fourier Series
- Fourier transformation
- Laplace transform
- Z transform

5.3. Calculation of the Fourier spectrum

Decomposing a picture into its sine and cosine components, the Fourier Transform is a crucial tool in image processing. The picture as it appears in the Fourier or frequency domain is represented by the transformation's output, while the image that is fed into the transformation is its counterpart in the spatial domain. Each dot in the Fourier domain picture stands for a different frequency found in the corresponding pixel in the spatial domain image.

The Fourier Transform has several uses, including those related to image processing (analysis, filtering, reconstruction, and compression).

Working

This explanation will be limited to the Discrete Fourier Transform (DFT) because we are only interested in digital pictures.

As the Discrete Fourier Transform (DFT) is the sampled Fourier Transform, it does not include all frequencies contributing to an image but a collection of samples that is big enough to depict the picture in its whole in the spatial domain. Since the number of frequencies is equal to the number of pixels in the spatial domain picture, this indicates that the images in the spatial domain and the Fourier domain have the same dimensions.

The two-dimensional discrete Fourier transform for a N × N square image is:

$$F(k,l) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} f(i,j)\, e^{-\iota 2\pi \left(\frac{ki}{N} + \frac{lj}{N}\right)}$$

When *f(a,b)* represents the image in the spatial domain and the exponential term is the basis function that corresponds to each point *F(k,l)* in the Fourier space, the picture is in the spatial domain. Each point's value at *F(k,l)* is calculated by multiplying the spatial image by the appropriate base function and adding the resulting products, according to one interpretation of the equation.

Specifically, *F(0,0)* stands in for the image's average luminance, or the DC-component, whereas *F(N-1,N-1)* stands in for the greatest frequency of the sine and cosine waves that make up the basis functions.

The Fourier picture may be spatially re-transformed in a similar fashion. The formula for the inverse Fourier transform is:

$$f(a,b) = \frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} F(k,l)\ e^{\iota 2\pi \left(\frac{ka}{N} + \frac{lb}{N}\right)}$$

Take into account the normalisation term of $1/N^2$ in the inverse transformation. This normalisation is occasionally used to the forward transform rather than the inverse transform, but it should not be used for both of these transformations at the same time.

Each picture point requires a double sum to be computed in order to acquire the solution for the aforementioned equations. The Fourier Transform, however, is decomposable, hence it may be expressed as

$$F(k,l) = \frac{1}{N} \sum_{b=0}^{N-1} P(k,b)\ e^{-\iota 2\pi \frac{lb}{N}}$$

Where

$$P(k,b) = \frac{1}{N} \sum_{a=0}^{N-1} f(a,b)\ e^{-\iota 2\pi \frac{ka}{N}}$$

Based on these two equations, N one-dimensional Fourier Transforms are applied to the spatial domain picture to produce an intermediate image. From that, N one-dimensional Fourier Transforms are applied to the intermediate picture to get the final image. It is possible to reduce the amount of calculations needed for the two-dimensional Fourier Transform by writing it as a sum of 2N one-dimensional transforms.

The typical one-dimensional DFT still has N^2 complexity, even when these optimizations are applied. If we use the Fast Fourier Transform (FFT) to calculate the one-dimensional DFTs, this time requirement drops to $N \log_2 N$. The boost is especially noticeable when dealing with huge photos. Depending on the specific implementation of the FFT, the maximum size of the input picture that may be converted is usually limited to $N=2^n$, where n is an integer. The literature provides a thorough description of the underlying mathematical facts.

The real and imaginary halves, or magnitude and phase, of a complex number are both represented in the two pictures that result from applying the Fourier Transform. Because it provides so much information on the geometric structure of the spatial domain picture, the magnitude of the Fourier Transform is frequently all that is shown in image processing. It is important to keep the magnitude and phase of the Fourier image unchanged if we plan on re-transforming it back into the right spatial domain after some processing in the frequency domain.

A picture in the Fourier domain might cover a lot more ground than the same image in the spatial domain.

Therefore, its values are often computed and stored as float values to provide appropriate accuracy.

5.4. Complex Fourier series

A periodic function may be expressed as the sum of sine and cosine waves, which is what a Fourier series does. Each harmonic wave in the sum has a frequency that is a whole number multiple of the fundamental frequency of the periodic function. In order to find the phase and amplitude of each harmonic, a technique called "harmonic analysis" must be used. The number of harmonics in a Fourier series might be limitless. An approximation to a function may be obtained by summing some but not all of the harmonics in its Fourier series. An example of this would be applying the first few harmonics of the Fourier series to the problem of describing a square wave; the result would be an approximation of the square wave.

A convergent Fourier series may be used to represent almost any periodic function. Convergence of Fourier series indicates that the total of partial Fourier series becomes closer and closer to the true function as more and more harmonics are added. Eventually, the sum of all partial Fourier series will equal the true function, even if there are an unlimited number of harmonics. All the related mathematical proofs may be grouped under the name "Fourier Theorem."

Only periodic functions may be represented using Fourier series. However, an extension of the Fourier series known as the Fourier transform may be used to manage non-periodic functions by treating them as periodic with unlimited period. A waveform's ability to be translated between its time domain representation and its frequency domain representation is made possible by a transform that can yield frequency domain representations of non-periodic functions in addition to periodic functions.

Since Fourier's time, other methods have been developed to define and explain the notion of Fourier series; these methods are compatible with one another but place differing emphasis on certain parts of the issue. Some of the more effective methods rely on mathematical concepts and procedures that did not exist during Fourier's day but have become standard fare. At first, Fourier defined the Fourier series for real-valued functions with real arguments, and he did so by decomposing the sine and cosine functions as the fundamental examples. Since then, numerous more transformations associated with the Fourier series have been created, broadening the scope of his original notion and giving rise to a new branch of mathematics known as Fourier analysis.

The Fourier series on the square has several applications, including partial differential equations like the heat equation and picture compression. Discrete variant of the Fourier cosine transform, using cosine alone as the basis function, is used in the jpeg image compression standard.

Half of the Fourier series coefficients vanish for two-dimensional arrays having a staggered appearance, because of extra symmetry.

Decomposing a picture into its sine and cosine components, the Fourier Transform is a crucial tool in image processing. The transformation produces a Fourier or frequency domain representation of the picture from the input image, which is in the spatial domain. Each dot in the Fourier domain picture stands for a different frequency found in the corresponding pixel in the spatial domain image.

The Fourier Transform has several uses, including those related to image processing (analysis, filtering, reconstruction, and compression).

5.5. The 2-D Fourier transform

The two-dimensional (or 2D) Fourier transform is a time-tested method in the field of image analysis. The well-known Fourier transform for signals, which breaks down a signal into a sum of sinusoids, has a more generalized counterpart called the Fast Fourier Transform (FFT). Therefore, the Fourier transform reveals details about the image's frequency composition.

The Fourier Transform, which will be referred to as the 2D Fourier Transform below, is the series expansion of an image function (within the context of the 2D space domain) expressed in terms of "cosine" image (orthonormal) basis functions.

The definitions of the inverse transform and the transform (to expansion coefficients) are provided below:

```
    F(u,v) = SUM{ f(x,y)*exp(-j*2*pi*(u*x+v*y)/N) }
and
    f(x,y) = SUM{ F(u,v)*exp(+j*2*pi*(u*x+v*y)/N) }

where u = 0,1,2,...,N-1 and v = 0,1,2,...,N-1
      x = 0,1,2,...,N-1 and y = 0,1,2,...,N-1
      j = SQRT( -1 )
      and SUM means double summation  over proper
      x,y or u,v ranges
```

Before diving into the Fourier Transform proper, let's take a look at its "basic" functions. Every picture is reduced to a sum of cosine-like components in the FTs attempt to represent it. Since cosines are the simplest of the wave functions, pictures that are entirely composed of cosines have FTs that are very easy to understand.

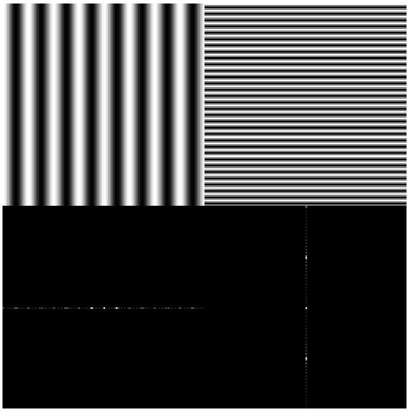
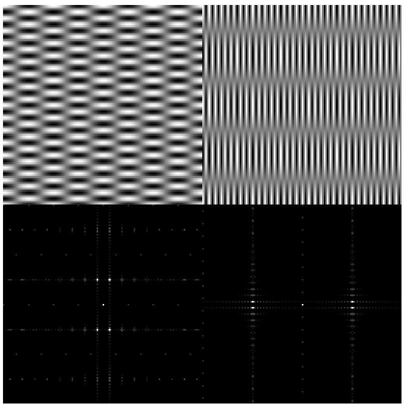


**Figure 5.2 2-D Fourier Transform*

*https://www.cs.unm.edu/~brayer/vision/fourier.html

Two pictures are shown, each with its Fourier Transform displayed just below it. Both the horizontal and vertical pictures are pure cosines of 8 and 32 cycles, respectively. It should be noted that the FT for each only consists of a single component, which is shown by two bright spots that are arranged symmetrically around the middle of the FT picture. The frequency x, y origin is located near the middle of the picture. As the horizontal component of frequency, the u-axis goes from left to right along the middle. The vertical (or v) axis is in the middle and represents the vertical frequency component. The (0,0) frequency term, or average value, of the picture is represented by a dot in the middle of both. For this reason, FT pictures often exhibit a bright blob of components towards the centre, where the average value is high (such as 128) and where there is a great deal of low frequency information. Take note that dots that are brighter on the edges of the vertical direction are caused by higher frequencies in that direction. Additionally, it is important to note that high frequencies in the horizontal direction will result in bright dots that are located distant from the centre in the horizontal direction.

Two images of the more generic Fourier components are shown below. They are images of the horizontal and vertical components of cosines in two dimensions. On the left, we see a pattern with four horizontal and sixteen vertical repetitions. The rightmost one consists of 32 horizontal cycles and 2 vertical ones. (Note that the grey band appears whenever the function passes through grey = 128, which occurs twice every cycle.) Some symmetry may start to stand out to you. Since the FT is symmetrical around the origin, the first and third quadrants are identical for all REAL (as opposed to IMAGINARY or COMPLEX) images, and vice versa for the second and fourth. Four-fold symmetry derives from x-axis symmetry (as in the cosine images).



5.6. The inverse Fourier transform and reciprocity

From a Fourier transform, an inverse discrete Fourier transform may calculate the original picture by doing the following:

$$f(m,n) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) e^{+j\,2\pi\left(\frac{um}{M} + \frac{vn}{N}\right)}$$

It is represented below as F-1.

Properties

- The DFT is linear:

$$\mathcal{F}[af + bg] = aF + bG \qquad \text{where } a, b \in \mathbb{C}.$$

Multiplying the discrete Fourier transforms (DFTs) of two images is comparable to performing a convolution:

$$f * g = \mathcal{F}^{-1}[F \times G]$$

- One way to achieve the 2D DFT is to first compute the 1D DFT on the rows, and then the 1D DFT on the columns (the DFT is separable).
- The DFT is periodic with periods M and N $(k, l \in \mathbb{Z})$:

$$F(u, v) = F(u + kM, v) = F(u, v + lN) = F(u + kM, v + lN)$$

When the image is translated, the corresponding DFT phase shift occurs:

$$\mathcal{F}[f(m - m_0, n - n_0)] = F(u, v) \exp\Big(-j2\pi \left(\frac{um_0}{M} + \frac{vn_0}{N}\right)\Big)$$

Any rotation made to the image will result in the same rotation being made to the DFT.

5.7. Understanding the Fourier transform: frequency-space filtering

By removing high or low frequency components, Frequency Domain Filters may be used to smooth and sharpen an image. Extremely high and low frequencies may be filtered out sometimes. In contrast to filters that operate in the spatial domain, those that operate in the frequency domain concentrate primarily on the frequencies that are present in the images. There are two primary purposes for this process: smoothing and sharpening.

There are three categories for these:

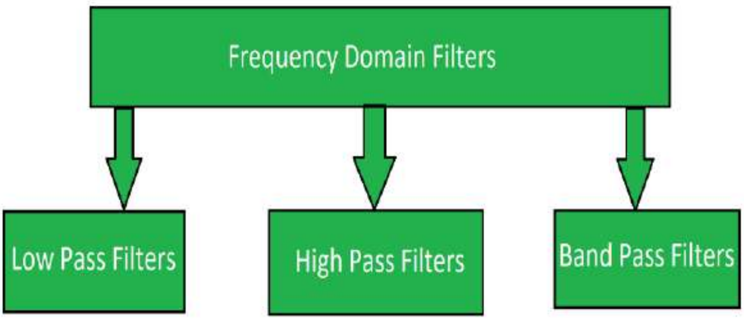


**Figure 5.3 Classification of frequency domain filters*

1. Low pass filter: Since a low pass filter is designed to filter out higher frequencies, it is designed to preserve lower frequencies. It's a standard image for making images seem less choppy. As a means of image smoothing, it works by reducing the prominence of high-frequency details while leaving low-frequency details unaltered.

---

*https://www.geeksforgeeks.org/frequency-domain-filters-and-its-types/#:~:text=Frequency%20Domain%20Filters%20are%20used,the%20frequency%20of%20the%20images.

In the frequency domain, the mechanism of low-pass filtering may be represented as:

```
G(u, v) = H(u, v) . F(u, v)
where F(u, v) is the Fourier Transform of original image
and H(u, v) is the Fourier Transform of filtering mask
```

2. High pass filter: If a high pass filter is used, the low frequency components will be removed, but the high frequency components will be preserved. The image is sharpened with its help. The image is sharpened by reducing the impact of low-frequency elements while keeping high-frequency details intact.

The following provides the frequency domain high pass filtering mechanism:

```
H(u, v) = 1 - H'(u, v)
where H(u, v) is the Fourier Transform of high pass filtering
and H'(u, v) is the Fourier Transform of low pass filtering
```

3. Band pass filter: As the name suggests, a band pass filter is designed to pass just the frequencies in the middle frequency range, letting through only the extremely low and very high ones. Band pass filtering may improve edges while simultaneously decreasing noise levels.

5.1 The convolution theorem

By using the convolution theorem, we may determine how the spatial domain is related to the frequency domain.

As a representation of the convolution theorem:

$$f(x,y)*h(x,y) \leftrightarrow F(u,v)H(u,v)$$

$$f(x,y)h(x,y) \leftrightarrow F(u,v)*H(u,v)$$

$$h(x,y) \leftrightarrow H(u,v)$$

It is possible to express it by saying that filtering in the frequency domain is equivalent to convolution in the spatial domain, and vice versa.

The filtering may be expressed in the frequency domain as follows:

Figure 5.4 Filtering in frequency domain[*]

In image processing, the convolution theorem asserts that multiplying the Fourier transforms of two signals is equivalent to convolving them. Thus, like other theorems about the Fourier transform, is helpful because it provides us a different image from which to view the actions we do while processing images.

[*] https://www.tutorialspoint.com/dip/convolution_theorm.htm

Let's ignore 2-D images in favour of 1-D communications like digital audio signals. Let's pretend for a moment that the one-dimensional discrete signal and [1 1 1]/3 filter are the two signals in question. Without using the convolution theorem, this may be seen as swapping out individual points in the 1-D signal with an average of the two points immediately to either side. Yes, it sheds light on the situation. Since large fluctuations are likely to be smoothed down by averaging with neighbouring data, averaging causes the signal to shift more slowly.

The convolution theorem makes it easier to visualise what this filtering operation does to the signal's temporal frequencies (or spatial frequencies, of course; here, I'm focusing on a 1-dimensional signal, which naturally lends itself for being conceived of as a signal in the time domain, such as an audio signal). We may examine the 1-D signal's original shape using Fourier analysis, which suggests that the signal contains a variety of temporal frequencies. The filter's Fourier transform is a sine function, denoted as k*sin(af)/(af), where k, a, and f are constants, and time frequency, respectively.

According to the convolution theorem, multiplying the Fourier transform of the original 1-dimensional signal by this sine function yields the same result in the Fourier domain as convolving with [1 1 1]/3. Where does it lead us, to a first approximation, we can see that filtering our 1-D signal with [1 1 1]/3 will result in the multiplication of low temporal frequencies by a relatively large number and the multiplication of high temporal frequencies by a relatively small number. This is because the sine function has a lot of bumps and wiggles. Low temporal frequencies will be substantially preserved while higher frequencies will be suppressed (although not totally). Consistent with our understanding from considering convolution in the time-domain, as mentioned above.

This theorem may even be used to speed up convolution in some circumstances. To filter, convert both signals to the Fourier domain, multiply them, and then convert back to the time or space domain. Both multiplication and performing the Fourier transform are quick operations.

The convolution theorem is also useful when combined with other Fourier theorems, as in the following applications:

- A thorough comprehension of what takes on throughout the process of converting a continuous analogue audio signal into a discrete digital signal
- Comprehending the concept of aliasing in computer graphics
- Creating improved filters
- Having a basic understanding of how an AM radio operates

5.8. The optical transfer functions

In order to describe how various spatial frequencies are processed by an optical device like a camera, microscope, human eye, or projector, one must look at its optical transfer function (OTF). Light is projected onto a photographic film, detector array, retina, screen, or simply the next component in the optical transmission chain, and this is a term used by optical engineers to explain how the optics work. Modulation transfer function (MTF) is a version that is equal to the OTF in many cases while ignoring phase effects.

Either transfer function describes how the lens system reacts to a sinusoidal waveform as a function of the wave's spatial frequency or period and the direction in which it is incident. In mathematical terms, the OTF may be expressed as the Fourier transform of the point spread function (PSF, which stands for "point source field," is the impulse response of the optics and represents the picture of a point source.). The OTF is complex-valued since it is a Fourier transform, but in the usual scenario of a symmetric PSF, its values will be real. The magnitude (absolute value) of the complex OTF is the official definition of the MTF.

Panels (a) and (b) to the right of the picture depict the optical transfer functions for two distinct optical systems (d). The former is representative of a diffraction-limited, circular-pupil ideal imaging system. Its transfer function drops down somewhat gradually with increasing spatial frequency up until it meets the diffraction-limit, which in this instance occurs at 500 cycles per millimetre or a period of 2 μm. This imaging system has a resolution of 2 μm because it can detect periodic structures with a period as tiny as this period. In panel (d), an unfocused optical system is shown. When compared to a diffraction-limited imaging system, this drastically lowers contrast. About 250 cycles/mm, or periods of 4 μm, the contrast is completely attenuated. This explains why the diffraction-limited system (d) produces sharper pictures than the out-of-focus system (e,f) (b,c). Keep in mind that the out-of-focus system has diffraction-limited contrast around the diffraction limit of 500 cycles/mm, but has extremely poor contrast at spatial frequencies about 250 cycles/mm. If you look closely at the picture in panel (f), you can see that the spoke structure is pretty crisp for the high spoke densities close to the centre of the spoke target.

Because it is derived from the Fourier transform of the point-spread function (PSF), the optical transfer function (OTF) is often a function of spatial frequency with complex values. A complex number having an absolute value and complex argument proportionate to the relative contrast and translation of the projected projection, respectively, is used to indicate the projection of a certain periodic pattern. This number also has a complex argument.

In many cases, a pattern's decrease in contrast is more important than its translation. The absolute value of the optical transfer function, also known as the modulation transfer function, is what determines the relative contrast of an image. How much contrast of the item is recorded in the picture as a function of spatial frequency may be inferred from its values. Although the MTF typically decreases from 1 to 0 (the diffraction limit) as the spatial frequency increases, this relationship is not always monotonic. In contrast, the complex argument of the optical transfer function may be represented as a second real-valued function, generally referred to as the phase transfer function (PhTF), where also the pattern translation is relevant.

5.9. Digital Fourier transforms: the discrete fast Fourier transform

The discrete Fourier transform, also known as the DFT, is a mathematical operation that turns a finite series of equally-spaced samples of a function into a same-length sequence of equally-spaced samples of the discrete-time Fourier transform, also known as the DTFT. The DTFT is a complex-valued function of frequency. The DTFT is sampled at intervals proportional to the inverse of the input sequence's duration. The samples from the DTFT are used as the coefficients of complex sinusoids at the relevant DTFT frequencies in an inverse DFT, which is a Fourier series. It is a series of sample values that is identical to the input sequence. It is for this reason that the discrete Fourier transform (DFT) is referred to as a frequency domain representation of the initial input sequence. If the original sequence includes both zero and nonzero values, then the DTFT of that function is continuous (and periodic), whereas the DFT delivers discrete samples of a single cycle. When applied to a sequence that represents one cycle of a periodic function, the DFT yields all the non-zero values that occur during that cycle.

In many real-world contexts, Fourier analysis is performed using the DFT, making it the most essential discrete transform. The function in digital signal processing is a time-varying quantity or signal, such as the amplitude of a sound wave, the frequency of a radio broadcast, or the average daily temperature (often defined by a window function). Pixel values along a row or column of a raster picture may serve as samples in image processing. The DFT is also useful for performing other operations, such as convolutions or multiplying big numbers, quickly, and for solving partial differential equations.

Since it only involves a limited quantity of data, it is possible to implement it in computers using numerical techniques or even hardware that is specifically designed for the purpose. These implementations often make use of effective methods for the fast Fourier transform (FFT). In fact, the names "FFT" and "DFT" are sometimes used interchangeably because of how similar they are. At times, it is unclear what is meant by the term "finite Fourier transform," to which the "FFT" initialism may have previously referred.

CHAPTER 6

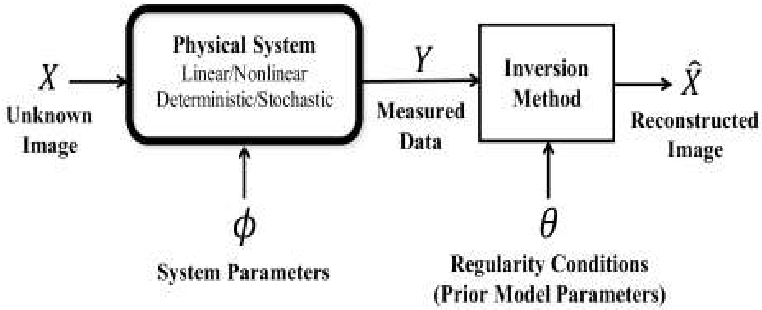
Image restoration

In order to approximate the ideal image field that would be seen if no image degradation were present in an imaging system, image restoration may be thought of as an estimating process in which operations are conducted on an observed or measured image field. In this chapter, we provide a mathematical model for image restoration across broad categories of imaging equipment.

6.1. Imaging models

The term "model-based image processing" refers to a group of methods that have been developed over the course of the last several decades. These methods provide a methodical framework for the solution of inverse issues that are posed by imaging applications.

When trying to solve an imaging issue, you may find yourself trying to solve an inverse problem, where you try to reconstruct a previously unseen image (X) from a set of measurements (Y). It is fairly uncommon for the physical system's properties and the regularity criteria to be determined by two extra, mysterious "nuisance" parameters, represented by θ.



[*]Figure 6.1 Imaging problems

Most inverse issues have the shape seen in Figure 6.1.

An unidentified signal or image (X) is used by some kind of physical system to generate some kind of measurable output (Y). The goal is to use this data to reconstruct the mysterious signal or image X. Due to the fact that X is not directly seen, the issue of deducing X from Y is referred to as an inverse problem. This is because it is necessary for the physical process that resulted in the observations to be inverted or reversed in order to get X from Y.

In practise, imaging systems often encounter inverse difficulties. In this sense, Y might stand in for the voltage read-outs from a CMOS sensor in a mobile phone camera or the measurements of a volume X acquired by an optical or

[*]https://engineering.purdue.edu/~bouman/publications/pdf/MBIP-book.pdf

electron microscope. Alternately, Y might stand for the measurements obtained from a radio telescope on unidentified astronomical objects, or it could refer to the photon counts obtained from a medical PET scanner. This common framework is shared by all of these imaging systems and many more besides.

In general, the structure or components of any method used to compute the answer to an inverse issue will be similar. The goal of any inversion method is to get a value for X, the estimated value of the unknown image, from Y, the observed value. Quite often, there are also unknown nuisance parameters of the system, which we will designate with the symbol φ. Unknown calibration factors, such as focus or noise gain, are often of little direct relevance but must be identified in order to solve the inversion issue, and these parameters might stand out from them. The degree of regularisation or smoothing that is necessary for the inversion process is determined by the value of parameter, which is denoted by θ.

Because probability is the cornerstone of the model-based method, the physical system and the image to be discovered, X, are both treated as random quantities. $P(y|x)$, the conditional distribution of the data, Y, given the unknown image, X, is an example of the so-called forward model of the system. The assumed prior distribution, p, represents the prior model of the unobserved image (x). $p(y|x)$ intuitively reveals all information regarding the relationship between the observations and the unknown.

This covers both the deterministic features of the imaging system, such as its geometry and sensitivity, as well as the probabilistic properties, such as the noise amplitude. For example, the geometry of the imaging system is an example of a deterministic property.

Quantitatively characterising the image deterioration impacts of the physical imaging equipment, the image digitizer, and the image display is crucial for designing an efficient digital image restoration system. Modeling the consequences of image deterioration and then undoing the model using operations yields a restored image. It is important to underline the fact that precise image modelling is often the key for successful image restoration. It is possible to simulate the consequences of image deterioration using either a priori or a posteriori method. In the former, the imaging system, digitizer, and display are all measured to find out how they react to a certain image field. In certain scenarios, it will be able to represent the reaction of the system in a deterministic manner, while in other circumstances, it will be possible to predict the response of the system only in a stochastic way. The goal of the posteriori modelling strategy is to construct a model of the image degradations using just the data from the image that needs to be recovered. The primary difference between the two methods is in the details of the data collection used to characterize the image degradation.

6.2. Nature of the point-spread function and noise

The captured image in fluorescence microscopy is always an approximation of the true object. The so-called Point Spread Function (PSF) characterizes this fuzziness. How a single point inside an item appears in an image is described by the Point Spread Function (PSF).

A light microscope's image production process is linear, so if you take a picture of two things, A and B, at the same time, you'll get a picture that's the same as the picture of only one of them. Because of this linearity quality, it is possible to reconstruct an image of any given object by first splitting it into smaller portions, then imaging each of these, and then combining the resulting images. The item may be broken down into infinitesimally little point objects by subdividing it into smaller and smaller pieces. PSFs are created in the image by each of these point sources, but they are moved and scaled according to the position and brightness of the source points. Therefore, the final image is a mosaic of PSFs that often overlap with one another. The mathematical representation of how an image is formed is a convolution equation, where the object is convolved with the point spread function of the imaging system to get the obtained image.

The PSF is an accurate indicator of an optical system's quality since it shows how points are blurred in an image. Because the point spread function (PSF) is always normalised (that is, the integral across its full width is equal to 1), it is simple to compare the PSFs of various systems and by extension compare the imaging quality of each system.

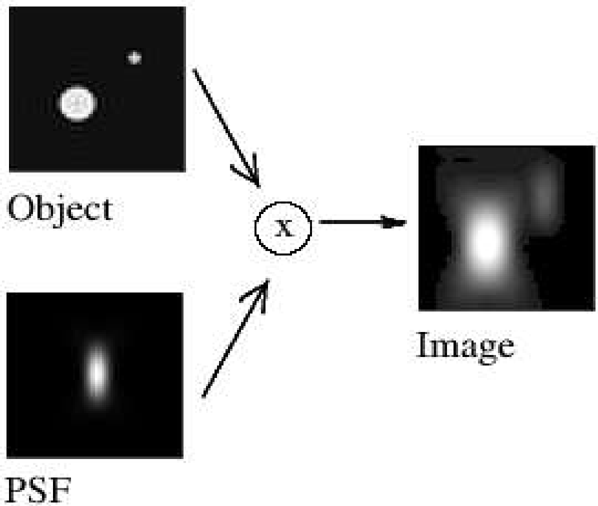


Figure 6.2 Point Spread Function [*]

Noise

Image noise in digital photography is analogous to film grain in traditional cameras. Image noise often appears as random speckles on a smooth surface, and its presence may have a significant negative impact on the overall quality of the image. However, there are situations in which it may be beneficial to improve the perceived sharpness of a digital image.

---

[*]https://svi.nl/Point-Spread-Function-(PSF)

Increases in noise tend to occur when exposure times, ambient temperatures, and/or camera sensitivities are all increased. The quantity of certain kinds of image noise that are present at a particular setting varies from one camera model to another and is directly tied to the technology that is used in the sensor.

Three Types of Image Noise

Random noise, fixed pattern noise, and banded noise are the three most common forms of image noise. Fluctuations in hue beyond the image's true brightness indicate random noise.

Long exposures at high temperatures result in noise with a consistent pattern.

Banding noise, which is directly tied in-camera technological aspects, is also created when the camera reads data from the sensor.

By going to manual exposure mode and modifying the parameters that often create noise on a certain camera model, a photographer may eliminate some forms of image noise. The camera may have a noise reduction option that may be used in certain situations. This characteristic is typical with more expensive cameras.

The alternative is to expose the image as brightly as possible, such that as little shadow as possible is present.

The physical temperature of the camera should be lowered by putting it away for a while before usage.

Luminosity noise and color noise are the two main categories of image noise.

The individual controls are easily accessible in the panels. You can keep tabs on your development by creating frequency layers. To discover the optimal middle ground, some trial and error is required.

6.3. Restoration by the inverse Fourier filter

Even though Wiener filtering is the best compromise between inverse filtering and noise smoothing, it actually amplifies the noise when the blurring filter is a unique value. As a result, it seems that a denoising process is required to get rid of the boosted noise. The wavelet-based denoising strategy offers a natural approach that may be used for this purpose.

An image new method for the restoration of images is suggested, and this method is broken up into two distinct stages, namely wavelet-domain image denoising and Fourier-domain inverse filtering. The first step involves applying a Wiener filter to the input image, and then feeding the filtered image into the adaptive threshold wavelet denoising stage. The selection of the threshold estimate is accomplished by conducting an investigation into the statistical characteristics of the wavelet sub band coefficients. These statistical parameters include the standard deviation, the arithmetic mean, and the geometrical mean. We initially extract the various frequency bands by decomposing the noisy image into numerous levels. Then the noisy coefficients are eliminated using soft thresholding, which involves determining the optimal thresholding value.

Based on experimental findings using a test image, this approach has been shown to provide a far higher Peak Signal to Noise Ratio (PSNR) and overall image quality .In order to demonstrate the effectiveness of this approach in image restoration, we have compared it against a variety of other restoration methods, such as the Wiener filter by itself and the inverse filter.

Image restoration is the technique of improving the look of an image by using a restoration procedure that removes image deterioration via a mathematical model. This method is used to restore images. Examples of deterioration include geometric distortion brought on by flawed lenses, overlaid interference patterns brought on by mechanical systems, and noise from electronic sources. For instance, while taking pictures with a CCD (Charge Coupling Device) camera, light levels and sensor temperature have a significant impact on the amount of noise in the final image.

The noise function and the degradation function are the two main components of the degradation process model. The overarching geographical model looks like this:

$$d(x,y) = h(x,y) ** f(x,y) + n(x,y)$$

Where the ** denotes two dimensional convolution process

d(x,y) = degraded image

h(x,y) = degradation function

f(x,y) = original image

n(x,y) =additive noise function

Because the operation of convolution in the spatial domain is analogous to the operation of multiplication in the frequency domain, the frequency domain model is:

$$D(u,v) = H(u,v)F(u,v) + N(u,v)$$

where D(u,v) = Fourier transform of the degraded image

H(u,v) = Fourier transform of the degradation function

F(u,v) = Fourier transform of the original image

N(u,v) = Fourier transform of the additive noise function

Therefore, the challenge of restoring an image from one that has been deteriorated is known as the linear image restoration problem in this paradigm, as the genuine image and the noise are connected linearly.

6.4. The Wiener–Helstrom Filter

A number of different strategies have been offered in the body of academic research in order to recover an image that has been degraded as a result of blurring and additive noise.

The inverse filter, the Wiener filter, the parametric Wiener filter, the power spectrum filter, and the geometric mean filter are all examples.

When an image is blurred by a known low pass filter, it is feasible to recover the image by using inverse filtering or extended inverse filtering. This is a restoration approach for deconvolution. Inverse filtering, however, is very vulnerable to additive noise. Vienna Filtering is a great compromise between inverse filtering and noise smoothing. Together, the additive noise is nullified and the blurring is inverted. Because of this, the Wiener filtering method achieves the best results in terms of the mean square error. This is due to the fact that it reduces the overall mean square error while simultaneously smoothing out the noise.

Even though Wiener filtering is the best compromise between inverse filtering and noise smoothing, it actually amplifies the noise when the blurring filter is a unique value. Since the noise has been enhanced, this indicates that a denoising step is required to get rid of it. It is crucial to use image denoising algorithms in order to get rid of the random additive noises while keeping as many of the key signal properties as feasible. Some statistical filters, such as

Average filter, may be used to get rid of these disturbances, however wavelet-based denoising algorithms have shown to be more effective. In most cases, image de-noising requires a trade-off between minimising noise and protecting crucial image information. Specifically, a well-performing denoising algorithm will learn to handle image transitions gracefully.

Building spatially adaptable algorithms is a natural process aided by the wavelet representation. It does this by condensing the fundamental information in a signal into a relatively small number of big coefficients, which represent image features at varying resolution scales. Wavelet offers a suitable foundation for separating noisy signal from image signal, that's why there has been a significant amount of study on wavelet thresholding and threshold selection for signal and image denoising during the last several years. The majority of these wavelet-based thresholding methods have shown superior efficiency in image denoising. By examining the statistical properties of the wavelet coefficients, we examine a thresholding strategy that is effective for image denoising.

To recover an image after it has been blurred by a lowpass filter whose parameters are known, a deconvolution method called inverse filtering may be used. Inverse filtering, however, is very vulnerable to additive noise. We may create a restoration algorithm for each kind of deterioration and then simply merge them using the step-by-step technique of lowering each degradation individually. An ideal compromise between inverse filtering and noise smoothing is achieved by the Wiener filtering. Together, the additive noise is nullified and the blurring is inverted.

Vienna filtering minimises the square of the mistake, making it the best option. In other words, it does inverse filtering and noise smoothing while minimising the mean square error. A linear approximation of the original image is what the Wiener filtering does. Stochastic modelling is the basis of this strategy. Since the Wiener filter in the Fourier domain is subject to the orthogonality principle, it may be written as follows:

$$W(f_1, f_2) = \frac{H^*(f_1, f_2) S_{xx}(f_1, f_2)}{|H(f_1, f_2)|^2 S_{xx}(f_1, f_2) + S_{\eta\eta}(f_1, f_2)},$$

Where the power spectra of the original image and the additive noise are denoted by $S_{xx}(f_1, f_2)$ and $S_{nn}(f_1, f_2)$, respectively, and the blurring filter is denoted by $H\ (f_1, f_2)$. The Wiener filter may be broken down into two distinct components: the inverse filtering section and the noise smoothing section. In addition to remove the noise using a compression operation, it also carries out the deconvolution via inverse filtering (highpass filtering or lowpass filtering).

Implementation

In order to put the Wiener filter into operation, it is necessary for us to make an estimate of the power spectra of both the original image and the additional noise. With white additive noise, the power spectrum is proportional to the noise's variance. A wide variety of techniques may be employed to make an educated guess regarding the power spectrum of the source image. An example of a direct estimate would be the period gram estimate of the power spectrum that is generated based on the observation:

$$S_{yy}^{per} = \frac{1}{N^2}[Y(k,l)Y(k,l)^*]$$

If the DFT of the observation is Y(k,l), then the phrase "where" is unnecessary. The estimate's main benefit is its simplicity of implementation, which eliminates the need to deal with the singularity of inverse filtering. Another estimate that, when combined with the previous one, results in a cascade implementation of inverse filtering and noise smoothing is as follows:

$$S_{xx} = \frac{S_{yy} - S_{\eta\eta}}{|H|^2},$$

Which is an obvious consequence of the fact: $S_{yy} = S_{nn} + S_{xx}|H|^2$. With the help of the period gram estimate, it is possible to immediately derive from the observation an estimation of the power spectrum S_{yy}. When using this approximation, inverse filtering and noise smoothing are applied in a cascading fashion:

$$W = \frac{1}{H} \frac{S_{yy}^{per} - S_{\eta\eta}}{S_{yy}^{per}}.$$

One of the drawbacks of using this solution is that when the inverse filter is single, we have no choice but to make use of the generalised inverse filtering. It has also been suggested that models like the $1/f^a$ model may be used to approximate the power spectrum of the original image.

Experimental Result

In order to demonstrate the Wiener filtering process that is used in image restoration, we will use the typical Lena test image that is 256 pixels x 256 pixels. We use a lowpass filter to make the image blurry.

$$H = \frac{1}{16} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

After that, add 100 standard deviations of white Gaussian noise to the image that had already been blurred. To improve the quality of the image, we use a cascaded implementation of inverse filtering and noise smoothing to apply the Wiener filter. Below you'll find a list of the photos along with their corresponding PSNRs and MSEs. You have noticed that the visual performance of the recovered image has improved, despite the MSEs not reflecting this improvement. This is because MSE is not an appropriate statistic for deconvolution.

6.5. Origin of the Wiener–Helstrom filter

The Wiener filter is a linear time-invariant (LTI) filter used in signal processing to generate an approximation of a target random process from the observed noisy process under the assumptions of stable signal and noise spectra and additive noise. The Wiener filter is designed to reduce, as much as possible, the mean square error that occurs when comparing the estimated random process to the intended process.

Norbert Wiener suggested the filter in the 1940s, and it was published in 1949. Andrey Kolmogorov independently calculated the discrete-time equivalent of Wiener's idea, which he published in 1941. Filtering theory after Wiener and Kolmogorov for this reason. Many other filters, such as Kalman filter, owe their existence to the Wiener filter, the first filter of its kind to be statistically developed.

The Wiener filter's purpose is to calculate a statistical estimate of a signal for which the value is unknown by taking as its input a signal to which it is connected and then filtering the known signal in order to obtain the estimate as its output. For instance, the known signal might be made up of a previously unknown signal of interest that has been tainted by additive noise. To estimate the original, uncorrupted signal, the Wiener filter may be used to remove the background noise.

A typical deterministic filter will have a certain frequency response in mind when it's created. The Wiener filter, on the other hand, is built in a different way. The goal is to find the linear time-invariant filter whose output is most similar to the original signal, given that one is supposed to know the spectral features of both the signal and the noise. There are a few distinguishing features of Wiener filters:

1. **Assumption:** The stationary linear stochastic processes that make up the signal and noise have known spectral properties or known auto- and cross-correlations.
2. **Requirement:** The filter has to be causal and physically realizable (this requirement can be dropped, resulting in a non-causal solution)
3. **Performance criterion:** minimum mean-square error (MMSE)

In the process of deconvolution, this filter is used often for more information on this application, see Wiener deconvolution.

6.6. Constrained deconvolution

The term "deconvolution" refers to a computer approach that was developed to partially correct for the picture distortion that was brought on by the usage of a microscope. Improvements in both spatial resolution and the dampening of out-of-focus light may be substantial. The technique was first developed at MIT for use in seismology, but it has found other uses in fields as diverse as astronomy and 3D optical fluorescence microscopy.

Since it may create artefacts or further degrade poor quality photographs, it should not be seen as a "black box" to better image quality.

It can work with numerical scales (should even improve). In order to get optimal results, the sample must be thin (50 um), transparent to light, opaque, and brilliant.

Live microscopy is difficult because of the short exposure time required to minimize motion blur (limit spherical aberrations).

With convolution, we construct a picture by summing the overlapping contributions of neighboring points and then replacing each original point with its blurred image in all dimensions.

In order to digitally de-convolve noisy, degraded photographs of incoherently lighted objects, a general-purpose alternative to the approach of spatial filtering is presented by capitalizing on the identity between the processes of vector convolution and polynomial multiplication. The technique is somewhat linked to linear programming techniques, but it drastically departs from them by making use of convolution's unique characteristics. Arrays of sampled images are seen as discrete points in an n-dimensional Euclidean space. Linear restrictions on the restored image-irradiance values are defined by the convolution relation in addition to limitations on individual recorded and point-spread image irradiance values. A convex set of feasible restorations in n-space is defined by these restrictions. A technique is provided here for picking a point (i.e., an estimate of the restored picture) from this region that is somewhat close to the center of the area. The human observer may then make any necessary adjustments to the initial limitations in order to take into account the newly discovered information that was brought to light by his interpretation of the restored-image estimate. It is therefore possible to rerun the deconvolution computations while taking into account the new limitations, which may result in a more accurate approximation. Both the recorded picture and the point-spread image might have noise, and this technique can be used to fix the issue. Finally, it may be used for any application where a convolution equation with measurable data has to be numerically solved.

6.7. Estimating an unknown point-spread function or optical transfer function

Modeling the blurring of a picture that is caused by the impacts of the equipment used for image capture is an essential part of doing quantitative analysis on photographs. When the impact of picture blur is considered to be translation invariant and isotropic, it can often be described as convolution with a radially symmetric kernel, which is referred to as the point spread function (PSF). It is not always possible to image a bright point source, which is the standard method for measuring the PSF (e.g. high energy radiography). The PSF may be estimated from a calibration picture of a vertical edge. In addition to offer a means for estimating, the strategy does so inside a hierarchical Bayesian framework that provides a measurement of uncertainty in the estimate via the use of Markov Chain Monte Carlo (MCMC) techniques.

De-blurring of out-of-focus OCT pictures using an automatically estimated point spread function (PSF). This technique deconvolutes noisy defocused pictures using a variety of Gaussian PSFs with varying beam spotsizes using the Richardson-Lucy deconvolution algorithm. Next, the information entropy of the recovered pictures is automatically used to determine the ideal beam spot size. Therefore, de-convoluting a picture does not need familiarity with the parameters or PSF of an OCT system. Light diffraction and coherent scattering by the sample are not accounted for in the model. In order to demonstrate the efficacy of the suggested approach, a number of tests have been carried out on digital phantoms, a phantom that was constructed specifically for the purpose and doped with microspheres, a fresh onion, and a human fingertip. PSF estimation and picture recovery are only two potential applications of the technology when combined with additional deconvolution methods.

6.8. Blind deconvolution

Blind deconvolution is a technique used in electrical engineering and applied mathematics, and it refers to the process of performing deconvolution without having explicit knowledge of the impulse response function that was used in the convolution. Typically, this is done by evaluating the output and making educated guesses about the input in order to predict the impulse response. Without prior knowledge of the input and impulse response, blind deconvolution cannot be solved. Most approaches for finding a solution to this issue presuppose that the input and the impulse response are in completely bounded subspaces. Despite this simplification, blind deconvolution continues to be a formidable non-convex optimization issue.

Blind deconvolution is a deconvolution method used in image processing to recover the intended picture from a single or series of blurred images where the point spread function is either poorly specified or unknown. The point spread function (PSF) is used in conventional linear and non-linear deconvolution methods. In blind deconvolution, the PSF is approximated from the input picture or series of images. Researchers have been looking into blind deconvolution techniques for decades, and they've taken many various approaches to the issue in that time.

In the early 1970s, researchers began focusing on blind deconvolution. Images taken in the night sky or in the operating room often benefit from blind deconvolution.

Non-iterative blind deconvolution uses just external information to extract the PSF in a single application of the method, whereas iterative blind deconvolution uses many applications of the technique to improve the estimate of the PSF and the scene. Maximum posteriori estimation and expectation-maximization algorithms are two examples of iterative approaches. While a precise estimate of the PSF isn't required to achieve rapid convergence, it does assist.

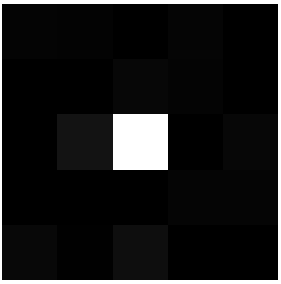
Techniques like SeDDaRA, the cepstrum transform, and APEX are all examples of non-iterative methods. Both the cepstrum transform and APEX techniques need an estimate of the PSF's width based on the assumption that the PSF has a predetermined shape. The scene data used by SeDDaRA comes in the form of a reference picture. By comparing the blurred picture's spatial frequency information to that of the target image, the method can estimate the PSF.

Examples

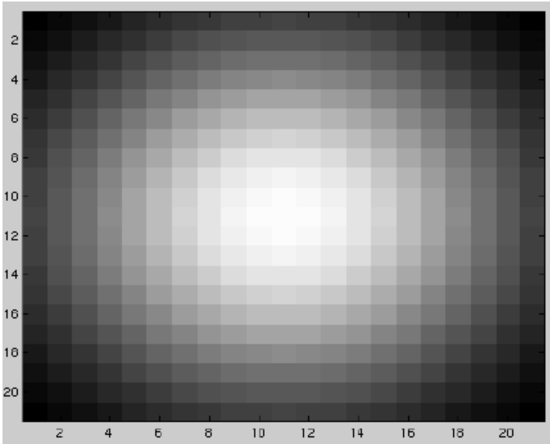
The blind deconvolution method can deblur any blurry picture if supplied as input, but the necessary conditions for its operation cannot be broken. Considering that $L > K + N$, the recovered image from the first case (the form picture) was very high quality and an identical match to the original. In the second illustration (the girl's photo), the crucial condition is broken because L < K + N. As a result, the recovered image is very different from the original.

So far, we've spoken about methods for deconvoluting a picture or performing inverse filtering based on its point spread function. Blind deconvolution, on the other hand, does not need the user to have any previous knowledge of the picture or the point spread function, and therefore it is easier to understand how it might be far more effective in real life scenarios. Let's pretend we have a degraded picture, g(x,y), which is simply image h(x,y) convolved with point spread function, f(x,y). Therefore, g(x,y)=h(x,y)*f(x,y). The deteriorated picture is all that is available to us at the outset.

When recovering an image using the Iterative Blind Deconvolution (IBD) Algorithm, first the algorithm makes an estimate of the restored picture and then an estimate of the PSF. Our implemented approach presupposes that h is a 2-dimensional impulse, as seen below:



Normal picture blurring was accomplished using a gaussian point spread function like the one illustrated here (21x21 point PSF):



The picture is subjected to a number of limitations, one of which is the constraint of a limited support. As the name implies, finite support implies that the picture ends at a finite point. This would be an acceptable estimate if we knew that the genuine image did not exist beyond this area. If the image estimate goes over this threshold, we default it to the value of the surrounding picture. We use a technique called iterative blind deconvolution (IBD) for our approach. Below is a block diagram illustrating this:

Using the Fast Fourier Transform (FFT) of the deteriorated picture and the estimation of the PSF, we can establish the first set of fourier constraints:

$$Hk(u,v) = \frac{G(u,v)\,\mathrm{conj}(\bar{F}(u,v))}{|\bar{F}(u,v)|^2 + \mathrm{alpha}/|\bar{H}(u,v)|^2}$$

The following list of Fourier restrictions includes

$$Fk(u,v) = \frac{G(u,v)\,conj(\bar{H}(u,v))}{|\bar{H}(u,v)|^2 + alpha/|\bar{F}(u,v)|^2}$$

The premise that we know, or have some understanding about, the magnitude of the PSF underlies the blur limits that are imposed. In this case, we simply disregard any information that falls beyond this range.

6.9. Iterative deconvolution and the Lucy–Richardson algorithm

6.9.1. Iterative deconvolution

The first step of iterative deconvolution is to make an educated approximation as to what the actual image is. This first educated estimate is indicated. If this assumption holds, then the convolution will provide the shown image. The residual between the observed image and the blurred estimate may be used to correct the guess if it is incorrect. In fact, sometimes all that needs to be done to make things right is to add that disparity to.

An initial estimate will be made based on the observed image. Reducing the sharpness of the observed image is the initial stage in iterative deblurring methods. This may come as a surprise, but it's valid since observational data is the most accurate image we have of the real image. If a flat field were used as the first estimate, the correction factor would equal the observed image, and the resulting estimate would also be the observed image in the second iteration.

The Basic Iterative Deconvolution (BID) process may be characterised in a more technical sense as follows:

This technique is basically the Jacobi method for solving simultaneous linear problems. It was first used to signal processing by Van Cittert (1931), and then it was expanded by Jansson (1968, 1970a, and 1970b), and finally it was independently created by Iinuma (1967a, 1967b). If an appropriate inverse filter exists, convergence occurs; otherwise, the process may be stopped after a fixed number of iterations at the closest approximation to the original image. The iterative approach may alternatively be seen as a means of calculating an identity-based power series expansion of the inverse filter.

After the first few iterations, convergence is sluggish because of decreasing returns. Additionally, the method is very vulnerable to signal noise or inaccurate PSF estimates.

The mathematical impact of the BID algorithm is best grasped in the context of the frequency domain.

6.9.2. Richardson–Lucy deconvolution

Richardson–Lucy deconvolution is an iterative process for recovering an underlying image that has been blurred by a known point spread function. This procedure is also known as the Richardson–Lucy algorithm. Richardson–Lucy deconvolution is another name for this procedure. Richardson–Lucy after the two men who separately described it.

It is impossible to create a sharp image using an optical system and then detect it using a charge-coupled device or a photograph; the point spread function describes the blurring that occurs throughout this process. Extended sources may be broken down into the sum of many individual point sources; hence, the observed image can be represented as the product of a transition matrix p that is applied to the underlying image:

$$d_i = \sum_j p_{i,j} u_j$$

Where u_j is the pixel's original image intensity on the j^{th} iteration and *di* is the pixel's detected intensity on the i^{th} iteration. In general, the amount of light from source pixel *j* that is detected in pixel *i* is described by a matrix whose members are *pi, j*.

The topic of restoring digital pictures from a measurement that has deteriorated has long been of considerable interest. The kind of degradation occurrences often determines the approach used to the challenge of image restoration. Therefore, it relies heavily on the characteristics of the background noise. The Richardson-Lucy Algorithm may be used to fix a ruined image if one has access to the noise function. This method was first described by W.H. Richardson (1972) and L.B. Lucy (1974).

6.10. Matrix formulation of image restoration

Radio Tomographic Imaging, or RTI, is a technology that has a lot of potential for use in imaging nonmetallic objects that are located inside wireless sensor networks. The usage of RTI may be seen in a wide variety of difficult settings. The image acquired by the RTI system is a degraded target image, which cannot supply sufficient information to discriminate between distinct targets due to the accuracy of the Radio Tomographic Imaging system model and interference between the wireless signals of sensors. To extract the degradation function from the shadowing-based RTI model, we will herein approach the RTI system as an image degradation process and present an estimate methodology based on a mixed Gaussian distribution. Finally, we use a technique called limited least squares filtering to utilise this degradation function to restore the original image. There have been several suggested imaging models for localization, but none of them have achieved a level of imaging accuracy that is satisfactory. Results from both simulations and experiments support the claim that our suggested strategy improves image accuracy and is practical in a wide range of real-world settings.

Imaging the attenuation of nonmetallic objects in the range of a wireless sensor network is now possible using a new method called radio tomographic imaging (RTI). The existence of targets in the path between the transmitters and receivers causes variations in the measurements of the received signal strength, abbreviated as RSS, made by the receivers. An image of the propagation field may be reconstructed by RTI using these variations. Target locations and motion information may be gleaned from the photos. As a result, RTI has been receiving a lot of attention from many fields, such as traffic monitoring, medical diagnostics, through-wall tracking, and spatial planning.

Wilson and Patwari presented the first imaging method, called shadowing-based RTI (SRTI), which makes use of RSS variation collected from a wireless network. SRTI presupposed that the wireless connections that were blocked by the targets had significant shadowing loss, while the ones that weren't blocked by the targets maintained a constant RSS. Since this assumption holds only in open areas, SRTI is unsuitable for use in buildings, where the multipath effect causes RSS to fluctuate more often. In order to enhance tracking performance in enclosed spaces, Wilson and Patwari devised Variation-Based RTI (VRTI), which included the variance of RSS. The connections were recommended to be separated into deep fade links and antifade links according to a fade level-based spatial model for RTI. The kernel distance between the RSS's short- and long-term histograms was employed in a human presence estimation image called kernel distance-based RTI (KRTI). Electronically switched directional (ESD) antennas were used in directional RTI (dRTI) systems, which reduced the impact of multipath. However, the size and cost of radio sensors will grow if directional antennas are used. Indoor RTI image quality and tracking accuracy were improved by Enhanced SRTI (ESRTI), which used the interference link cancellation approach.

The problem with target imaging in RTI, which is that we concentrate on obtaining the "original," undistorted target image rather than making progress in either the locating or tracking performance of targets. Previous studies have focused extensively on target location and tracking, leading to an expansive imaging area for the targets using RTI techniques. A high enough level of detail is lacking in the imaging result for the targets to be recognised. Deficiencies in wireless connections are to blame for this stretching of time. When the number of connections increases, a greater number of wireless sensor nodes as well as a more extensive amount of time spent scanning all communication lines will be required. Meanwhile, the inter-node interference between the sensors increases, producing subpar images. As a result, we suggest using an image restoration method to address the issue of subpar imaging in RTI. Image restoration is a technological method that makes use of past knowledge of the process by which an image degrades in order to attempt the recovery of an "original" image from a "degraded" image. To restore the "original" image, one must first get or estimate the deterioration process in order to use the inverse procedure. For this reason, we suggest a new method of obtaining clean target photos. To estimate the degradation function of the RTI system, which characterises the degradation phenomena in the RTI system, we use matrix theory and the Gaussian mixture model. In addition, in order to evaluate how well our suggested method works, we use virtual items and active humans as the benchmarks to measure its effectiveness.

6.11. Constrained least-squares restoration

Reconstructing digital photos that have been blurred due to separate motion blur is now possible with this innovative technique. The approach relies on applying the least squares solutions of certain matrix equations that describe the separable motion blur numerous times, in combination with established image deconvolution methods. The fact that the suggested algorithms can only be employed in conjunction with other image restoration methods reflects the fact that this characteristic is the most important aspect of the algorithms.

Because of the inherent faults of the imaging and capturing process, the recorded image will almost always be an inferior representation of the scene that was originally captured. This is an unavoidable aspect of the process. Images used in medicine, satellite imagery, astronomical photography, and even low-quality family photos sometimes have a blurry look. It is necessary to take into consideration a broad variety of various types of deterioration, including noise, blur, flaws in light and colour, and geometrical deterioration. Image of these flaws is essential in many image processing and analysis jobs. The original image may be reconstructed using image restoration techniques.

There has not been a lot of research done on how image processing, and image restoration in particular, may benefit by using least squares solutions. The process of blur removal from photos using least squares solutions is studied. In particular, an application of the least squares solution of minimum norm in image deblurring is being researched.

The solution that generates the least number of squares, represented takes into account both the Moore-Penrose inverse of the blurring matrix and an arbitrary matrix. The specific least squares solution, based on the Moore-Penrose inverse, was studied in The unfolding of spectroscopic and other data convolved with a window function or an instrumental impulse response may be seen as the solution of an integral equation. When data are contaminated by noise or experimental error, solving such an integral equation becomes the challenge of constructing an estimate that is a linear functional of the data and minimises the mean squared error between the correct answer and itself. The estimate is described in terms of the assumptions made about the picture and noise spectral densities.

An examination of least-squares-based restoration of image points is conducted. Point-by-point computations provide the same visual results as global Fourier-based restorations, as we demonstrate. In addition, characteristics associated with noise, point-spread functions, or object texture may be readily adjusted from pixel to pixel, giving a degree of adaptability that is only possible via computationally demanding methods of global restoration. To restore individual pixels, we need to think about a limited number of close points and the corresponding inverse matrices are computationally manageable in size. If the blurring point-spread function possesses symmetry, the sizes of these matrices may be drastically decreased.

- In order to acquire a meaningful solution to the restoration issue, it is required to have prior knowledge of the blur function h(m, n).
- Knowledge of h(m, n) is often imperfect and prone to errors.
- By basing optimality of restoration on a measure of smoothness like the image's second derivative, we may reduce the result's susceptibility to inaccuracies in h(m, n).
- The Laplacian, or second derivative, will be approximated by a matrix Q. So, we will begin by defining the limited restoration issue and finding its solution in terms of a generic matrix Q.

Suppose Q is any matrix (of appropriate dimension). In constrained image restoration, we choose $\hat{\mathbf{f}}$ to minimize $\|Q\hat{\mathbf{f}}\|^2$, subject to the constraint, $\|\mathbf{g}-\mathbf{H}\hat{\mathbf{f}}\|^2 = \|\mathbf{n}\|^2$. (Recall the degradation equation $\mathbf{g} = \mathbf{H}\hat{\mathbf{f}} + \mathbf{n} \Rightarrow \mathbf{g} - \mathbf{H}\hat{\mathbf{f}} = \mathbf{n}$.)

- The introduction of the matrix Q provides a great deal of leeway in the creation of suitable restoration filters. The formalisation of our issue is as follows:

$$\min \quad \|Q\hat{\mathbf{f}}\|^2$$
$$\text{subject to } \|\mathbf{g}-\mathbf{H}\hat{\mathbf{f}}\|^2 = \|\mathbf{n}\|^2 \text{ or } \|\mathbf{g}-\mathbf{H}\hat{\mathbf{f}}\|^2 - \|\mathbf{n}\|^2 = 0$$

Image restoration by the method of least square

To minimise the noise in speckle interferometric readings, a numerical technique is presented. This straightforward least-squares fit makes use of the collocation technique. The key characteristics are:

i) Any item may be used to create an accurate image of it.
ii) It is simple to determine the standard deviation of the estimated intensity for each meshpoint on the object.
iii) no presumption of isoplanicity is made for distances within the object's range
iv) The accuracy of the derived parameters is improved and the corresponding standard deviations are obtained directly by substituting other unknowns, such as the diameter and limb darkening coefficient for a single star or the co-ordinates and intensities of a double star, for the mesh points of the object.
v) The approach even makes use of the data from exposures that only comprise a single photon; it is anticipated that the method will reach the theoretical limit in magnitude.

vi) The huge number of numerical calculations required is a drawback of the approach.

vii) The approach makes advantage of two approximations: first, it assumes that the photon noise follows a Gaussian distribution rather than a Poisson distribution; second, it linearizes the non-linear equations describing the observation process (iteration by the Newton method).

Some test calculations used to illustrate the proposed technique reveal high resolution and signal-to-noise ratio of the generated object profile even in the presence of significant photon noise.

6.12. Stochastic input distributions and Bayesian estimators

For noisy grayscale photos, the conventional processing techniques will provide a poor denoising result under severe noise condition, leading to the loss of certain image information. A parallel array model of Fitzhugh–Nagumo (FHN) neurons has been presented. This model has the ability to successfully recover noisy grayscale pictures in situations with a low peak signal-to-noise ratio (PSNR), and it does a better job of preserving the image features. The 2D grayscale picture was first transformed into a 1D signal using the row-column scanning technique, and then the 1D signal was modulated to produce a binary pulse amplitude modulation (BPAM) signal. Modulated signal was sent into a parallel array of FHNs for stochastic resonance (SR). At last, we converted the array's output signal to a 2D grayscale picture and assessed the result using the PSNR and Structural SIMilarity (SSIM) indexes. It has been shown that the SR effect is capable of being displayed in an array of FHN neuron nonlinearities by increasing the array size. Not only does this result in an image restoration effect that is noticeably superior to that of the conventional image restoration approach, but it also enables the achievement of bigger PSNR and SSIM values. A novel approach to grayscale picture restoration in low PSNR conditions is provided.

During both the capture and transmission phases, noise may have an impact on the picture, degrading its quality. Denoising a picture results in the loss of some visual information; typical techniques of image restoration, such as filtering, concentrate primarily on suppressing and decreasing noise; however, denoising does not remove all noise. The responsiveness of nonlinear systems has been shown to be improved by the presence of internal or external noise, a phenomenon known as SR that has emerged with the rise of nonlinear dynamics.

Benzi was the first person to propose the idea of SR in order to explain the cyclical variations that may be seen in glacial periods and mild climatic periods in ancient meteorology. Since then, studies of nonlinear systems have advanced fast. While SR has found a lot of use in the area of image processing. Enhancing the MR picture using the SR neuron model requires adaptively adjusting the parameters of a bistable system, that is what image restoration is all about.

The grayscale picture then be restored using aperiodic stochastic resonance, and the SR approach can be used to the reconstruction of scattering images acquired from underneath the waves. Although these techniques work well in high PSNR conditions, they fail to provide the intended result at low PSNR settings. However, nonlinear systems are often used in the field of control engineering, which provides a platform upon which SR in nonlinear systems may grow. Amazing progress in SR has been made due to a system of nonlinearities that operates in tandem. The theory of array SR was first introduced in 1995, and the findings shown that the output signal-to-noise ratio may be increased with the use of array SR. It was discovered that the parallel bistable system could identify interference characteristic signals with a smaller input signal-to-noise ratio. We present a threshold-based parallel array model and a saturated parallel array model for the cascaded bistable system, which allows for the detection of perturbation-characteristic signals with a lower input signal-to-noise ratio. We employed array stochastic resonance to make logical stochastic resonance more stable and dependable while operating in the presence of coloured noise.

Also popular in the disciplines of chemistry, biology, and physics is the application of SR. In neuroscience, the study of the chemical and electrical characteristics of neurons is based on a model of a single neuron. Based on the more complex 4D Hodgkin–Huxley (HH) model, Fitzhugh and

Nagumo created the more straightforward and two-dimensional Fitzhugh–Nagumo model. The 2D FHN model was then simplified to offer the 1D FHN model. An increasing number of researchers are focusing their attention on SR in the nervous system, making it a highly sought after issue in the field of biological brain signal processing. Integer multiple discharge rhythms were first identified and studied by Longtin in 1991, who utilised theoretical models to mimic and examine the phenomena and draw the conclusion that the rhythm is connected to SR effects. Collins, in his analysis of the brain model of biostimulation, developed the idea of nonperiodic SR as a way to characterise the phenomena of SR in FHN. We learn that the frequency difference is crucial to the formation and control of SR in the neurological system. In the investigation of linked excitation of FHN neurons, which are able to efficiently detect subthreshold signals, the SR effect in FHN neurons was observed, and stochastic multiple resonance was discovered. For nonlinear systems with a number of inputs and outputs, an adaptive neural network command filter was developed.

For this reason, we present an SR-based FHN neuron model implemented as a parallel array for the sake of picture restoration. First, using row and column scanning, the 2D picture signal is reduced to 1D, and then using pulse amplitude modulation, the 1D signal is converted to a 1D binary aperiodic signal. The nonlinearities of the FHN array are then applied to the aperiodic 1D BPAM signal, and the resulting signal is decoded, demodulated, and restored to recover the original picture.

Stochastic Image Denoising

The challenge of eliminating noise from images, as well as the broader issue of reconstructing a signal after it has been tainted in some manner, has a long and illustrious history. There has been a lot of work done on both broad strategies and more niche applications, such as those that employ signal-specific information to fine-tune the estimating procedure. Despite the quantity of current research, there remains a gap between the release of state-of-the-art denoising algorithms, and their general acceptance outside of highly specialised applications.

There are a number of causes that have culminated at this point. Current state-of-the-art approaches are sophisticated, and either need training picture sets that offer relevant statistics about the area of application, or assume certain distributions based on actual observation. Because of this, it is challenging to build, modify, or adapt these approaches to operate with images from certain domains. In addition, there is not yet a reliable standard against which present denoising algorithms may be evaluated. As a consequence of this, there are no strong foundations upon which to make an educated choice of denoising approach for a given situation. This is a problem that arises in many fields, such as medical imaging, astronomy, photography (especially in low light settings), and the restoration of archive film.

Keeping the above in mind, we present a new picture denoising method. Our approach is theoretically straightforward, employing Monte Carlo simulation to sample a subset of all potential random walks that begin at a particular pixel, and then combining these samples using the likelihood of travelling between pairs of pixels as a weight to predict what the noise-free pixels should look like. On images from the Berkeley Segmentation Database(BSD), we compare our technique against three other methods in detail. When compared to competing algorithms, ours produces cleaner denoised output while keeping more original information. We demonstrate our algorithm's utility by applying it to images drawn from medical imaging, high-resolution digital photography, and astronomy, and we suggest ways in which our framework may be expanded.

6.12.1. Bayesian estimators

A Bayes estimator, also known as a Bayes action, is an estimator or decision rule used in estimating theory and decision theory that minimises the posterior anticipated value of a loss function (i.e., the posterior expected loss). In other words, it optimises the utility function's posterior expectation. An alternate technique of expressing an estimator inside Bayesian statistics is maximal a posteriori estimate.

The estimate of the intrinsic image information from observed images is involved in a significant variety of image and spatial information processing issues. For example, image restoration, image registration, image partitioning, depth estimation, shape reconstruction, and motion estimation are all examples of these types of challenges. These are inverse difficulties and typically ill-posed. Bayesian models, which infer the required picture information from the measured data, lend themselves well to the formulation of such estimate issues. For more than three decades, geographic data analysis has relied heavily on Bayesian concepts, which have found several useful applications.

An estimate of an unknown parameter θ that minimises the anticipated loss for all observations x of X is what's known as a Bayesian estimator.

What this means is that the term is an estimate for the unknown parameter that sacrifices the least possible precision.

The Bayesian approach to image analysis is broken down into its component, beginning with its fundamental principles. Using previous information about the scenario being studied is an advantage of the Bayesian technique in image processing and interpretation. A variety of examples are used to explain the underlying notions. These examples range from a problem in one dimension to a problem in two dimensions to big challenges in picture reconstruction that make use of complex previous knowledge.

The Bayesian method allows for the use of previous information throughout the data analysis process. The posterior probability is the essential concept in Bayesian analysis since it captures the overall level of confidence in a particular conclusion. The posterior probability is calculated using Bayes' rule, which stipulates that it is equal to the product of the likelihood and the prior probability. The probability takes into account all of the information provided by the most recent data. Before any data are collected, the prior reflects how confident one is in their understanding of the issue.

Despite the fact that the posterior probability gives a full account of the degree of confidence associated with every given picture, it is frequently required to choose a single image as the result or reconstruction. It is common practise to choose the picture that maximises the MAP estimate of the posterior probability. Other options for the estimator, such as the mean of the posterior density function, can be preferable in some circumstances.

The data may not be adequate to provide a unique solution to the issue in cases when only extremely little data is available. When using the Bayesian approach, the prior supplied may steer the final outcome in the desired direction. The prior is the only factor that differentiates the maximal a posteriori (MAP) solution from the maximum likelihood (ML) solution; thus, selecting the prior is one of the most important components in Bayesian analysis.

Jaynes is widely recognised for revitalising the Bayesian method of analysis. Bayesian methodology relies on probability theory, which allows for the ranking of alternatives according to their relative likelihood or preference, and the performance of inference in a consistent manner.

6.13. The generalized Gauss–Markov estimator

If your linear regression model meets the first six classical assumptions, the Gauss-Markov theorem asserts that ordinary least squares (OLS) regression will provide unbiased estimates with the minimum variance of all conceivable linear estimators.

The demonstration of this theorem's proof is much beyond the scope. However, if you make sure that you're getting the greatest possible coefficient estimates by sticking to the classical assumptions, then everything else doesn't matter. The Gauss-Markov theorem does not specifically indicate that these are the best possible estimates alone for the OLS technique; rather, it states that they are the best possible values for any linear model estimator.

In the field of statistics, the Gauss–Markov theorem (or simply the Gauss theorem) says that ordinary least squares (OLS) estimator has the lowest sampling variance within the class of linear unbiased estimators. This is the case if the errors in the linear regression model are uncorrelated, have equal variances, and an expectation value of zero. There is no need for the mistakes to be typical or for them to be randomly distributed (only uncorrelated with mean zero and homoscedastic with finite variance). Since there are biassed estimators that have smaller variance, the criterion that the estimator be unbiased cannot be abandoned. Examples include the ridge regression method, any degenerate estimator, and the James-Stein estimator.

The theorem was called after Carl Friedrich Gauss and Andrey Markov, despite the fact that Gauss' work was completed a considerable amount of time before Markov's. But although Gauss obtained the conclusion under the premise of independence and normalcy. Alexander Aitken extended this concept to non-spherical inaccuracies.