
CHAPTER
ONE

Introduction

Terry S. Yoo
National Library of Medicine, NIH

This book is an introduction to the theory of modern medical image processing. Image processing is the manipulation and analysis of image-based digital data to enhance and illuminate information within the data stream. One common approach to this topic developed by many texts is to treat the matter as a high-dimensional form of signal processing, concentrating on filtering and frequency analysis as the foundations with later chapters discussing applications and advanced algorithms. This type of organization reflects a bias toward electrical engineering, viewing images as similar to complicated audio or radiofrequency signals. Another approach is to start from the mathematical perspective, treating the issue as a series of exercises in statistics or applied mathematics through equations and theorems. Our approach is to explore the topic through implementations of advanced algorithms, providing the mathematical, statistical, or signal processing as needed for background.

This book takes a software engineer's viewpoint, and describes methods and algorithms available in the Insight Toolkit as open source software. The goal of this book is to explain the background specific to our implementations, enabling the reader or student to use complex tools in medical image processing as rapidly as possible. We do not claim an exhaustive treatment of these ideas, but rather a working and functional introduction to these topics, partnered with working implementations of the methods described.

1.1 Medical Image Processing

The medical mission differs from the other forms of image processing arising from non-medical data. In satellite surveillance analysis, the purpose is largely a screening and cartographic task, aligning multiple types of data and corresponding them to a known map and highlighting possible points of interest. In computer vision, camera views must be analyzed accounting for the perspective geometry and photogrammetric distortions associated with the optical systems that are the basis for robotic sensors. In many of these systems, autonomous navigation, target identification and acquisition, and threat avoidance are the primary tasks. For the most part, the incoming information arrives as 2D images, data arrays that can be organized using two cartesian dimensions. In addition, the tasks are to be performed independently by the machine, relying on the development of machine learning and artificial intelligence algorithms to automatically accomplish the tasks.

In medicine, the problem as well as the input data stream are usually three-dimensional, and the effort to solve the primary tasks is often a partnership of human and machine. Medicine is notably a human enterprise, and computers are merely assistants, not surrogates nor possible replacements for the human expert. The medical task can often be split into three areas: (1) data operations of filtering, noise removal, and contrast and feature enhancement, (2) detection of medical conditions or events, and (3) quantitative analysis of the lesion or detected event. Of these subtasks, detection of lesions or other pathologies is often a subjective and qualitative decision, a type of process ill-suited for execution by a computer. By contrast, the computer is vastly more capable of both quantitative measurement of the medical condition (such as tumor volume or the length of a bone fracture) and the preprocessing tasks of filtering, sharpening, and focusing image detail. The natural partnership of humans and machines in medicine is to provide the clinician with powerful tools for image analysis and measurement, while relying on the magnificent capabilities of the human visual system to detect and screen for the primary findings.

We divide the problems inherent in medical image processing into three basic categories:

- **Filtering:** These are the basic tasks involved in filtering and preprocessing the data before detection and analysis are performed either by the machine or the human operator.
- **Segmentation:** This is the task of partitioning an image (2D array or volume) into contiguous regions with cohesive properties.
- **Registration:** This is the task of aligning multiple data streams or images, permitting the fusion of different information creating a more powerful diagnostic tool than any single image alone.

These three basic divisions represent the organization of both this book and the Insight Toolkit for which this book was written. In order to best understand the approaches and difficulties associated with these tasks, we begin with a history of medical imaging and a brief overview of modern imaging modalities.

1.2 A Brief Retrospective on 3D Medical Imaging

In the broad realm of the sciences, medical imaging is a young field. The physics enabling diagnostic imaging are barely one hundred years old. In 1895, Wilhelm Roentgen discovered x-rays while experimenting with a Crookes tube, the precursor to the cathode ray tube common in video applications today. It is worth noting that he immediately recognized the potential of x-ray radiation in diagnostic imaging and that one of his earliest images is of the bones of his wife's hand. Roentgen's discovery was so noteworthy and revolutionary that he received a Nobel prize within six years of announcing his initial work. Thus began a bountiful and tightly-coupled one hundred year partnership between physics and medicine.

The meaning and purpose of medical imaging has been to provide clinicians with the ability to see inside the body, to diagnose the human condition. The primary focus for much of this development has been to improve the quality of the images for humans to evaluate. Only recently has computer technology become sufficiently sophisticated to assist in the process of diagnosis. There is a natural partnership between computer science and radiology.

Early in the twentieth century, a Czech mathematician named Johann Radon derived a transform for reconstructing cross-sectional information from a series of planar projections taken from around an object. While this powerful theory had been known for over fifty years, the ability to compute the transform on real data was not possible until digital computers began to mature in the 1970s.

Imaging in 3D emerged in 1972 when x-ray computed tomography (CT) was developed independently by Godfrey Hounsfield and Alan Cormack. These innovators later shared the 1979 Nobel Prize in Medicine. Their achievement is noteworthy because it is largely based on engineering, the theoretical mathematics and the underlying science had been described decades earlier. The contribution is in the application of mechanical engineering and computer science to complete what had previously only been conceived on paper. Clinical systems were patented in 1975 and began service immediately thereafter.

While techniques for using x-rays in medical imaging were being refined, organic chemists had been exploring the uses of nuclear magnetic resonance (NMR) to analyze chemical samples. Felix Bloch and Edward Purcell were studying NMR in the mid 1940s. Together, they shared the Nobel Prize in Physics in 1952. Paul Lauterbur, Peter Mansfield, and Raymond Damadian were the first to develop imaging applications from NMR phenomena. Lauterbur created tomo-

graphic images of a physical phantom constructed of capillary tubes and water in a modified spectrometer in 1972. Damadian later was able to create animal images in 1975. The means of creating medical images using magnetic fields and radio waves was later renamed Magnetic Resonance Imaging (MRI). Lauterbur and Mansfield received the 2003 Nobel Prize in Medicine. Raymond Damadian has not been so honored, but his contribution is significant and should not be overlooked by us.

Radiologists were thus simultaneously presented with the possibilities for slice (or tomographic) images in the axial plane from x-ray CT and from MRI in the early 1970s. Arising from the long development of x-ray technology, CT matured first. The difficulties and expense of generating strong magnetic fields remained an obstacle for clinical MRI scanners until engineers were able to produce practical superconducting magnets. While still expensive, MRI technology is now available worldwide, creating precise images of deep structures within the body, outlining the anatomy and physiology of internal objects other than bones. MRI and CT have joined the established 3D imaging modalities of nuclear imaging and the growing field of volumetric medical ultrasound. The creation of new modalities with differing strengths has led to the need to align or register these multiple data streams.

1.3 Medical Imaging Technology

The majority of medical visualization involves the display of data acquired directly from a patient. The radiologist is trained through long years of education and practice to read relatively simple presentations of the raw data. The key to improving a diagnosis is in the careful crafting of the acquisition, applying the physics of radiology to maximize the contrast among the relevant tissues and suppressing noise, fog, and scatter that may obscure the objects of interest. This is no less true for more complex visualizations that incorporate 3D renderings or volume projections; improving the quality of the acquired data will fundamentally affect the quality of the resulting visualization.

This section will briefly cover some of the more common sources of three or higher-dimensional medical imaging data. This treatment is necessarily superficial, serving mostly as an annotated glossary for many of the terms taken for granted in the radiology community. To create effective medical visualization tools, the computer scientist requires a fundamental understanding of the source of the image data, the technology involved, and the physical principles from which the image values are derived. This cursory introduction will be insufficient for in-depth research, but will serve as background for this text; the reader is encouraged to continue the exploration of this topic. This additional command of the basics of medical image acquisition will enhance your practice in medical visualization,

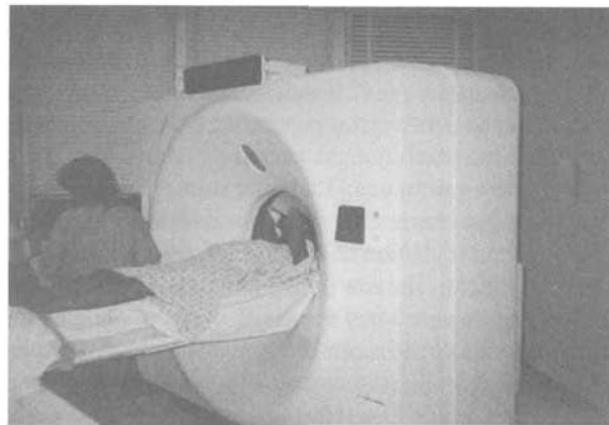


Figure 1.1. The business end of a CT scanner (circa 1998).

ease communication between you and the doctor and the technologist, improve the quality of the source data, and smooth transitions among the many interfaces from acquisition to display.

1.3.1 Computed Tomography

By far the most familiar form of 3D medical imaging is X-ray Computed Tomography or CT (formerly referred to as Computer Assisted Tomography, Computerized Axial Tomography, CAT scanning, Computerized Transaxial Tomography, CTAT, Computerized Reconstruction Tomography, CRT, and Digital Axial Tomography, DAT). The mathematics for tomographic reconstruction from multiple views have been known for most of this century. It took almost fifty years before the components required for x-ray computed tomography were sufficiently developed to make the procedure and the ensemble of instruments economically feasible.

A CT scanner is a room sized x-ray instrument, requiring a shielded environment to protect the technologists and other clinic staff from exposure from routine use. The number of manufacturers of CT scanners has been steadily decreasing with only a handful of vendors providing them today. The cost of these devices ranges from approximately \$400,000 to well over \$1,000,000. There are some examples of portable CT machines that can be moved into a trauma care center or into operating rooms to aid in the diagnosis and treatment of patients. Emerging technologies in flat-panel digital x-ray devices is enabling 3D imaging using more conventional fluoroscopic tools, but at the time of this writing, the following is the most common configuration for these devices.

An accurately calibrated moving bed to translate the patient through the scanner, an x-ray tube mounted in such a way to allow it to revolve about the patient, and an array of x-ray detectors (gas filled detectors or crystal scintillation detectors) comprise the essential system components of a CT machine. The x-ray tube and detector array are mounted in a gantry that positions the detector assembly directly across from the x-ray source. The x-ray source is collimated by a pair of lead jaws so that the x-rays form a flat fan beam with a thickness determined by the operator. During the acquisition of a "slice" of data, the source-detector ring is rotated around the patient. The raw output from the detector array is backprojected to reconstruct a cross-sectional transaxial image of the patient. By repositioning the patient, a series of slices can be aggregated into a 3D representation of the patient's anatomy. Figure 1.1 is a picture of a CT table and gantry.

Within the last fifteen years there have been significant advances in CT technology, allowing for faster spiral acquisition and reduced dose to the patient as well as multislice detector arrays permitting simultaneous acquisition of several slices at a time. Early CT gantries were constructed with a revolving detector array positioned directly across the patient from the moving x-ray source. Cable lengths connecting the moving detector assembly and the x-ray tube permitted only a single slice to be acquired at one time. The revolving assembly then had to be "unwound", the patient advanced the distance of one slice, and the process repeated.

As small affordable detectors have become available, scanners have been designed with a fixed array of x-ray detectors. The only remaining revolving part of the gantry is the x-ray tube. This has simplified the engineering and cabling of the gantry. The x-ray tube can now be cabled using a slip ring, permitting continuous revolution of the tube about the patient. This type of design is capable of helical (or spiral) CT acquisition. By simultaneously revolving the x-ray source about the patient and continuously moving the patient through the bore of the gantry, the data are acquired via a spiral path. These methods have enabled very fast image acquisition, improving the patient throughput in a CT facility, reducing artifacts from patient motion, and reducing absorbed dose by the patient. Combined with multiple layers of sensors in the detection ring, these new scanners are generating datasets of increasingly larger sizes.

A CT scanner is an x-ray modality and is subject to all of the physics associated with the generation, dispersion, absorption, and scatter associated with all x-ray photons. The process of generating a CT scan is similar to creating a standard x-ray film; however, while a single x-ray exposure generates a complete film-based exam, the CT image is not acquired in a complete form. Rather, it must be reconstructed from multiple views. The advantage of being an x-ray modality is that laymen and clinicians alike have considerable intuition when dealing with x-ray-based images. The concepts of dense objects like bone absorbing more photons relative to less dense tissues like muscle or fat come naturally from our

experience and expectations about x-ray imaging. A typical CT scanner can generally acquire the data for a transaxial slice in a matter of seconds (within 1 to 5 seconds). An exam can include several series of slices, in some cases with and without pharmaceutical contrast agents injected into the patient to aid in diagnostic reading. Slices can be spaced such that they are either overlapping or contiguous, though some protocols call for gaps between the slices. A large study can include well over 100 separate 512×512 pixel images. The radiation dose from a CT scan is comparable with that of a series of traditional x-rays.

The concept of “resolution” should be divided in the researcher’s mind into spatial resolution (i.e., how much area in each dimension a voxel covers) and sampling resolution (how many voxels in each dimension of the slice). Sampling resolution in current scanners usually creates images that are either 256×256 or 512×512 voxels square. Sampling resolution in the longitudinal direction is limited only by the number of slices acquired. Spatial resolution in the longitudinal direction is bound by physical limitations of collimating the photons into thin planes. The physical lower limit is approximately 1 mm; narrower collimation requires inordinate amounts of x-ray flux to image and also leads to diffraction interference. Sampling resolution in the transaxial dimensions of a voxel is based on the field of view selected by the operator of the CT scanner and the matrix (256×256 or 512×512) yielding pixel dimensions that are generally 0.5 to 2 mm. Attempting to achieve higher spatial resolution will lead to voxels with too little signal to accurately measure x-ray absorption. With today’s diagnostic equipment, if pixels smaller than 0.25 mm are attempted, low signal-to-noise ratios become a problem.

Units of measure for the pixel values of CT imaging are standard across the industry. Each pixel ideally represents the absorption characteristics of the small volume within its bounds. By convention, these measurements are normalized relative to the x-ray absorption characteristics of water. This is a unit of measure known as Hounsfield units (HU). The Hounsfield unit scale is calibrated upon the attenuation coefficient for water, with water reading 0 HU. On this scale air is -1000 HU, fat tissue will be in the range of -300 to -100 HU, muscle tissue 10–70 HU and bone above 200 HU.

Every CT image contains artifacts that should be understood and handled when visualizing the data. The process of reconstruction and sampling leads to aliasing artifacts, just as any sampling procedure in computer graphics or signal processing. Another common artifact is partial voluming, the condition where the contents of a pixel are distributed across multiple tissue types, blending the absorption characteristics of different materials. Patient motion while scanning generates a variety of blurring and ring artifacts during reconstruction.

There is another class of artifacts that arises from the fact that CT is an x-ray modality. Embedded dense objects such as dental fixtures and fillings or bullets lead to beam shadows and streak artifacts. More common is the partial filtering

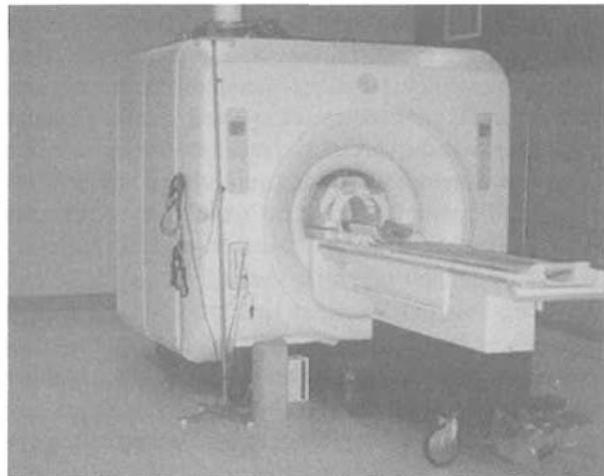


Figure 1.2. The business end of an MRI scanner (circa 1998).

of the x-ray beam by dense tissue such as bone which leads to beam hardening, a condition which causes slight shadows to halo the dense features of the image. When creating visualizations, beam hardening artifacts may cause the researcher to underestimate the volume of dense objects, making them seem smaller than they are.

1.3.2 Magnetic Resonance Imaging

Like CT, Magnetic Resonance Imaging or MRI (formerly referred to as Nuclear Magnetic Resonance imaging) was pioneered in the early 1970s. Introduced under the name “Zeugmatography” (from the Greek word *zeugma*, meaning “that which draws together”) it remained an experimental technology for many years. Unlike CT, MRI does not use ionizing radiation to generate cross-sectional images. MRI is considered to be a newer modality since feasible commercial development of diagnostic MRI scanners had to await affordable super-conducting magnets. Today’s diagnostic systems ranges from \$500,000 to \$2 or \$3 million.

An MRI scanner is a large magnet, a microwave transmitter, a microwave antenna, and several electronic components that decode the signal and reconstruct cross-sectional images from the data. Generally, the magnet is superconducting and must be operated at cryogenic temperatures (4 degrees Kelvin), necessitating its immersion in a bath of liquid helium. Some MRI scanners with relatively poorer resolution are being constructed using fixed magnets without the need for a liquid helium container, permitting more open designs at the cost of image con-

trast and resolution. Figure 1.2 is a picture of an MRI magnet and table, showing a head coil, an antenna for imaging the head and neck.

Unlike CT scanners, the bore of an MRI scanner is often up to two meters in length (6–8 feet). Patients are inserted into the middle of the magnetic field, often inducing claustrophobia. The environments for MRI scanners must be shielded for magnetic and radiofrequency interference. Large Faraday cages and substantial masses of iron usually surround the magnet and sometimes the entire room about the scanner.

In MRI scanning, the patient is placed within a high intensity magnetic field. Field strengths vary from 0.35 Tesla to 1.5 Tesla for most diagnostic MRI devices (for reference, 1 Tesla = 10,000 Gauss, and the earth's magnetic field, though variable, is approximate 0.5 Gauss). The induced magnetic field causes the magnetic moments of the hydrogen atoms within the patient to align along the principal direction of the superconducting magnet. Low-level radio waves in the microwave frequencies (approximately 15 to 60 MHz) are then transmitted through the patient, causing the magnetic moments of the hydrogen nuclei to resonate and re-emit microwaves after each pulse. The microwaves emitted by the body are recorded using a radio frequency antenna, filtered, amplified, and reconstructed into tomographic slices. While all of the hydrogen nuclei typically resonate at a frequency fixed by the strength of the induced magnetic field, different tissue types resonate longer than others, allowing the viewer to discriminate among them based on the magnitude of the signal from different points in space over time.

Spatial locations can be determined by varying the magnetic field about the patient in different directions at different times. Linear gradients in the induced magnetic field are produced by magnets that supplement the main superconducting magnet. These gradient coils are activated in a variety of sequences, altering the phase and frequency of the microwave pulses that are received and re-emitted by the hydrogen nuclei. The design and crafting of pulse sequences is a field of research unto itself. The careful selection of pulse sequences can illuminate a variety of clinical conditions including the display of swelling and bleeding, even enhancing the blood vessels deep within the brain. Often, several sequences will be taken of the patient during an exam to capture the variety of information available in the different pulse sequences. The visualization researcher is encouraged to learn about these pulse sequences and their properties in order to select the best ensemble of them to use when crafting visualizations.

The output of an MRI scanner is similar to CT. Slices representing slabs of the object scanned are produced. However, unlike CT which always produces transaxial slices, the slices from MRI can be oriented in any plane. The output values at each image element are not calibrated to any particular scale. Generally they are 10-bit data samples. The values will vary depending upon the scan parameters, and the patient's size and magnetic characteristics. Additionally, the values are not constant over the entire scan space since inhomogeneity

in the magnetic field causes pixels that may represent the same tissue, but located some distance apart to give different signals. This lack of an absolute scale for a dataset is a cause of much consternation to the researcher attempting to segment MRI data.

Visualization researchers seldom have good intuition for the meaning of MRI signals. Unlike x-ray-based imaging, there are no physical analogs to what the viewer is seeing. An MRI device measures the radio signals emitted by drops of water over time. Usually skin and fat are much brighter than bone which has virtually no signal at all. Segmentation and classification are therefore significantly harder and are the subject of much research.

As with CT, the concept of resolution must be divided into spatial resolution and sampling resolution. The matrix of the image can typically be selected to be either 256×256 or 512×512 , depending on the sampling resolution of the frequency and phase decoders of the receiving equipment. Square pixels in a rectangular matrix are often selected since reducing the number of rows or columns can significantly reduce the time required to complete a scan. Spatial resolution, or field of view, is primarily dependent on the strength of the gradient magnets and their ability to separate the slices along their gradient directions. The radiofrequency receiving equipment must be able to distinguish among frequencies and phases which have only slight variations. Stronger gradients create greater separations and improve spatial resolution.

Remember that the MR scanner is attempting to measure the radio signal resonating from a drop of water that is as small as $1 \text{ mm} \times 1 \text{ mm} \times 2 \text{ mm}$. There are significant trades to be made when attempting to increase spatial resolution. Signal to noise will degrade as electronic distortion of the antenna coil and the amplifiers begin to overcome the small signals involved in diagnostic imaging. Imaging with larger voxels, thicker slices, or repeating and averaging multiple acquisitions are solutions. The relative consequences are increased aliasing, partial voluming, and possible patient motion artifact.

MRI data is subject to several artifacts. The issues of partial voluming, patient motion, and aliasing are common with CT. While MRI does not have x-ray-related artifacts, it has its own host of radio and magnetic artifacts of which the visualization expert should be aware. Like any microwave used for cooking, the mass of the patient (or food) will affect how well the material absorbs the radiofrequency energies. This leads to “cold” spots in the imaging volume. In addition, the distribution of the antenna coverage for both transmission and reception and inhomogeneity in the induced magnetic field lead to inconsistent quantitative values and even inaccurate geometry within an image. There is significant research being conducted addressing methods to correct for these distortions.

Most MRI scanners can acquire a set of slices (30 to 50) within five to ten minutes. An entire study of a patient generally represents two to three series of slices, with a study time of 30 to 45 minutes. Each slice generally represents a

thickness of 2 to 10 mm and contains 256×256 pixels. As with CT, the pixel dimensions are set by the image matrix and the field of view parameters.

New work in MR imaging is leading to novel capabilities in this modality. Studies in perfusion and diffusion of various agents across the body are being enabled by new capabilities in fast imaging. These studies can illuminate blood flow, creating high-resolution images of vascular structure. Functional MRI has been introduced not only to record the patient's anatomy, but also the physiological functions of the tissues being studied, largely used today to map the cerebral cortex of the brain. The complex pulse sequences of Diffusion Tensor Imaging (or DTI) are beginning to reveal the direction and course of nerve bundles deep within the white matter of the brain, providing new research areas of tractography and patient specific anatomical analysis. The pace of research in MRI is increasing as new capabilities are enabled.

1.3.3 Nuclear Medicine

Imaging using pharmaceuticals tagged with radioactive agents is an older technology than either CT or MRI. In nuclear medicine imaging, a radioactive source is injected into the patient and the distribution of that source is measured using a detector array to catch and quantify the radiation emitted from the body. The detector array is sometimes configured similar to that for CT, while other systems utilize a two-dimensional array of detectors. The major difference between CT and this type of study, is that with CT the radiation source's location is external and known, while in nuclear medicine, the source is internal and its distribution unknown. Figure 1.3 is a picture of a two-headed Gamma Camera (Anger Camera) used for Single Photon Emission Computed Tomography (SPECT). The gantry has two detector arrays for gamma radiation emitted from the patient. The detectors revolve about the patient, and tomographic images are reconstructed.

While the images generated from measuring radioactive decay are often blurry with poor spatial resolution, nuclear medicine allows clinicians to image the physiological activity (the biological activity) of the patient, rather than just the geometry of the patient's anatomy. The selection of the pharmaceutical to which the radionuclie is attached will determine the specificity and performance of the imaging study. For instance, radioactive iodine generates very specific studies of the thyroid and parathyroid glands. Radioactive tracers can be attached to hundreds of different substances.

Nuclear medicine studies produce images with fairly low resolution and a high amount of noise. This is due to our inability to use high radiation doses because of the consequences of high doses of radiation to the patient. Choosing persistent radiopharmaceuticals will cause the agent to reside longer within the body, improving our capacity to resolve interal objects while potentially inflicting additional damage on the patient.

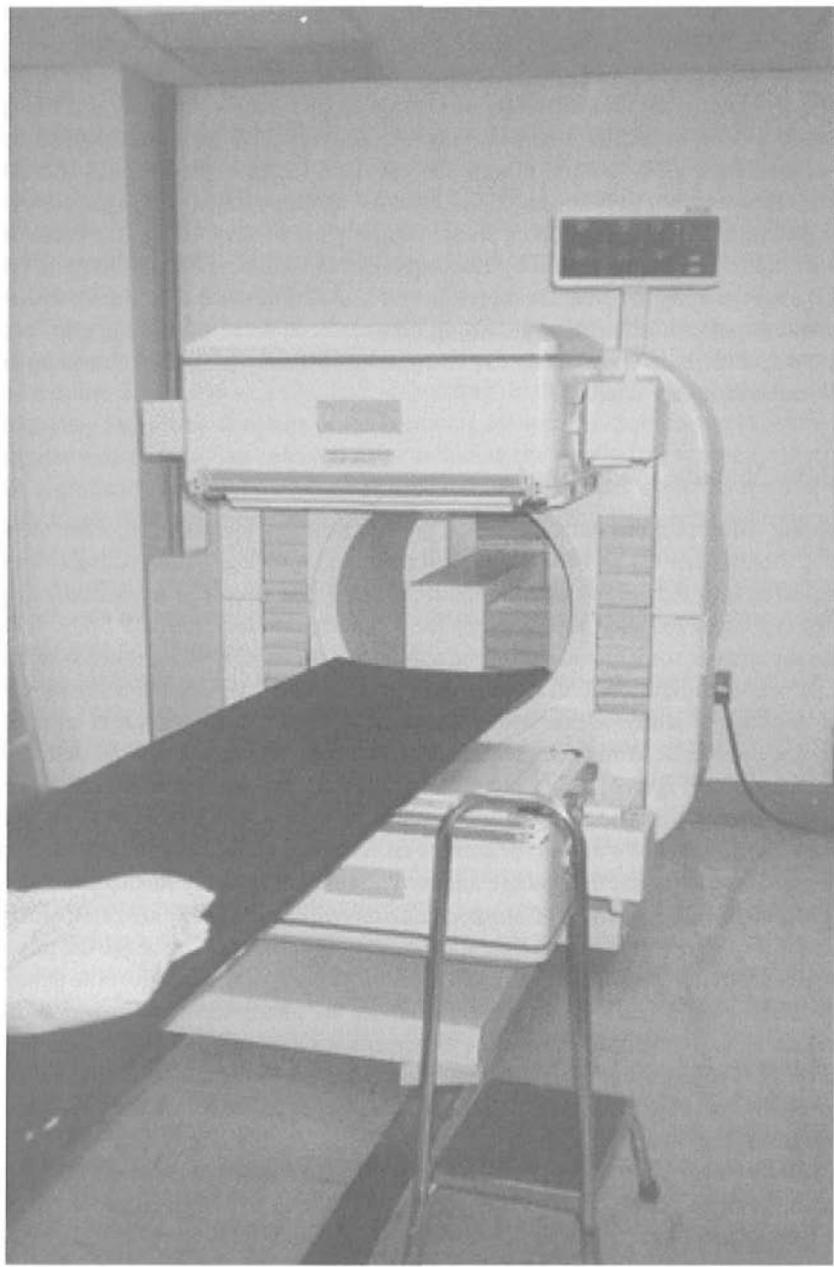


Figure 1.3. A two-headed Gamma Camera used in nuclear medicine (circa 1998).

Most nuclear medicine is still based on the 2D planar representations generated by simply recording the activity of a detector over a period of time. If the detectors are rotated about the patient, reconstruction is possible. The two most common types of 3D nuclear medicine studies are Single Photon Emission Computed Tomography (SPECT) and Positron Emission Tomography (PET). SPECT studies utilize radiotracers that emit photons while decaying. The radioactive agents used in SPECT have half-lives measured in hours and can be easily generated or transported to the clinic. PET studies use radioactive isotopes that decay by positron emission. The resulting positrons annihilate when they encounter electrons generating two high energy photons that conserve momentum by traveling in opposite directions. The half-lives of the isotopes used in PET are often measured in minutes, necessitating their production at the clinic. A particle accelerator such as a cyclotron is required to create most of the radionuclei for PET imaging, increasing the cost of a PET installation.

The output of a SPECT or PET scanner is typically a set of 10 to 30 transaxial slices. Each slice is typically 5 to 10 mm thick and contains pixels 5 to 10 mm in size. There may be gaps between the slices. Often multiple sets of scans are taken five to fifteen minutes apart allowing for time-resolved studies. Nuclear medicine studies produce images which have few anatomic cues. The images mimic physiologic activity which may or may not be easily transferable to the underlying anatomy.

1.3.4 Ultrasound

Unlike the three other modalities described in this chapter, Ultrasonography uses classical physics to perform its imaging rather than the more esoteric phenomena of nuclear decay or x-ray quantum mechanics. No photons are used. A piezoelectric quartz crystal, similar to those used to clock CPUs, wristwatches, or to start electronic ignition lighters for barbecue grills, is used to create high frequency acoustic energy (3 to 10 Megahertz) which is then reflected off surfaces and interfaces between organs deep within the body.

The same device that is used to create the acoustic signal, a transducer, is then used to measure the returning echo information. Partial reflections are created by interfaces between areas of differing acoustical impedance. The result is imaging very similar to the SONAR systems used in maritime and undersea naval imaging. The sonographer places the probe against the patient and moves it to obtain images of various parts of the body. The result is usually a 2D representation of the area under the transducer. Most ultrasound machines consist of a linear array of transducers, and produce an image representing a pie-shaped slice of the body. One of the advantages of ultrasound is that it produces images in real time. Another advantage is the absence of ionizing radiation.

Ultrasound machines are fairly inexpensive compared with the other diagnostic 3D modalities in common use in the health care industry. A high-end diagnostic ultrasound system can be purchased for approximately \$250,000, while functional, more moderately priced units can be acquired for less than \$100,000 today.

3D images can easily be created by extruding several slices through space. Most commercial ultrasound equipment allows for 3D imaging; however, accurate spatial information is seldom provided. Spatial tracking of the transducer array is imperative for clinical 3D visualization. Some approaches to correcting this defect have been to use physical, optical, and electronic devices for locating the transducer in space. Providing a rigid path for the transducer creates some compelling results. Other techniques involve rotating the transducer assembly, sweeping a volume in a cylindrical section similar to aeronautical RADAR imaging. Finally, recent advances in transducer design have yielded demonstrable 3D transducers that can acquire volume images from ultrasound with no mechanically moving parts.

However, once the position and orientation of the probe are known, the data are still often sampled in irregular intervals, and defy many image processing techniques. Ultrasound images typically contain a large amount of noise termed speckle, that adds to the problem of identifying structures. Object detection in volume ultrasound is the subject of much advanced medical image processing research.

1.4 Acquisition, Analysis, and Visualization

Most acquisition of medical data is not optimized for 3D presentation. Indeed, to improve the signal to noise, thick slices are usually requested. The result is better contrast at the cost of higher partial-voluming artifact and poor spatial resolution in the effective *z*-direction. In order to reduce the dose to the patient and the time required to perform a scan, slices are not always contiguous. Rather small gaps are introduced between slices. Since most presentation is 2D, these practices seldom affect the clinician; however, they can be fatal to 3D visualization.

Radiologists often must balance the need for image quality with the interests of the patient. Many times the means to improving the diagnostic power of an image will mean increased dose to the patient. The factors affecting the trade-off between image quality and acceptable dose include a wide variety of concerns. For example, since the inadvertent long range biological effects (chromosomal damage) of ionizing radiation are most profound in children, pediatric radiology seldom trades increased dose for improved contrast. Similar trades are made throughout medical imaging (e.g., trading spatial resolution or increased imaging time in MRI for an improved signal-to-noise ratio). The computer scientist inter-

ested in medical visualization would be well served to learn where these trades are being made and learn to cope with their consequences.

An even deeper understanding of the clinical findings of a case is required to validate a 3D visualization. Knowledge of what the physician is seeking in the patient will help to direct the acquisition so that the results not only capture the desired pathology in high detail, but also assure the computer scientist that the data are in an appropriate form for advanced rendering.

In his 1993 presentation on 3D Medical Visualization, Derek Ney, an assistant professor for the Johns Hopkins Department of Radiology, wrote:

A succinct statement of the relationship of visualization and acquisition is that the single easiest method for improving visualization is to use better (higher resolution, higher contrast, higher signal to noise, etc.) acquisition techniques. This implies that the method for acquisition should be as optimal as possible. There is no room in most systems for poor acquisition technique. The researcher in visualization is warned to learn about the potential pitfalls in the acquisition stage, so that they are presented with data that was acquired with the best technique possible.

Indeed, as with all data processing, “if garbage goes in, only garbage comes out.” Or more precisely, if one beautifully renders garbage data, then one only has a picture of well-dressed garbage. The result serves no one in particular and wastes valuable resources that would be better used to serve the patient, to serve the community, and to serve future generations of researchers and patients alike.

1.5 Summary

Keeping the basic concepts of medical image acquisition in mind, we turn our attention to the processing and analysis of the generated data. The multiplication of imaging modalities with differing strengths and weaknesses in imaging anatomy v. physiology and hard tissue v. soft tissue requires careful registration and alignment of multimodal data, and differences in size and resolution lead to multiscale methods as well. We refrain from covering the algorithms and methods for generating visualizations of the data, instead concentrating on the analysis and refinement of the data stream in preparation for the ultimate creation of rendered images of the patient anatomy and physiology.

This book is divided into three main parts:

- **Basics:** These chapters describe techniques for preprocessing the image data before complex semantic, knowledge-based operations are performed either by the machine or the human operator. Statistical, linear, and more

advanced nonlinear methods are covered, matching implementations from the software.

- **Segmentation:** As stated before, this is the task of partitioning an image (2D array or volume) into contiguous regions with cohesive properties. A variety of approaches are given including hybrid mixtures of statistical and structural methods. We cover the use of finite element models, Voronoi decomposition, fuzzy-connectedness, and other techniques useful for segmentation implemented within ITK.
- **Registration:** This is the task of aligning multiple data streams or images, permitting the fusion of different information creating a more powerful diagnostic tool than any single image alone. We cover basic registration strategies as well as more complex deformable techniques, all implemented under the umbrella of the ITK registration framework.

This book is intended as a companion to the Insight Toolkit covering the practices and principles used to develop the corresponding software. Implementation and integration details are covered elsewhere in the ITK Software Guide. The ultimate arbiter of all of this material is the source code of ITK itself, made available, by intent and design, in the public domain. We hope that this text serves as a summary to guide and explain the many choices made and roads taken in the design and use of the software.

Enjoy ITK.

Basic Image Processing and Linear Operators

Terry S. Yoo
National Library of Medicine, NIH

George D. Stetten
University of Pittsburgh

Bill Lorensen
GE Global Research

2.1 Introduction

The topics discussed throughout this book fall into the realm of what is usually called *image analysis*. Before researchers adopted that term, they often referred to their field as *image processing*, an extension of traditional signal processing to images. An image can be thought of as a signal in space rather than time, arrayed in two or three dimensions rather than one. This leads to fundamental differences between conventional signal processing and image processing. While one generally only moves forward through time, in space, one moves left or right with equal ease, as well as up or down, forwards or backwards. The extension to multiple dimensions raises new issues, such as rotation and other interactions between the axes.

As with conventional signals, images can be either continuous or discrete. Optical images, consisting of light, call for analysis in the continuous domain, as they pass through lenses, for example. In contrast, images found on a computer

generally consist of individual samples on a regular grid, and therefore require discrete analysis. Digital sampling theory is intrinsic to image processing. Familiar artifacts such as aliasing can be understood through digital sampling and filtering theory. Likewise, effective reconstruction, the inverse of sampling, requires a clear understanding of digital sampling theory.

Both continuous and discrete image analysis often begin with linear operators adapted from signal processing. These operators range from simple algebra to convolution and the Fourier transform. Linear operators assume that the light or other input adds by superposition, i.e., that individual components of the input do not affect each other within the system. Linear operators preserve the independence of basis functions useful in extracting information from an image. Some linear operators affect each sample in the image, or *pixel*, independently from its neighbors, while others take neighboring pixels, or even the entire image, into consideration at once.

This chapter is a review (or introduction, depending on your background) covering the digital theories that underlie sampling and linear operators. The material will be covered from a practical viewpoint: What should we know about the basic operators of image processing? What do they mean in terms of getting at the underlying information in the image? What are the repercussions of digital sampling theory? The presentation is cursory; we will not explore proofs or rigorous mathematics associated with the presented theorems or tools. For example, the presentation of the Fourier transform will be far from exhaustive, not even listing all of the important mathematical properties of the transform. We also will not be presenting implementations. For more in-depth discussion, we refer the reader to standard textbooks in the field, in particular, *Signals and Systems*, by Oppenheim and Willsky [2], and *Digital Image Processing* by Castleman [1].

2.2 Images

In this book, we often are concerned with 3D, or *volume*, images. A volume image, ϕ , is a mapping from \mathbb{R}^3 to \mathbb{R}^n where $n = 1$ for the typical, scalar-valued volume. More precisely:

$$\phi : U \mapsto \mathbb{R} \text{ and } U \subset \mathbb{R}^3, \quad (2.1)$$

where U is the domain of the volume. The image ϕ is often written as a function $\phi(x,y,z)$.

Throughout this chapter, many examples will be presented as either 1D or 2D, where they may be more easily examined (at least in print), and then generalized to higher dimensions. In general, for a specific number of dimensions, we will use the corresponding function notation ($\phi(x)$, $\phi(x,y)$, or $\phi(x,y,z)$).

We denote the sampling of continuous images into discrete, digital images as follows. If F is a discrete sampling of a 2D image $\phi(x, y)$ then we can say that

$$F_{i,j} = \phi(x_i, y_i). \quad (2.2)$$

We likewise use notation to distinguish between discrete and continuous differential operators. To find the partial derivative in the x -direction of a discretely sampled image we can say that

$$\phi_x(x_i, y_i) = \left. \frac{\partial \phi}{\partial x} \right|_{x_i, y_i} \approx \delta_x F_{i,j}, \quad (2.3)$$

using a method such as

$$\delta_x F_{i,j} \equiv \frac{F_{i+1,j} - F_{i-1,j}}{x_{i+1} - x_{i-1}} = \frac{F_{i+1,j} - F_{i-1,j}}{2h}, \quad (2.4)$$

to approximate the partial derivative, where h is the grid spacing (normally assumed to be 1 pixel). The value $\delta_x F_{i,j}$ is an approximation of the instantaneous partial derivative of $\phi(x_i, y_i)$ at (x_i, y_i) in the x -direction. Equation (2.4) is the method of *central differences*, one of a number of commonly used methods to approximate the derivatives of uniformly, discretely sampled datasets.

2.3 Point Operators

The simplest operators on an image are those that treat each pixel independently from its neighbors. Point operators usually ignore information about pixel location, relying only on pixel intensity. A surprising number of useful things can be done with such operators. Some examples follow.

2.3.1 Thresholding

The simplest form of segmentation involves thresholding the image intensity. Thresholding is inherently a binary decision made on each pixel independent of its neighbors. Intensity above the threshold yields one classification, below the threshold another. This simple operation can be surprisingly effective, especially in data such as Computerized Tomography (CT), where the pixel value has real-world significance. The pixel value in a CT image is reported in Hounsfield units, which are calibrated to correspond to the attenuation of x-rays measured within the particular sample of tissue. Since bone is much more radiopaque than other tissues, bone can be segmented effectively in a CT image by setting the threshold between the Hounsfield value for bone and other tissues. An example is shown in Figure 2.1, in which a surface was reconstructed at the threshold between bone and other tissue in a CT scan of a fractured tibia.

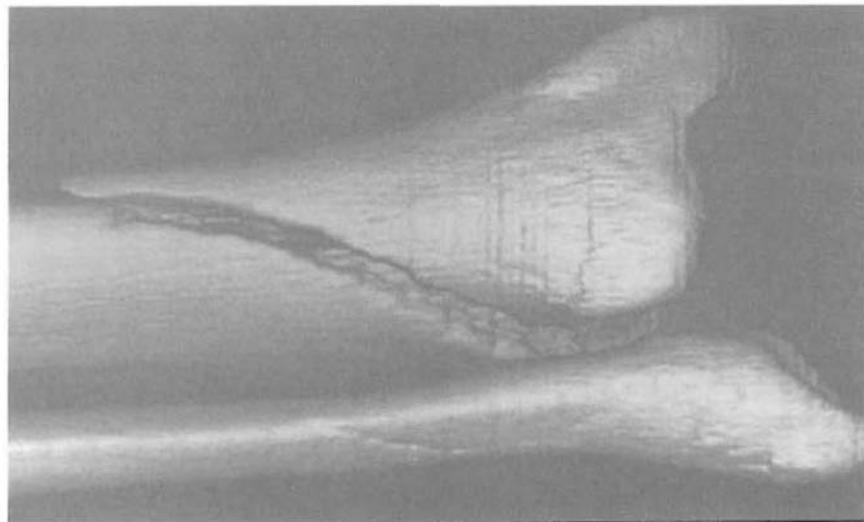


Figure 2.1. A simple threshold segments the bone from other tissue in a CT scan of a fractured tibia.

2.3.2 Adaptive Thresholding

In most applications of thresholding, the optimal threshold is not known beforehand, but depends instead upon the particular imaging modality, tissue-type, patient, and even the individual image acquisition. For example, the intensity of a given pixel in an ultrasound image will depend on the scanner's current gain settings and the intervening tissue types. In such cases, an optimal threshold might be chosen manually by looking at the image itself or at a histogram of intensity values from the tissues to be segmented. Ideally, such a histogram will look like Figure 2.2, in which the two tissue types (A and B) have pixel intensities whose ranges hardly overlap. In such a case, where the histogram is bimodal, it is fairly easy to select the optimum threshold. In the real world, this is often not the case, however, and segmentation must take other factors into account, such as neighborhood statistics and expected shape.

2.3.3 Windowing

Often it is advantageous to change the range of intensity values for an image to adjust the brightness and contrast of what will be displayed on a screen. Display is an essential part of image analysis, because it is the portal to the human visual system, whose capabilities far surpass any computer system likely to be

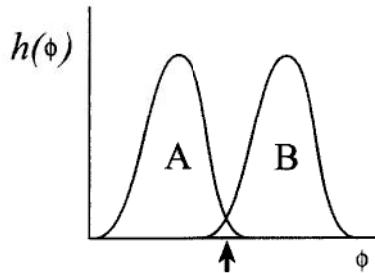


Figure 2.2. Histogram $h(\phi)$ of pixel intensity ϕ for tissue types A and B, with optimum threshold for segmentation (arrow).

built in the near future. The dynamic range of image display systems, between the brightest and darkest pixel the screen can produce, is quite small compared to the dynamic range of the human eye. Many medical imaging modalities are also capable of a greater dynamic range than can be displayed. The display itself therefore represents a bottleneck in the overall system. To minimize loss of information, the brightness and contrast of the image may be adjusted so that the brightest and darkest pixels in the image exactly match the extremes of the display, as shown in Figure 2.3. This is a linear function remapping the intensity of each pixel, with contrast being the slope of the function and brightness being the offset. In some clinical systems, the process is called *windowing*.

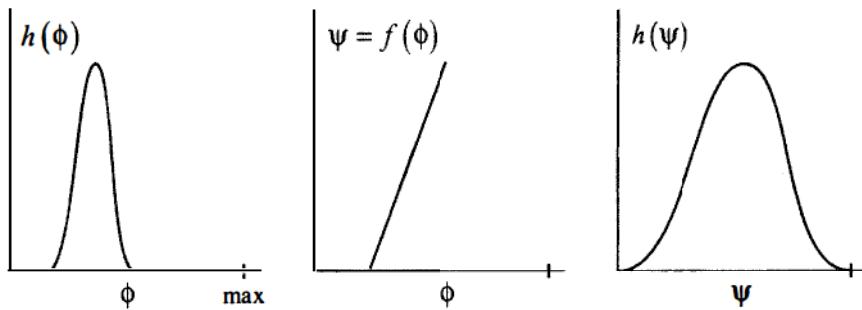


Figure 2.3. Windowing intensity histograms. Left: Histogram $h(\phi)$ of pixel intensity ϕ of original image, on a scale from zero to the maximum value of the display; Center: windowing function $\psi = f(\phi)$ to yield a new intensity ψ ; Right: new histogram $h(\psi)$ in which the dynamic range of the image matches that of the display.

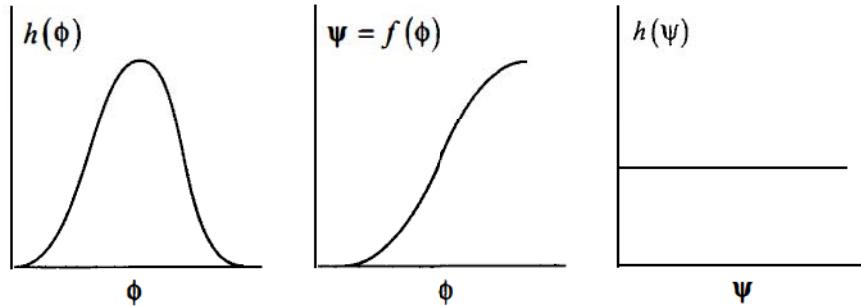


Figure 2.4. Histogram equalization. Left: Histogram $h(\phi)$ of pixel intensity ϕ of original image; Center: equalizing function $\psi = f(\phi)$ to yield a new intensity ψ ; Right: new histogram $h(\psi)$ in which each intensity is equally represented.

2.3.4 Histogram Equalization

A further step may be taken to optimize the match between the image and the display system, which ensures that each level of pixel intensity is equally represented in the image. This is called **histogram equalization**, and although it is non-linear, we include it here as a useful point operator. As shown in Figure 2.4, histogram equalization entails finding a monotonic function $\psi = f(\phi)$ that remaps pixel intensity so that the histogram becomes a constant function of intensity ψ .

2.3.5 Color Maps

Color video images typically have three separate channels to record the red, green, and blue levels from the camera. Color displays are capable of communicating these images to the human visual system, but such displays also can be used to enhance the display of single-channel gray-scale images through the use of colors, artificially assigned to particular pixel intensities. Such color maps are often used to overlay additional information in anatomical images, such as Doppler flow in ultrasound images. The color map usually follows some perceptually consistent ordering scheme, such as the rainbow, or especially bright colors may be assigned to emphasize pixel values above certain levels.

2.3.6 Algebraic Operators

Some point operators combine two images, keeping each pixel separate from its neighbors, while combining it with the corresponding pixel in another image. Algebraic point operators may be used to add images together, for example, in averaging many images acquired from the same sample to reduce noise. Or they

may be used to subtract one image from another, such as in Digital Subtraction Angiography where the difference between sequential fluoroscopic images, before and after contrast, leaves just the contrast without the anatomical structures. Another use of algebraic point operators is to mask an image by multiplying it by a second, binary image, containing ones where the first image is to be passed through, and zeroes where it is to be masked.

2.3.7 Location-Dependent Point Operators

All the point operators described so far apply the same operation to every pixel in the image. In some cases it is useful to make the operation depend upon pixel location, for example, to correct for spatial inhomogeneity in the image acquisition. Pixels are kept independent of each other, but in a framework where location within the image makes a difference.

2.4 Linear Filtering

Although many useful things can be done to an image using point operators on individual pixels, most image processing tasks require the simultaneous consideration of multiple pixels. Techniques that combine multiple pixels in a linear and space-invariant manner are known collectively as *linear filtering*. Two standard techniques for linear filtering, *convolution* and the *Fourier transform*, comprise the rest of this chapter.

2.4.1 Convolution

If you have ever adjusted the focus on a camera you have performed convolution in the continuous domain. If you have run a blurring function or performed edge enhancement using a commercial graphics program, it is likely that you were using discrete convolution. In image processing, convolution filters are used to make measurements, detect features, smooth noisy signals, and deconvolve the effects from image acquisition (e.g., deblurring the known optical artifacts from a telescope). The effect of the particular filter depends on the nature of the input filter, or *kernel*.

Convolution, denoted with the operator \otimes can be described in 1D as a continuous operation applying the filter kernel $h(x)$ (itself an image) to an image $\phi(x)$ using the following integral form:

$$\phi(x) \otimes h(x) = \int_{-\infty}^{\infty} \phi(x - \tau) h(\tau) d\tau. \quad (2.5)$$

Note that the expression $\phi(x) \otimes h(x)$ itself describes an image mapping. Thus the following equality holds:

$$\phi(x) \otimes h(x) = (\phi \otimes h)(x). \quad (2.6)$$

The expressions $\phi(x) \otimes h(x)$ and $(\phi \otimes h)(x)$ are used interchangeably, with the position index x often omitted to help streamline and clarify the notation especially when higher dimensional (e.g., 3D) images are involved. In the practice of digital image processing, $h(x)$ typically does not have infinite extent, nor is it infinitely dense; however, when working in the continuous domain of linear filtering theory, considering kernels that are infinitely wide (thus avoiding the complications of truncation) is often more convenient. We will elaborate later on discrete functions with finite extent in the discussion on sampling.

We can generalize the 1D convolution operation to 2D and 3D images. Thus, in 2D, convolution becomes

$$\phi(x,y) \otimes h(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi(x-\tau, y-v) h(\tau, v) d\tau dv. \quad (2.7)$$

Similarly, convolution in 3D is expressed as

$$\begin{aligned} \phi(x,y,z) \otimes h(x,y,z) = \\ \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi(x-\tau, y-v, z-w) h(\tau, v, w) d\omega dv d\tau. \end{aligned} \quad (2.8)$$

2.4.2 Example—Convolution as Noise Reduction (1D)

One of the most common uses of convolution is as a filter to suppress noise in an image. Consider the Gaussian function as a smoothing filter kernel. If convolved with an image containing high-frequency noise, the result is an output image that locally averages the image intensities, reducing noise at the cost of some “sharpness” of the original image. That is, the resulting image has less high-frequency noise, but also has blurred edges.

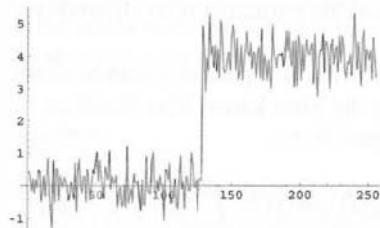


Figure 2.5. A 1D noisy step edge: an example input signal $\phi(x)$.

Figure 2.5 shows a noisy 1D step discontinuity, or edge. The function $\phi(x)$ is shown from $x = 0$ to $x = 255$ with a step function at $x = 128$. The signal to noise ratio is approximately 4:1.

Figures 2.6, 2.7, and 2.8, show a progression of filtered versions of the input from Figure 2.5 with Gaussian filter kernels of increasing aperture or width σ . The Gaussian function $g(x)$ is defined as

$$g(x, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}. \quad (2.9)$$

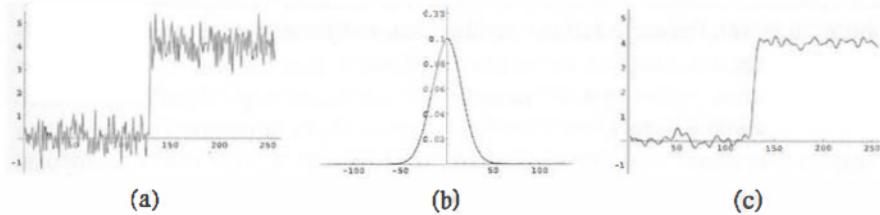


Figure 2.6. Gaussian filtering of input $\phi(x)$ where $\sigma = 16$. (a) Input $\phi(x)$; (b) a 1D Gaussian kernel, $g(x, \sigma)$ where $\sigma = 16$; (c) output of $\phi(x) \otimes g(x, 16)$.

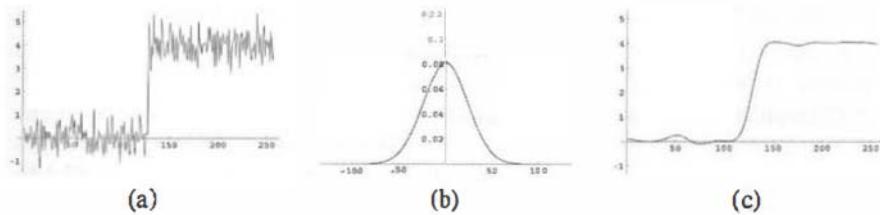


Figure 2.7. Gaussian filtering of input $\phi(x)$ where $\sigma = 24$. (a) Input $\phi(x)$; (b) a 1D Gaussian kernel, $g(x, \sigma)$ where $\sigma = 24$; (c) output of $\phi(x) \otimes g(x, 24)$.

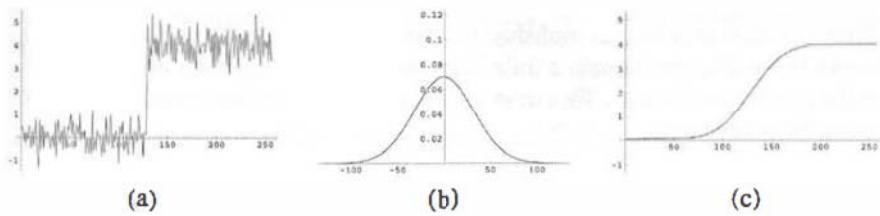


Figure 2.8. Gaussian filtering of input $\phi(x)$ where $\sigma = 32$. (a) Input $\phi(x)$; (b) a 1D Gaussian kernel, $g(x, \sigma)$ where $\sigma = 32$; (c) output of $\phi(x) \otimes g(x, 32)$.

Notice how increasing the aperture of the Gaussian decreases the noise but also blurs the edge. This trade-off of resolution for noise reduction is one of the many considerations in the design of linear filter systems. An entire approach to image analysis, called “scale space,” depends upon the Gaussian to produce a broad range of blurred versions of a given image.

2.4.3 Properties of the Convolution Operation

The nature of a linear filtering operation depends on the properties of the filter kernel. For instance, the “shape” of a 2D or 3D kernel will determine whether the operation remains invariant with respect to rotation. Independent of kernel shape, however, the convolution operation has many useful properties, including the following:

Convolution is linear:

$$(Ap + Bq) \otimes r = A(p \otimes r) + B(q \otimes r). \quad (2.10)$$

Convolution is commutative:

$$p \otimes q = q \otimes p. \quad (2.11)$$

Convolution is associative:

$$(p \otimes q) \otimes r = p \otimes (q \otimes r). \quad (2.12)$$

Convolution is distributive over addition:

$$p \otimes (q + r) = p \otimes q + p \otimes r. \quad (2.13)$$

These combined properties create the justification for using convolution as the principal operation in linear filtering.

2.4.4 Differentiation by Convolution

Differentiation may be accomplished by using convolution. We will see how this works in the discrete domain a little later, but first let us consider differentiation in the continuous domain. We can explicitly denote differentiation as convolution using the \otimes operator:

$$\frac{\partial}{\partial x} \phi = \frac{\partial}{\partial x} \otimes \phi. \quad (2.14)$$

Equation (2.14) depicts the differential operator as a kernel by which convolution accomplishes differentiation. This is hard to illustrate using the ideal differential

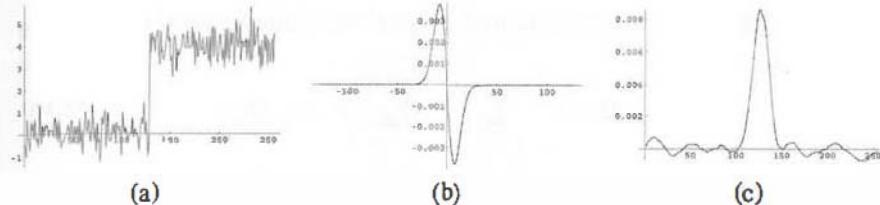


Figure 2.9. Taking derivatives of a noisy input by convolution with the derivative of a Gaussian kernel. (a) 1-D input $\phi(x)$; (b) $\frac{\partial}{\partial x}g(x, \sigma)$ where $\sigma = 3$; (c) $\phi(x) \otimes \frac{\partial}{\partial x}g(x, \sigma)$.

kernel, because such a kernel has infinitesimal width and infinite height. In the real world such a kernel is impossible. In any event, because differentiation enhances high-frequency noise, it is often necessary to regularize (smooth) a noisy image before computing its derivative, by first convolving the function with a smoothing kernel $h(x)$. It follows from the associative and commutative properties of convolution that

$$h(x) \otimes \frac{\partial}{\partial x} \phi(x) = \frac{\partial}{\partial x} h(x) \otimes \phi(x). \quad (2.15)$$

In other words, the derivative of a function $\phi(x)$ convolved with a filter kernel $h(x)$ is equivalent to convolving $\phi(x)$ with the derivative of $h(x)$. This suggests that one of the easiest ways to compute the derivative of a function is through convolution with the derivative of some smoothing kernel. We have already demonstrated the use of the Gaussian $g(x)$ as a smoothing kernel. The Gaussian's infinitely differentiable properties make it attractive for taking derivatives as well as smoothing.

Figure 2.9 depicts a noisy 1D input signal for which derivative information is desired. When convolved with the derivative of a Gaussian, the resulting output reports the *derivative of a smoothed version* of the input. By the commutative and associative properties of convolution, it can just as well be described as a *smoothed version* of the *derivative of the input*. This technique for differentiating functions can be extended to higher order derivatives, and to higher numbers of dimensions.

2.4.5 Convolution of Discretely Sampled Data

Convolution of discretely sampled data is similar to the continuous form, except that discrete summation is substituted for integration. Also, since discrete filter kernels cannot be implemented with infinite extent, the limits of the summations do not range from $-\infty$ to ∞ but are rather constrained to the size of the filter

kernel. In 2D, a discrete convolution of image P with finite kernel Q looks like

$$P_{x,y} \otimes Q_{x,y} = \sum_i^{\text{domain}_x[Q]} \sum_j^{\text{domain}_y[Q]} P_{x-i,y-j} Q_{i,j}. \quad (2.16)$$

The discrete version of convolution shares all the above-mentioned attributes of its continuous cousin. Convolution of discretely sampled data is linear, commutative, associative, and distributive over addition. As in the continuous domain, convolution in the discrete domain can be used to smooth noisy data or detect boundaries, depending on the choice of kernel. An example of discrete convolution for differentiation in 2D has already been seen in the *central differences operation* (Equation (2.4)). The kernel in that case would be

$$\begin{bmatrix} 0 & 0 & 0 \\ -0.5 & 0 & 0.5 \\ 0 & 0 & 0 \end{bmatrix}, \quad (2.17)$$

which has the effect of subtracting the pixel to the left from the pixel to the right, just as Equation (2.4) specifies, to represent the x -component of the gradient. The following section shows more examples using convolution for smoothing and taking derivatives in 2D.

2.4.6 The Binomial Kernel

One particularly useful kernel for discrete convolution is the binomial kernel, generated by repeated convolutions with a simple box of identical values, as shown here:

$$[1], \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}. \quad (2.18)$$

With successive iterations, the binomial kernel approaches a Gaussian shape (by the Central Limit Theorem). The scale of the Gaussian is determined by the number of iterations. (The kernels shown in Equation (2.18) would be normalized by 1/4 with each iteration.)

2.4.7 Difference of Offset Gaussians

Boundaries in an image represent areas of high gradient magnitude, i.e., the image intensity increases or decreases rapidly across the edge. The gradient vector,

$$\left[\frac{\partial \Phi}{\partial x}, \frac{\partial \Phi}{\partial y} \right], \quad (2.19)$$

is oriented in the direction of the steepest change in image intensity, normal to the implied boundary. The gradient magnitude,

$$\left(\frac{\partial\phi}{\partial x}\right)^2 + \left(\frac{\partial\phi}{\partial y}\right)^2, \quad (2.20)$$

represents an orientation-independent measure of boundary strength. How can we determine the individual components of the gradient vector?

We have seen the *central difference operator* already. Another way to represent a gradient component is to use a kernel known as the *difference of offset gaussians* (DoOG).

$$[-1 \ 0 \ 1], \begin{bmatrix} -1 & -2 & 0 & 2 & 1 \\ -2 & -4 & 0 & 4 & 2 \\ -1 & -2 & 0 & 2 & 1 \end{bmatrix}. \quad (2.21)$$

The two DoOG kernels shown in Equation (2.21) correspond to iterations 1 and 3 in Equation (2.18). Each kernel is the difference between two copies of the corresponding binomial kernels displaced along the x -axis. Convolution with these DoOG kernels measures the gradient component in the x -direction. Similar kernels can be constructed by displacing copies of the binomial kernel in the y -direction to measure the y -component of the gradient. DoOG kernels can be constructed for images in 3D or higher dimensions. The number of iterations of the underlying binomial kernel determines the scale of the DoOG.

Figure 2.10 shows the results of detecting the individual gradient components as well as the gradient magnitude of a simple image containing a rectangular object. Notice the orientations of the edges detected by the individual gradient components, and the orientation-independence of the gradient magnitude.

2.4.8 2D Example—Higher-Order Differentiation

We have seen how the gradient vector represents the strength and orientation of the boundary. But what of higher order differentials? Where the first derivative (gradient) is represented as a vector, the second derivative is a matrix, known as the Jacobian

$$\begin{bmatrix} \frac{\partial^2\phi}{\partial x^2} & \frac{\partial^2\phi}{\partial x\partial y} \\ \frac{\partial^2\phi}{\partial y\partial x} & \frac{\partial^2\phi}{\partial y^2} \end{bmatrix}. \quad (2.22)$$

The elements of this matrix are useful in a number of ways. For example, the Laplacian,

$$\frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial y^2}, \quad (2.23)$$

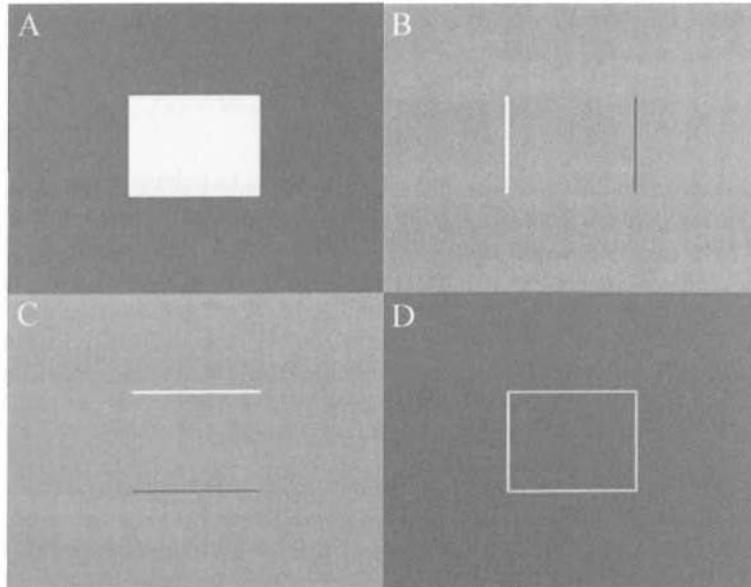


Figure 2.10. Gradient components and gradient magnitude: (A) original image; (B) x -component of gradient; (C) y -component of gradient; (D) gradient magnitude.

sums the diagonal terms of the Jacobian to yield a rotationally invariant representation of the second derivative of image intensity. A common 3×3 kernel representing the Laplacian in 2D is

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}. \quad (2.24)$$

Convolution with this matrix yields zero along a boundary, no matter what orientation the boundary has. For example, convolution with the following 3×3 patch of an image containing a diagonal boundary (between regions with intensities of 1 and 3 respectively),

$$\begin{bmatrix} 2 & 3 & 3 \\ 1 & 2 & 3 \\ 1 & 1 & 2 \end{bmatrix}, \quad (2.25)$$

yields zero for the center pixel (which is directly on the boundary).

For an intuitive understanding of why the second derivative should be zero on the boundary let us examine the 1D case.

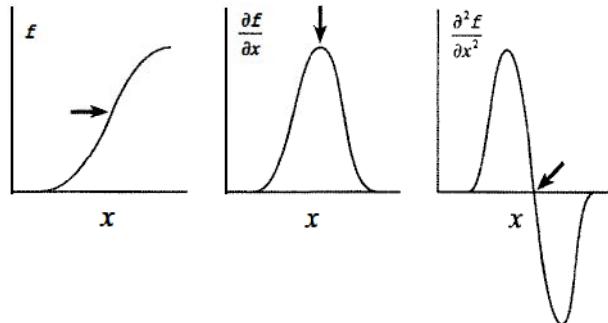


Figure 2.11. 1D case of the intensity function across a boundary, its first derivative, and its second derivative showing a zero crossing at the boundary location.

As shown in Figure 2.11, the boundary is located at an inflection point (arrow) in the intensity curve

$$\phi(x)$$

corresponding to a maximum in the first derivative

$$\frac{\partial \phi(x)}{\partial x},$$

and a zero crossing in the second derivative

$$\frac{\partial^2 \phi(x)}{\partial x^2}.$$

In two or more dimensions, we use the Laplacian to capture the “total” second derivative at a pixel. Mathematically, the Laplacian is the divergence of the gradient. Divergence is a common concept in fluid dynamics, describing how much more fluid enters a region than leaves it. In our case, it describes how much more gradient is “entering” a pixel than “leaving” it from any direction (making it rotationally invariant). A boundary exists where there is no net change in gradient (i.e., the gradient is at a maximum on the boundary), making the Laplacian zero, just as in the 1D case. The difference of Gaussian (DOG) kernel, which we will discuss in Section 2.4.9, yields results similar to the Laplacian, because the DOG kernel has a similar shape to the Laplacian.

Let us now consider the terms *off* the diagonal of the Jacobian matrix (Equation (2.22)). These can be used to measure the curvature of boundaries. The individual components of the gradient at a curved boundary change as one moves orthogonally to that particular component. This type of change shows up as partial second derivatives of intensity off the diagonal of the Jacobian. For illustration,

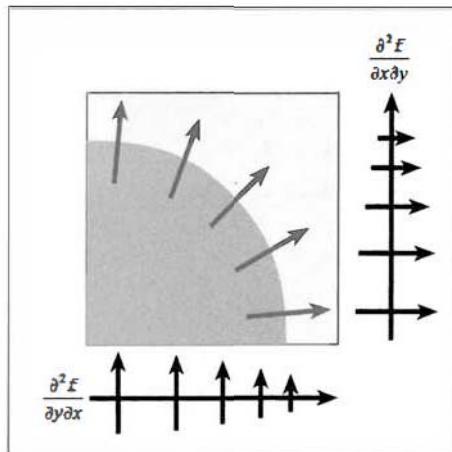


Figure 2.12. The curvature of a boundary shows up in the terms of the Jacobian off the diagonal.

consider the curved boundary segment shown in Figure 2.12. The component of the Jacobian,

$$\frac{\partial^2 \phi}{\partial y \partial x},$$

indicates a decrease in the y -component of the gradient as one moves in the positive x -direction, whereas

$$\frac{\partial^2 \phi}{\partial x \partial y}$$

shows a decrease in the x -component of the gradient as one moves in the positive y -direction.

It is possible to construct features using even higher order derivatives. In general, approaches to image analysis based on partial differential equations (PDEs) have been quite popular and well researched.

2.4.9 Difference of Gaussians (DOG)

Results similar to those just described for the Laplacian can be achieved using the difference between two concentric Gaussian functions with different apertures. This function is called a difference of Gaussians, or DOG, kernel (not to be confused with the difference of *offset* Gaussians, or DoOG, kernel already described; see Equation (2.21)). As with the Laplacian, the DOG kernel can detect edges independent of orientation, but not the orientation itself. The process is also called “unsharp masking.” The result is an *edge-enhanced* image.

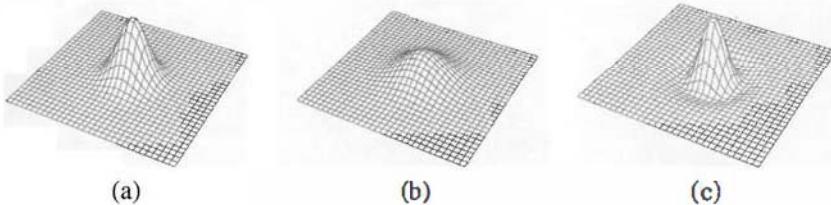


Figure 2.13. A difference of Gaussians filter kernel (depicted in 3D as a height field): (a) 2D Gaussian kernel $g(x,y,\sigma_1 = 16)$; (b) 2D Gaussian kernel $g(x,y,\sigma_2 = 32)$; (c) $g(x,y,\sigma_1) - g(x,y,\sigma_2)$.

Specifically, given a 2D input image $\phi(x,y)$ and two Gaussian filter kernels of differing aperture, $g(x,y,\sigma_1)$ and $g(x,y,\sigma_2)$ where $\sigma_1 < \sigma_2$, an edge enhanced image $\phi'(x,y)$ can be formed by a linear combination of the two filters. That is,

$$\phi'(x,y) = \phi(x,y) \otimes (g(x,y,\sigma_1) - g(x,y,\sigma_2)). \quad (2.26)$$

From Equation (2.13) it follows that $\phi'(x,y)$ is also the subtraction of two filtered images

$$\phi'(x,y) = (\phi(x,y) \otimes g(x,y,\sigma_1)) - (\phi(x,y) \otimes g(x,y,\sigma_2)). \quad (2.27)$$

In other words, the difference of two “blurred” versions of the same image yields a version of the original image where edge information is accentuated, and where regions of continuous intensity have a zero value. Areas near boundaries are strongly negative or strongly positive (depending on which side of a boundary you are on) and the boundary itself is denoted by the closed curve of *zero-crossings* (as with the Laplacian), where the image intensity crosses from positive to negative values.

Figure 2.13 shows a particular 2D difference of Gaussians kernel

$$g(x,y,\sigma_1 = 16) - g(x,y,\sigma_2 = 32) \quad (2.28)$$

as a 3D function where intensity is plotted as a height field above an x - y plane. Figure 2.14 is the same kernel shown in Figure 2.13 applied to a circular pulse function (white circle on a gray background), also represented as a height field. Figure 2.15 shows the same functions as Figure 2.14, but depicted as a 2D intensity field. The areas in the original image with constant intensity, inside and outside the circle, result in a flat signal value of approximately zero (gray). Moving outward from the center of the circle, at the boundary there is a rise in the output intensity followed by a negative dip. The zero crossing between them represents the boundary itself.

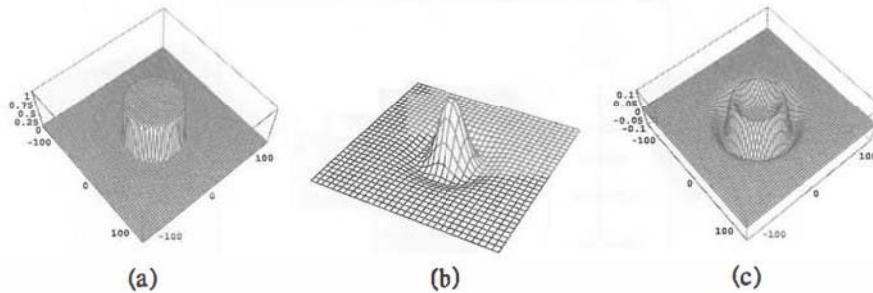


Figure 2.14. 2D image filtered by a difference of Gaussians (depicted in 3D as a height field): (a) 2D input $\phi(x,y)$; (b) $g(x,y,\sigma_1 = 16) - g(x,y,\sigma_2 = 32)$; (c) $\phi(x,y) \otimes (g(x,y,\sigma_1) - g(x,y,\sigma_2))$.

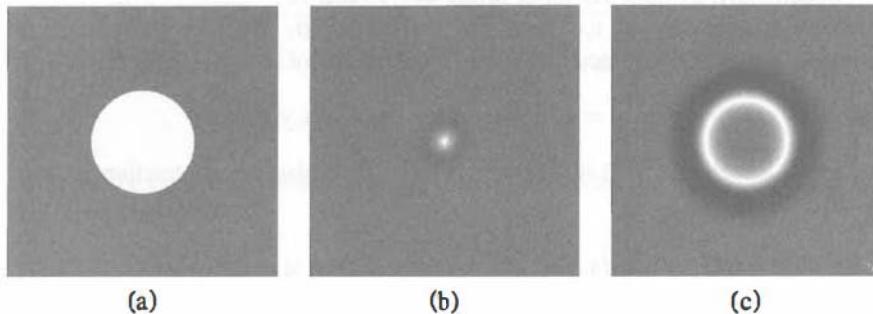


Figure 2.15. 2D image filtered by a difference of Gaussians (depicted as a 2D grayscale density field). (a) 2D input $\phi(x,y)$; (b) $g(x,y,\sigma_1 = 16) - g(x,y,\sigma_2 = 32)$; (c) $\phi(x,y) \otimes (g(x,y,\sigma_1) - g(x,y,\sigma_2))$.

Another version of the DOG kernel deserves mention. By doubling the amplitude of the Gaussian with the smaller aperture in Equation (2.28) to yield a new kernel,

$$2g(x,y,\sigma_1 = 16) - g(x,y,\sigma_2 = 32), \quad (2.29)$$

we modify its behavior from *edge detection* to *contrast enhancement*. The distinction is somewhat subtle. The area under the kernel is now 1 instead of 0. Thus, in image regions of constant pixel value, convolution passes through the pixel value unchanged rather than setting it to 0. Figure 2.16 shows the results. This DOG kernel returns unchanged the non-zero pixel value inside the circle, while exaggerating the discontinuity in a band around the boundary. Compare Figure 2.15(c), in which the interior of the circle is set to 0 (gray), to Figure 2.16(c), in which the interior of the circle is passed through unchanged (white).

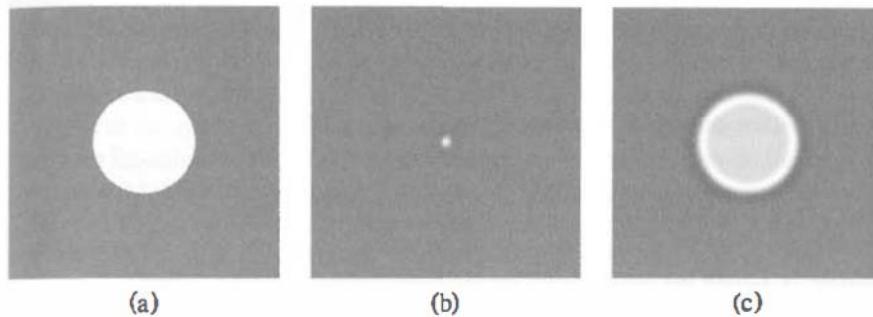


Figure 2.16. A difference of Gaussians filter used for contrast enhancement (depicted as a 2D greyscale density field). This method is also known as *unsharp masking*: (a) 2D input $\phi(x,y)$; (b) $2g(x,y,\sigma_1 = 16) - g(x,y,\sigma_2 = 32)$; (c) $\phi(x,y) \otimes (2g(x,y,\sigma_1) - g(x,y,\sigma_2))$.

2.4.10 Beware of Aliasing

The primary caveat regarding discrete convolution is that all such discrete operations are subject to error induced by the effects of sampling, a class of artifacts known as *aliasing*. Before we can appreciate the problems associated with sampling, we must first understand the nature of the sampling operation and its effects on the information embedded within the image. For that, we need the Fourier transform.

2.5 The Fourier Transform

One of the most important tools for understanding images (1D, 2D, 3D or even higher) is the Fourier transform. The Fourier transform of an image decomposes it into component sinusoidal spatial functions. The representation of an image as its constituent frequencies is a computationally useful means of viewing and manipulating image data. The relationship between an input function and its Fourier transform is governed by the following equation:

$$\mathcal{F}(\phi(x)) = \int_{-\infty}^{\infty} \phi(x) e^{-i2\pi\omega x} dx = \Phi(\omega). \quad (2.30)$$

This relationship maps the spatial domain x to the frequency domain ω . These are often referred to, respectively, as “image space” and “frequency space.” The Fourier transform \mathcal{F} of $\phi(x)$ is a complex number $\Phi(\omega)$, encompassing magnitude and phase at a particular frequency ω . As a transform, \mathcal{F} is reversible. The inverse Fourier transform \mathcal{F}^{-1} is

$$\mathcal{F}^{-1}(\Phi(\omega)) = \int_{-\infty}^{\infty} \Phi(\omega) e^{i2\pi\omega x} d\omega = \phi(x), \quad (2.31)$$

which reconstructs $\phi(x)$ from its spectrum $\Phi(\omega)$. It should be emphasized that, in the case of images, these are *spatial*, not temporal, frequencies: periodic variations of intensity in space. Fourier transforms can also be performed in 2D and 3D, as will be described in Section 2.5.3.

2.5.1 Key Properties of the Fourier Transforms

There are some key properties of frequency space and the Fourier transform that creates it, which we will discuss for both the continuous and discrete domains. We will emphasize two particular characteristics of the Fourier transform, which have to do with the effects on frequency space of scaling and convolution in image space.

Scaling in Image Space \leftrightarrow Inverse Scaling in Frequency Space

Compression of a function in the spatial domain leads to an inversely proportional broadening of the spectrum in the frequency domain, i.e., reducing the size of an image increases its spatial frequencies. Mathematically speaking, for $a \geq 1$,

$$\mathcal{F}(\phi(ax)) = \frac{1}{a} \Phi\left(\frac{\omega}{a}\right). \quad (2.32)$$

The opposite is also true. Increasing the size of an image decreases its spatial frequencies, which is to say, Equation (2.32) with $0 < a \leq 1$.

Convolution in Image Space \leftrightarrow Multiplication in Frequency Space

Convolution between an image and a kernel in image space translates to multiplication of their spectra in frequency space. That is, given a kernel $h(x)$, and its Fourier transform, $H(\omega)$, the following relation holds:

$$\mathcal{F}(\phi(x) \otimes h(x)) = \Phi(\omega)H(\omega). \quad (2.33)$$

This is known as the *convolution theorem*, and it is perhaps the most important theorem in linear system analysis. Given the convolution theorem, it is easy to conceive of why convolution is distributive over addition, symmetric, commutative, etc. It shares most of the properties of multiplication.

Linear filtering under these circumstances can now be considered in a different light. It can be changed from an integral with infinite extent in image space (a difficult continuous operation to implement with a digital computer) to a multiplication in frequency space. Moving to and from frequency space (simple multiplication replaces convolution) requires that only two forward Fourier transforms and one inverse Fourier transform be calculated

$$\phi(x) * h(x) = \mathcal{F}^{-1}(\mathcal{F}(\phi(x))\mathcal{F}(h(x))). \quad (2.34)$$

Depending on the size of both the filter kernel and the input signal, casting the convolution problem in the frequency domain often makes difficult iterative problems more tractable.

Due to the symmetry of image space and frequency space (note the similarity between Equations (2.30) and (2.31)), the converse of the convolution theorem is also true: Multiplication in image space corresponds to convolution in frequency space. This leads to the sampling artifact, *aliasing*, as will be discussed with regards to the Comb function in Section 2.5.2.

There are many other properties of the Fourier transform associated with its symmetry and its use of complex exponentials, which we cannot hope to cover here. However, the two properties just discussed, of spatial/frequency scaling and convolution/multiplication, allow us to make some important points in the following sections.

2.5.2 Four Important Transform Pairs

Four functions commonly used in manipulating images are the comb function (used to sample data discretely in the image domain), the box filter (a square pulse, used as a nearest neighbor interpolant in reconstruction), the pyramid filter (a triangular shaped filter used for linear interpolation in reconstruction), and the Gaussian filter (a good all purpose filter, except for its infinite extent).

Figures 2.17, 2.18, 2.19, and 2.20 show these functions represented in 1D image space, along with their Fourier transforms in frequency space. By studying these four transform pairs, a number of important characteristics of the relationships between image space and frequency space become clear.

Comb Function

A comb function is a series of *impulses*. Impulses, or *delta functions*, play a central role in systems theory as the identity element of convolution. In the continuous domain, the impulse is an infinitely narrow, infinitely high spike with finite area. Convolution of any function $f(x)$ with an impulse yields a perfect copy of $f(x)$, as if taking a picture with an infinitely high-resolution camera. This is because multiplication of $f(x)$ by an impulse isolates a perfect sample of $f(x)$, suitable for integration, at the location of the impulse. Multiplying $f(x)$ by the comb function captures a set of such samples, evenly spaced, to produce a digital image.

It is essential to understand what happens in frequency space, as the price for such sampling. As shown in Figure 2.17, the spectrum of a comb function is itself a comb function in frequency space. Each impulse in the spectrum represents a single frequency of infinite energy but finite power, i.e., a sinusoid. Multiplying (sampling) $f(x)$ with the comb function in image space leads to convolving their

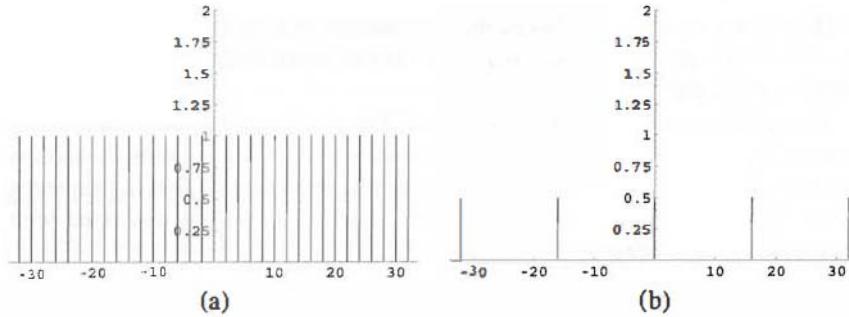


Figure 2.17. The comb filter and its Fourier transform. This function is used for sampling continuous functions into discrete grids: (a) sampling comb; (b) the Fourier transform of the sampling comb.

Fourier transforms, as already discussed in Section 2.5.1. So whatever spectrum is produced by the Fourier transform $F(x) = \mathcal{F}(f(x))$ will be repeated by the comb function in frequency space at each impulse. As long as $F(x)$ is narrow enough not to overlap with its neighboring copy, this is not a problem. But if $F(x)$ is broader than the distance between two adjacent impulses (in the spectrum of the comb function), aliasing will occur. This restriction on the width of the spectrum $F(x)$ is known as the *Nyquist criterion*. (It is also known as the *Shannon sampling theorem*). The maximum allowable frequency to avoid aliasing is called the *Nyquist frequency*, which is half the sampling frequency.

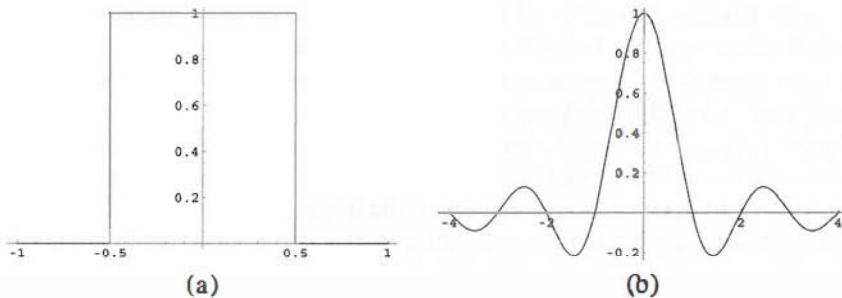


Figure 2.18. The box filter and its Fourier transform, the sinc function. This function is used as a nearest-neighbor interpolant to quickly reconstruct sampled functions: (a) the box function (or the nearest-neighbor interpolant); (b) the Fourier transform of the box function (also known as the sinc function).

Box Filter

The box filter, shown in Figure 2.18(a), is used as a reconstruction function in nearest-neighbor interpolation. Interpolation is a common task in image analysis, to re-sample an already discrete image to a different lattice of sample locations. Nearest-neighbor interpolation, as its name suggests, entails copying the value of the nearest pixel, a brute-force but rapid procedure. The Fourier transform of the box filter in Figure 2.18(b) demonstrates the potential for aliasing. The spectrum takes the form of a *sinc function*. Whereas the primary lobe of the sinc function is fairly well contained, side-lobes continue at regular intervals quite high up in frequency. If significant lobes extend beyond the Nyquist frequency, aliasing will occur. In a graphical display of the image, this artifact expresses itself as “jaggies.”

Pyramid Filter

While the box filter just discussed is used for nearest-neighbor interpolation, the pyramid filter, shown in Figure 2.19(a), is used for linear interpolation. This is a computationally more expensive, but more accurate, operation than nearest-neighbor interpolation, with a correspondingly better behavior in terms of aliasing. The spectrum of the pyramid filter, shown in Figure 2.19(b), has smaller side-lobes than those of the box filter in Figure 2.18(b), and therefore has less tendency to produce aliasing.

Gaussian Filter

The ultimate filter in terms of avoiding side lobes is the Gaussian filter. Since the Fourier transform of a Gaussian is itself a Gaussian, as shown in Figure 2.20,

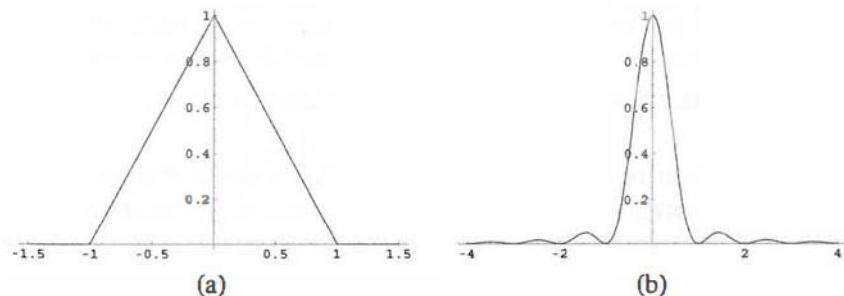


Figure 2.19. The pyramid filter and its Fourier transform. This function is used for linear interpolation and reconstruction of sampled functions. (a) The pyramid function (or the linear interpolant); (b) The Fourier transform of the pyramid function.

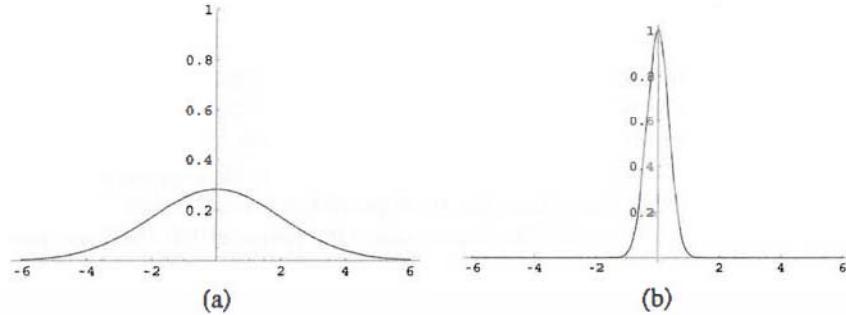


Figure 2.20. The Gaussian filter and its Fourier transform. The Gaussian is a smooth, continuous filter kernel in both image space as well as frequency space. The Fourier transform of a Gaussian function is also a Gaussian: (a) image space Gaussian; (b) the Fourier transform of a Gaussian (a Gaussian).

there are no side-lobes at all. Gaussian filters are widely used to blur images, and form the cornerstone of an entire approach to image analysis known as *scale space*.

2.5.3 The Fourier Transform in 2-D and 3-D

Thus far we have discussed Fourier transforms in only one dimension. Images, of course, tend to have greater numbers of dimensions. The Fourier transform, as shown in Equation (2.30), can be extended to 2D as follows:

$$\mathcal{F}(\phi(x,y)) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi(x,y) e^{-i2\pi(\omega x + \nu y)} dx dy = \Phi(\omega, \nu). \quad (2.35)$$

The inverse Fourier transform (Equation (2.31)) becomes

$$\mathcal{F}^{-1}(\Phi(\omega, \nu)) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \Phi(\omega, \nu) e^{i2\pi(\omega x + \nu y)} d\omega d\nu = \phi(x, y). \quad (2.36)$$

This process can be extended to 3D and beyond, in a similar manner. A number of important new properties apply to the multi-dimensional Fourier transform, of which we will mention three.

Separability

Among its most important attributes, the multi-dimensional Fourier transform is separable in each of the orthonormal basis dimensions. Thus Equation (2.35) can

be rewritten as

$$\mathcal{F}(\phi(x,y)) = \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} \phi(x,y) e^{-i2\pi\omega x} dx \right] e^{-i2\pi\nu y} dy = \Phi(\omega, \nu). \quad (2.37)$$

If the function $\phi(x,y)$ can be separated into component functions

$$\phi(x,y) = \phi_1(x)\phi_2(y), \quad (2.38)$$

then the Fourier transform can also be separated into components

$$\Phi(\omega, \nu) = \Phi_1(\omega)\Phi_2(\nu), \quad (2.39)$$

where

$$\Phi_1(\omega) = \mathcal{F}(\phi_1(x)) \text{ and } \Phi_2(\nu) = \mathcal{F}(\phi_2(y)). \quad (2.40)$$

Separability is applicable, for example, to the 2D Gaussian function, which is itself separable:

$$g(x,y) = g_1(x)g_2(y). \quad (2.41)$$

Therefore, the Fourier transform of $g(x,y)$ is

$$\mathcal{F}(g(x,y)) = G(\omega, \nu) = G_1(\omega)G_2(\nu), \quad (2.42)$$

where

$$G_1(\omega) = \mathcal{F}(g_1(x)) \text{ and } G_2(\nu) = \mathcal{F}(g_2(y)). \quad (2.43)$$

Rotational Invariance

Consider rotation of the coordinate system of a 2D image about the origin of the (x,y) plane by an angle θ through which each location (x,y) becomes (x',y') :

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}. \quad (2.44)$$

The 2D Fourier transform will likewise be rotated about the origin of the (ω, ν) plane by an angle θ with each location (ω, ν) becoming (ω', ν') , where

$$\begin{bmatrix} \omega' \\ \nu' \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} \omega \\ \nu \end{bmatrix}. \quad (2.45)$$

The image and its transform rotate together about their origins.

Projection

Consider the projection \mathcal{P}_x of $\phi(x,y)$ onto the x -axis:

$$\mathcal{P}_x(\phi(x,y)) = \int_{-\infty}^{\infty} \phi(x,y) dy. \quad (2.46)$$

Here, separability comes into play. By setting $v = 0$, Equation (2.37), reduces to

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi(x,y) e^{-i2\pi\omega x} dx dy = \Phi(\omega, 0) = \mathcal{F}(\mathcal{P}_x(\phi(x,y))). \quad (2.47)$$

In other words, the Fourier transform of the projection of $\phi(x,y)$ onto the x -axis is nothing but the 2D Fourier transform $\Phi(\omega, v)$ evaluated along the v -axis. Combining this result with rotational invariance leads to the somewhat surprising result that projecting a 2D image onto any line passing through the origin in image space transforms into a 1D spectrum taken from the same line along the Fourier transform in frequency space. This leads to the Radon transform, central to the process of filtered back projection, by which images are formed in computed tomography (CT), and other useful applications.

2.5.4 The Image as a Periodic Signal

The Fourier transform as shown in Equation (2.30) is an integral with infinite extent. A discrete implementation that can run on a digital computer is required to translate it from a useful mathematical abstraction to a practical tool. This raises several issues. In Section 2.5.2, while discussing the comb function, we addressed the issue of aliasing, which occurs when the spectrum contains frequencies above the Nyquist frequency. Now we take another look at sampling as it applies to Fourier analysis of images. The adaptation of the Fourier transform to discretely sampled data results in the *discrete Fourier transform*. Like its continuous cousin, the discrete Fourier transform can be applied to any signal, whether it is periodic or not. However, it is usually advantageous to treat an image as periodic. A digital image does not have infinite extent, nor can it be sampled at infinitesimal intervals. Computationally, it makes sense to use the Fourier series, instead of the transform. The Fourier series decomposes any periodic signal into discrete harmonics of the fundamental frequency.

To create a periodic signal, an image with n uniformly spaced samples is “wrapped” so that samples 0 through $n - 1$ index the data array exactly, while sample n “wraps” to 0. In fact, if n is the number of uniform samples, the effective index x_{eff} of a location x is, $x_{eff} = x \bmod n$. This repetition is applied to each of the dimensions (x , y , and z).

The interpretation of a discrete image as a periodic signal can also be viewed as an infinite concatenation of the image with itself. The wavelength corresponding to the fundamental frequency is the width of the image. This allows us to

represent the image as a sum of a finite number of sinusoidal functions: the discrete Fourier series, which is a linear transformation between n samples in image space and n samples in frequency space. The frequency space representation is capable of reproducing the original sampled image exactly without loss of information.

Computationally, the algorithm of choice for doing this is the fast Fourier transform (FFT), which is efficient even for very large data sets. One restriction of the FFT is that the number of samples in each dimension be a power of 2 (e.g., 128, 256, 512,...), but luckily many images already satisfy this requirement. When it is not the case, images may be “padded” with extra zeros.

2.6 Summary

In this chapter we have reviewed basic image processing and linear operators. These operators may affect pixels individually, or combine pixels using convolution and Fourier analysis. Filtering and frequency space are essential to the understanding of digital sampling theory. As we have seen in the convolution theorem, linear filtering and Fourier analysis are tightly linked, with profound implications in the understanding and manipulation of digital images. Aliasing and other sampling errors are directly related to the properties of the Fourier transform and to the behavior of filters and input signals when discretized. These concepts are borrowed from classical signal processing, and are widely used within the scanners that form the images to begin with. Adapted to the multidimensionality of image space, linear operators form the foundation upon which the higher levels of image analysis discussed in the remainder of this book are built.

References

- [1] Castleman, K. *Digital Image Processing*. Englewood Cliffs, NJ, Prentice Hall, 1996.
- [2] Oppenheim, A., and A. Willsky. *Signals and Systems*, (2nd Edition). Englewood Cliffs, NJ, Prentice Hall, 1996.