



***LECTURES ON
MEDICAL IMAGING***



Ge Wang, Ph.D.
Rensselaer Polytechnic Institute
Troy, New York, USA
2018

LECTURES ON MEDICAL IMAGING

FOR BIOMEDICAL ENGINEERING STUDENTS

GE WANG, PH.D.

Copyright © 2018

**Teaching Students Everywhere –
What a Delight!**

Mencius

得天下英才而教育之，三樂也。

孟子

**Teaching and Learning
Mutually Promoting.**

教学相长。

No End to Learning.

学无止境。

PREFACE

Medical imaging is an amazing field. It is not only a required course for our biomedical engineering students but also invaluable knowledge for your and your family members' healthcare. When you visit a hospital or a clinic, it is quite likely that some form(s) of medical imaging will be involved. Medical imaging is a highly interdisciplinary field and a multi-billion dollar business. Several Nobel prizes and a large job market are associated with medical imaging. Learning, teaching and researching in this field is a blessing.

This book is intended as a state of the art yet concise introduction of medical imaging, from the mathematical and physical foundation to individual and multi-modality tomographic imaging. The first part of this book is on the mathematical foundation. Then, each of the major medical imaging modalities will be described, along with modality-specific imaging principles. Multi-modality imaging and machine learning will be also explained.

Over the past five years, I have been teaching medical imaging in Department of Biomedical Engineering at undergraduate and graduate levels respectively. This has been a challenging yet rewarding experience. As a Chinese saying goes, teaching and learning are mutually promoting. This book is intended to address major challenges in teaching medical imaging, and reflects the current status of the course, especially so for undergraduate teaching by excluding the sections with “*”.

The first challenge is that students came to the class with diverse educational background, and were required to learn about various tomographic modalities that are quite complicated and highly interdisciplinary. All the involved imaging modalities are based on an elegant mathematical foundation which consists of linear system theory, Fourier analysis, and signal processing. While some of the students already have a good understanding of these subjects, a majority of them do not have clear concepts on linearity, convolution, continuous and discrete Fourier transforms, Nyquist sampling rate, and so on. A concise summary of key formulas is often given as an appendix to a medical imaging textbook but it is too dense to be digested. On the other hand, there are excellent books on these topics but they are for independent courses, and too much for this student population. Current practice for our course is to teach students about the needed mathematical knowledge over a quarter of the semester. PPTs are used with the teaching materials developed or adapted from different sources. However, in this way the information is not presented in a unified format, and the text in PPTs may not be always sufficiently clear. This situation motivated me to write the first part of the book to fill in the gap between a dry appendix and a rich textbook.

The second challenge is that tomographic imaging modalities are under rapid development. How to present them effectively within one semester requires dynamic optimization. Basically, I focus on key concepts, deep insights, and basic skills so that students are well prepared to master each of the major

tomographic modalities with confidence. Furthermore, combinations of imaging modalities are underlined in this course. The future of imaging I see is what we call “omni-tomography” to collect all kinds of data simultaneously within a single gantry for physical couplings, information synergies, end-to-end workflows, and convergence of prevention, diagnosis, and intervention.

The third challenge is that artificial intelligence / machine learning has recently attracted a major attention in many fields including tomographic imaging. I believe that this is a paradigm shift and will have a fundamental and lasting impact on the field of medical imaging. Hence, it is worthwhile incorporating machine learning ideas and data-driven imaging methods in this course.

Currently, there is no textbook that addresses the above challenges and fits for medical imaging teaching. I feel responsible and privileged to share my understanding and experience with this version-controlled “living” textbook, which will be made publicly available on Internet, along with my teaching PPTs and videos. My plan is to finish the foundation part first, then deal with each major tomographic modality, and finally discuss multimodality imaging and omni-tomography.

The book is based on my class videos recorded by MultiMedia Services, Rensselaer Polytechnic Institute. MATLAB is used to enhance the learning experience. Many IT experts, colleagues and students have, in different ways, helped facilitate or improve the development of this book, especially John Klucina

and Michael Juneau for IT assistance, Brad Osgood for permission to use some of his teaching notes for the class EE261 “Fourier Transform and its Applications”, and Joshua Goldwag for his transcribing the lecture videos and editorial assistance. If you find any typos or have any critiques or suggestions, please feel free to let me know. I am committed to refining this book promptly and improving the teaching outcomes constantly.

Ge Wang, PhD

Clark & Crossan Endowed Chair Professor

Director, Biomedical Imaging Center, CBIS/BME/SoE

Rensselaer Polytechnic Institute

110 8th Street, Troy, New York 12180, USA

ge-wang@ieee.org; <http://biotech.rpi.edu/centers/bic>

TABLE OF CONTENTS

Preface	3
Table of Contents.....	7
Chapter 1. System.....	9
Section 1.1. Linear System	9
Section 1.2. Additivity & Homogeneity.....	14
Section 1.3. Non-linear System	17
Section 1.4. Remarks	19
Chapter 2. Convolution	23
Section 2.1. Shift-invariant Linear System.....	23
Section 2.2. Continuous Convolution.....	26
Section 2.3. Discrete Convolution	30
Section 2.4. Remarks	34
Chapter 3. Fourier Series	40
Section 3.1. High Dimensional Space.....	40
Section 3.2. Fourier Series in Real Form.....	46
Section 3.3. Fourier Series in Complex Form.....	50
Section 3.4. Remarks	60
Chapter 4. Fourier Transform.....	64
Section 4.1. From Series to Transform	64
Section 4.2. Fourier Transform Properties.....	67
Section 4.3. High-dimensional Extension	72
Section 4.4. Remarks	77
Chapter 5. Signal Processing.....	81
Section 5.1. Series of Delta Functions	81
Section 5.2. Sampling Theorem	85
Section 5.3. Discrete Fourier Transform.....	88
Section 5.4. Remarks.....	94
Chapter 6. Network.....	102
Section 6.1. Electrical Network	102
Section 6.2. Exemplary Circuits.....	112
Section 6.3. Neural Network.....	118
Section 6.4. Remarks.....	124

Chapter 7. Image Quality.....	126
Section 7.1. General Measures.....	126
Section 7.2. System-specific Indices	135
Section 7.3. Task-specific Performance.....	142
Section 7.4. Remarks.....	149
About the Author.....	155

CHAPTER 1. SYSTEM

The **system** is one of the most fundamental concepts in many fields of science and engineering. A most important special case of the system is the **linear system**, especially the **shift-invariant linear system**. Based on the teaching experience with biomedical engineering students, here we discuss the key definitions and some subtle issues involving the linear system. Also, we touch upon the **nonlinear system** and associated **chaotic** behaviors.

SECTION 1.1. LINEAR SYSTEM

The concept of the **linear system** is simple and yet widely used (Oppenheim, Willsky et al. 1997). As stated in a popular medical imaging textbook (Suetens 2009):

Linear System: *“A system is linear if the superposition principle holds, that is, $L\{c_1s_1 + c_2s_2\} = c_1L\{s_1\} + c_2L\{s_2\}$, $\forall c_1, c_2 \in R$, with s_1 and s_2 as arbitrary signals.” (Suetens 2009).*

On Wikipedia, the definition of the linear system is consistent with the previous definition: *“Given two valid inputs $x_1(t)$, $x_2(t)$ as well as their respective outputs $y_1 = H\{x_1(t)\}$, $y_2 = H\{x_2(t)\}$, then a linear system must satisfy $\alpha y_1 + \beta y_2 = H\{\alpha x_1(t) + \beta x_2(t)\}$ for any scalar values α*

and β " (Wikipedia-Linear-System 2018), which is referred to as the superposition property.

The **superposition** property

$$\alpha y_1 + \beta y_2 = H \{ \alpha x_1(t) + \beta x_2(t) \} \quad (1.1.1)$$

implies both **additivity**

$$y_1 + y_2 = H \{ x_1(t) + x_2(t) \} \quad (1.1.2)$$

and **homogeneity**

$$\alpha y = H \{ \alpha x(t) \}. \quad (1.1.3)$$

The homogeneity property is also known as the **scaling** property. Likewise, the existence of additivity and homogeneity implies the superposition property.

Of a special note, a **scalar field** for a linear system can be different case by case. *"More generally, vector spaces can be defined in which the scalars are elements of an arbitrary field"* (Gallager 2006). On Wikipedia, it reads that *"The scalars can be taken from any field, including the rational, algebraic, real, and complex numbers, as well as finite fields"* (Wikipedia-Scalar 2018).

In the classic book *"Feynman Lectures on Physics"* (Feynman, Leighton et al. 2011), the linear system was presented in the same way. Feynman started with a second order differential

equation for oscillating systems, and used an operator $\underline{L}(x)$ to simplify the expression. Then, he formulated the additivity property $\underline{L}(x+y) = \underline{L}(x) + \underline{L}(y)$ and the homogeneity property $\underline{L}(\alpha x) = \alpha \underline{L}(x)$ in Eqs. (25.3) and (25.4) respectively. It appears that Feynman felt that the additivity would be more informative than the homogeneity, as he stated “(25.3) and (25.4) [were] very closely related, because if we put $x+x$ into (25.3), this is the same as setting $\alpha = 2$ in (25.4), and so on.”

A closely related concept is the **linear function** (Suetens 2009). It is well known that a polynomial of degree 1 is called linear, with a general form

$$f(x) = mx + b \quad (1.1.4)$$

where m is a **slope**, and b is an **intercept**. However, since the intercept can be a non-zero number, a linear function with a nontrivial intercept satisfies neither additivity nor homogeneity. Thus, there is a clear distinction between a linear system and a linear function, with the former being a special case of the latter.

However, the distinction between a linear system and a linear function seem not intrinsic. Taking $f(x) = mx + b$ as an example, in one coordinate system you may happen to have $b = 0$ while in another coordinate system you may see $b \neq 0$. By the definition of the linear system, $f(x) = mx + b$ represents a linear system in the former case but it does not in the latter case. If we want to define the linearity of a system

as an intrinsic property, it should not depend on the selection of a coordinate system; in other words, it is reasonable to require that the definition of a concept be independent of the selection of a coordinate system in which an observer thinks or works. Note that relativity theory was, in a good sense, established based on the equivalence of coordinate systems (Einstein 2005), and the requirement here for the concept invariability with respect to a coordinate system is in the same spirit.

As an exercise of implementing the invariability of this type, let us re-define the linear system for the frame irrelevance. *The apparent discrepancy between the linear system and linear function concepts can be eliminated by considering the system responses in terms of relative changes.* Given a status of a system in any coordinate system, let us look at changes in the input and the resultant changes in the output of the system. Then, the linear system can be re-defined as follows (existing alternative definitions of the linear system can be similarly modified).

Linear System: *Given two valid input changes $\Delta x_1(t)$, $\Delta x_2(t)$ as well as their respective outputs $\Delta y_1 = H \{ \Delta x_1(t) \}$, $\Delta y_2 = H \{ \Delta x_2(t) \}$, a linear system must satisfy*

$$\alpha \Delta y_1 + \beta \Delta y_2 = H \{ \alpha \Delta x_1(t) + \beta \Delta x_2(t) \} \quad (1.1.5)$$

for any allowed scalar values α and β . The **relative superposition** property implies both the **relative**

additivity and **relative homogeneity** or **relative scaling** properties

$$\Delta y_1 + \Delta y_2 = H \{ \Delta x_1(t) + \Delta x_2(t) \} \text{ and} \quad (1.1.6)$$

$$\alpha \Delta y = H \{ \alpha \Delta x(t) \}, \quad (1.1.7)$$

respectively.

In light of the relative superposition principle, it becomes immediately clear that the linear function is just a disguised linear system. For example, $f(x) = mx + b$ and $f(x) = mx$ are essentially the same, as long as an appropriate coordinate system is chosen so that $b = 0$ in $f(x) = mx + b$.

In a more complicated system, the concepts of **zero-state** and **zero-input** responses are involved (Sontag 1998). Let us consider a **circuit** (more details in Chapter 6 on Network) with an input $x(t)$ and an output $y(t)$. Generally speaking, the output $y(t)$ can be decomposed into the two parts. The first part is the response purely due to the input without any contribution from the initial condition of the circuit, which is called the zero-state response. The second part is the response purely due to the initial condition without any contribution from the input, which is called the zero-input response. In this case, the initial condition is the energy initially stored in some elements of the circuit, such as capacitors and inductors, which steers the behavior of the circuit. Hence, the zero-input response plays a similar role as b in $f(x) = mx + b$. In contrast to the case of $f(x) = mx + b$,

the zero-input response is normally time-varying. Therefore, a time-varying translation of a coordinate system must be used to remove the offset and have the zero-input response.

It may appear trivial having shifted our viewpoint from looking at the global variation to focusing on the relative change of the system response, but our relative viewpoint has a conceptual merit; that is, a well-posed definition should be irrelevant to the coordinate system associated with which the concept is defined, since the selection of a coordinate system is, to a good degree, arbitrary. Furthermore, it might be inspiring to mention Einstein's equivalence principle: a local gravitational force due to a mass is the same as the pseudo-force one experiences in an accelerated frame of reference (Einstein 2005).

SECTION 1.2. ADDITIVITY & HOMOGENEITY

It is very interesting to study the connection of homogeneity and additivity. As mentioned earlier, superposition implies both additivity and homogeneity. On the other hand, the combination of additivity and homogeneity immediately generates superposition. Now, two interesting questions can be asked: under what condition, additivity and homogeneity are equivalent? Are additivity and homogeneity independent each other?

First, in a simplest case of a continuous function $y = f(x)$, additivity and homogeneity are equivalent.

Let us now show that additivity $f(x_1 + x_2) = f(x_1) + f(x_2)$ implies homogeneity $f(\alpha x) = \alpha f(x)$. It is trivial to have homogeneity for a rational scalar. Indeed, $f(nx) = nf(x)$ by the same reason as what Feynman mentioned before, and in an alternative form we have $f(x) = mf\left(\frac{x}{m}\right)$ which is

$$f\left(\frac{x}{m}\right) = \frac{1}{m} f(x), \text{ for positive integers } n \text{ and } m. \text{ Hence,}$$

homogeneity holds for all positive rational scalars. Because $f(0) = f(x) + f(-x) = 0$ and $f(-x) = -f(x)$, the scalar can be any rational number, be it positive, zero, or negative. This proof is well known (Signal-Processing-Stack-Exchange 2018). To show that homogeneity holds for an irrational scalar, two somewhat cumbersome proofs were given on that website. It was pointed out by Antonio in an online comment (Signal-Processing-Stack-Exchange 2018) that to establish homogeneity for an irrational scalar we “*just need to take into account that there is a rational sequence that converges to that irrational and continuity of f* ”. Indeed, with such a limiting process assuming continuity, homogeneity must be valid for a real scalar.

On the other hand, homogeneity $f(\alpha x) = \alpha f(x)$ means additivity $f(x_1 + x_2) = f(x_1) + f(x_2)$ in this case. Without loss of generality, we set $x_1 = \alpha_1 x$ and $x_2 = \alpha_2 x$ for a given $x \neq 0$. Then, $f(x_1 + x_2) = f(\alpha_1 x + \alpha_2 x) = (\alpha_1 + \alpha_2)f(x)$ by homogeneity. By homogeneity again, we have

$(\alpha_1 + \alpha_2)f(x) = \alpha_1 f(x) + \alpha_2 f(x) = f(\alpha_1 x) + f(\alpha_2 x)$. By definition, we have $f(\alpha_1 x) + f(\alpha_2 x) = f(x_1) + f(x_2)$. That is to say $f(x_1 + x_2) = f(x_1) + f(x_2)$.

However, in a general case, additivity and homogeneity are independent. To show their independence, we need two counter-examples having one property without the other. Given the equivalence between additivity and homogeneity in the above common situation, it is a little bit tricky to find counter-examples. Fortunately, the work can be simplified when we choose a scalar field different from the real domain.

Let us show that additivity can exist without homogeneity with a well-known counter-example (Signal-Processing-Stack-Exchange 2018). Here a complex-valued system is used to transform a complex input into its conjugate with the complex scalar field. This system is clearly additive. However, the system is not homogeneous for a complex scalar. Specifically, $f(\alpha x) = \{\alpha x\}^* = \alpha^* x^* \neq \alpha x^* = \alpha f(x)$ unless the scalar is real or the variable is zero.

Furthermore, we can give counter-examples where homogeneity exists without additivity. Let us define a real-valued function as follows: $f(x) = m_1 x$ when x is rational, $f(x) = m_2 x$ when x is irrational, m_1 and m_2 are rational but $m_1 \neq m_2$, and with a rational scalar domain. Clearly, the homogeneity property is still present, since the rationality/irrationality of a number will not be changed by a

rational scaling (multiplication) operation. However, for two irrational numbers x_1 and x_2 that have a rational sum, $f(x_1 + x_2) = m_1(x_1 + x_2) \neq m_2(x_1 + x_2) = f(x_1) + f(x_2)$, directly violating additivity.

There are many other counter-examples to show the independence between additivity and homogeneity. Nevertheless, to the best of my knowledge, it still remains a challenge to show their independence for a real scalar. It has been proved above that when a function is continuous, additivity and homogeneity are equivalent. How should we break the equivalence with a discontinuous function and real scalar? You can try out and let me know.

SECTION 1.3. NON-LINEAR SYSTEM

A **non-linear system** is any system that is not a linear system. Just like a non-linear function is any function but not a linear function. In practice, linear systems/functions are typically approximations, and there are many more non-linear systems/functions than linear systems (After all, space-time is curved, not uniform/linear).

Nonlinear systems are generally difficult to analyze but often times they can be numerically solved using high-performance computing techniques. New phenomena are inherent in many nonlinear systems, such as chaos (random behaviors by deterministic systems), solitons (stable features emerged in complex processes), and singularities (numerical divergence

of differential equations), which can be counterintuitive or unpredictable, and hence very interesting!

Let us look at a common example of chaos. We assume that a system status (a number in the simplest case) will evolve in discrete steps according to $x_{n+1} = rx_n(1-x_n)$ where x is a normalized population of a certain type of animals in the range $(0,1)$, r is the growth rate of the animal, and n is the index for the time step in a chosen unit (such as a year). When x is small, the population will basically grow in a linear fashion with a slope r . This growth rate will be dramatically reduced to $r(1-x)$ when x is close to the upper bound of x , which is one. In the case of small x , the population growth is biologically limited. On the other hand, for a large x , the growth is resource-limited. As a result, the growth curve as a function of time is non-linear. Since the non-linear relationship is just a bit more complicated than a linear function (i.e., the additional factor $(1-x)$), a simple solution or structure of solutions may be expected. However, this population updating formula turned out to be a chaotic system, since when slope r is large the number of animals in the future years will not be predictable!

To understand the above paragraph visually, we can run the MatLab code that implements the aforementioned population updating formula (ShareTechNote 2018). The result is in Figure 1.1-1, showing how a random behavior emerges from a deterministic system. This phenomenon is representative of a chaotic behavior.

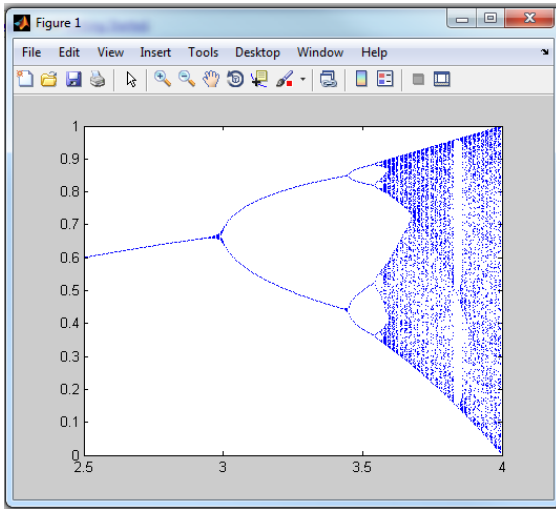


Figure 1.1-1. When the growth rate is large, the number of animals in the next year becomes random (From (ShareTechNote 2018)).

SECTION 1.4. REMARKS

In an earlier example, a complex scalar was used. Geometrically speaking, a complex scaling operation modifies both the magnitude and direction of a vector in the complex plane. This suggests a more general interpretation of a scalar as a linear system. In other words, a linear transformation has a similar effect on a vector in the Hilbert space, and can be treated as a scalar. Such a generalization can be easily made in a vector space (Gallager 2006). There are multiple definitions of a vector space equivalent to that in (Gallager 2006). As a convenient example, Wikipedia states that *“a vector space is a mathematical structure formed by a collection of elements*

called vectors, which may be added together and multiplied ('scaled') by numbers, called scalars in this context. Scalars are often taken to be real numbers, but there are also vector spaces with scalar multiplication by complex numbers, rational numbers, or generally any field. The operations of vector addition and scalar multiplication must satisfy certain requirements, called axioms" (Wikipedia-Vector-Space 2018). A simple yet general example of the linear system is a unitary transform in a vector space.

The importance of the linear system concept can be appreciated from several perspectives. In the book "*Feynman Lectures on Physics*" (Feynman, Leighton et al. 2011), Feynman gave three arguments for the linear system. First, he wrote, "*the answer is simple: because we can solve them!*" He continued, "*second (and most important), it turns out that the fundamental laws of physics are often linear. The Maxwell equations for the laws of electricity are linear, for example. The great laws of quantum mechanics turn out, so far as we know, to be linear equations. That is why we spend so much time on linear equations: because if we understand linear equations, we are ready, in principle, to understand a lot of things.*" Then, he made the final argument that "*when displacements are small, many functions can be approximated linearly.*" A picky comment could be made on his last point. Taking Ohm's law as an example, $I = V / R$ where I , V and R denote current, voltage and resistance respectively. By definition, Ohm's law is a linear system. However, if V is too large, the resistor will be burned out, which would follow a non-linear relationship; and on the other hand if V is too small, random motions of

free electrons would form a tiny current that is impossible to predict deterministically. Thus, the relationship between I and V is linear only in a range when V is neither too small nor too large. The statement that the smaller the magnitude the better the linearity is only a mathematical illusion.

At his time, Feynman's impression was that "linear equations are important. In fact they are so important that perhaps fifty percent of the time we are solving linear equations in physics and in engineering" (Feynman, Leighton et al. 2011). If Feynman had not passed away, he would most likely not modify this estimate much, as there are competing factors that have kept a balance between linear and nonlinear systems in science and engineering.

On one hand, with rapid development of high-performance computing techniques, many nonlinear systems can be handled despite the fact that most large-scale nonlinear computational tasks involve linear system solutions as intermediate steps (for example, with gradient search). More importantly, while most tasks encountered decades ago are forward problems, many current investigations are inverse problems. Even if a forward model is linear, its inverse solutions are often non-linear. Even if a generic solution is linear, the penalized versions (more details when we discuss iterative algorithms for tomographic reconstruction) are typically non-linear (Eldar and Kutyniok 2012).

On the other hand, with explosive expansion of datasets, dimensionality reduction is being actively studied to offer revolutionary tools. Amazingly, various locally linear

embedding methods seem so powerful that can unravel non-linear manifolds into low dimensional representations using linear algebraic algorithms (Roweis and Saul 2000, Donoho and Grimes 2003). Thus, linear system methods still play a key role in non-linear big data problems.

With the emergence of deep learning techniques, we again see this “fifty-percent” landscape. As a basic building block, neurons are half linear (inner product) and half non-linear (activation). For more details, see Chapter 6, (Wang 2016) and (Wang, Kalra et al. 2017).

CHAPTER 2. CONVOLUTION

In the linear system category, the most important subset of linear systems is the **shift-invariant linear system**, closely related to tomographic imaging. This chapter focuses on shift-invariant linear systems. As shown below, for a given characteristics of a shift-invariant linear system, the output of the system is completely determined by the input to the system and the **impulse response** or **point spread function (PSF)** of the system. Mathematically, the output is equal to the **convolution** of the input and the impulse response. Hence, the convolution plays the central role in analysis of shift-invariant linear systems.

SECTION 2.1. SHIFT-INVARIANT LINEAR SYSTEM

The **shift-invariability** concerns the input-output relationship of the shift-invariant system, and allows the input-output correspondence to remain the same after any shift in a relevant domain such as space and time. The shift-invariability greatly simplifies the system analysis, synthesis, and control. Formally, we have the following definition:

***Shift-invariant Linear System:** A linear system when the input to the system is shifted by a certain amount, the output of the system will be shifted by the same amount.*

An example of a shift-invariant system is rippling in a pond. Wherever a water drop falls a ripple is formed in the same waveform around the location. To a good approximation,

ripples from multiple water drops are linearly combined to give an overall texture. For this system, water drops are the input, the superimposed ripples are the output, and the linear system is location-invariant in the sense that the shape of any ripple is the same but the location of the ripple is where the water drop falls that produces the ripple. As another example, taking a photo of yourself with your smart phone. Wherever and whenever you do, you record an image of the same handsome/beautiful individual. In this case, yourself is the input, your digital picture is the output, and the smart phone as an imaging system is both location-invariant and time-invariant.

For a shift-invariant linear system, its response to a “point” or an “impulse” (a very short input with a unit “strength”; to be explained more below) is characteristic. Roughly speaking, if you know the **impulse response** or **point spread function** of a shift-invariant linear system, you can accurately infer the system response to any input function that can be represented as a number of impulses. Practically, any input function can be decomposed into a combination of impulses. Hence, the system behavior is completely determined by this impulse response or point spread function.

An impulse is an important concept in the study of dynamics. According to Newton's second law of motion, $F = ma$, where F is the amount of force used to accelerate an object, m is the mass of the object, and a is the acceleration. Since acceleration is the change of velocity over time, one can state

$F = m \frac{\Delta v}{\Delta t}$, and $F(t)\Delta t = m\Delta v(t)$. Hence, as long as the area under the curve $F(t)$ is the same, the net velocity change of the object will be the same. That is, the physical effect will be the same for any one of various impulses that all have the same area under the curve $F(t)$. In the limiting case, the time period of active forcing becomes infinitesimally small, causing the profile $F(t)$ extremely tall and narrow. This results in the generation of the well-known **Dirac delta function**.

Without loss of generality, the delta function can be defined as the limit of a series of rectangular functions

$$d_{\tau}(t) = \begin{cases} 1/(2\tau), & -\tau < t < \tau; \\ 0, & \text{otherwise.} \end{cases} \quad (2.1.1)$$

Let the parameter τ approach zero, we have

$$\lim_{\tau \rightarrow 0} d_{\tau}(t \neq 0) = 0, \quad (2.1.2)$$

$$\delta(t) = \lim_{\tau \rightarrow 0} \int_{-\infty}^{\infty} d_{\tau}(t) dt = 1.$$

This function is unconventional, and is truly a conceptual breakthrough relative to the conventional concept of a mathematical function which represents a one-to-one/many correspondence! Such a generalized function is defined in terms of distribution (i.e., the profile of $d_{\tau}(t)$ in the limiting case) and measurement (i.e., the integral operator is a measurement process, and as long as the result from the measurement is the same, who cares about any difference in

the functional shape/distribution that cannot be measured at all!). Indeed, the generalized function disrupts the tradition that a function is well defined everywhere, since although it equals zero everywhere but it is not defined at its singular point. It is infinitely tall at this singularity, and it has a unit area over the singularity.

This delta function has the following shift property:

$$\begin{aligned} \delta(t-t_0) &= 0 \text{ for } t \neq t_0, \\ \int_{-\infty}^{\infty} \delta(t-t_0) dt &= 1. \end{aligned} \tag{2.1.3}$$

SECTION 2.2. CONTINUOUS CONVOLUTION

The product of the delta function and a continuous function f can be measured to give a **sampling** result:

$$\begin{aligned} \int_{-\infty}^{\infty} \delta(t-t_0) f(t) dt &= \lim_{\tau \rightarrow 0} \int_{-\infty}^{\infty} d_{\tau}(t-t_0) f(t) dt \\ &= \lim_{\tau \rightarrow 0} \frac{1}{2\tau} \int_{t_0-\tau}^{t_0+\tau} f(t) dt \\ &= \lim_{\tau \rightarrow 0} \frac{1}{2\tau} [2\tau f(t^*)] \\ &= \lim_{\tau \rightarrow 0} f(t^*) \\ &= f(t_0). \end{aligned} \tag{2.2.1}$$

This shows that a continuous function $f(t)$ can be represented as a combination of infinitely many shifted and scaled Dirac delta functions. In other words, a continuous function can be modeled as a combination of infinitely many discrete impulses or points, or more formally:

$$f(t) = \int_{-\infty}^{\infty} \delta(\tau - t) f(\tau) d\tau. \quad (2.2.2)$$

Let us say we have a linear system, whose impulse response or point spread function is $h(t)$; that is, if the input is $\delta(t)$, the output of the shift-invariant system will be $h(t)$. Now, stop reading and consider for a while how you should compute the continuous output $g(t)$ of the system for a general continuous input function $f(t)$?

Since for the input $\delta(t)$ the output of the system is $h(t)$, we have the following inferences: for the input $\delta(t - \tau)$ the output of the system is $g(t) = h(t - \tau)$ by the shift-invariability; for the input $f(\tau)\delta(t - \tau)$ the output of the system is $g(t) = f(\tau)h(t - \tau)$ by the homogeneity property; and finally, for the input $f(t) = \int_{-\infty}^{\infty} f(\tau)\delta(t - \tau)d\tau$ the output must be $g(t) = \int_{-\infty}^{\infty} f(\tau)h(t - \tau)d\tau$ by the additivity property. This is how $g(t)$ is computed, and the involved integral is called a **convolution**. If we denote the input and output as $x(t)$ and $y(t)$ respectively, the above derivation can be visualized in the following figure.

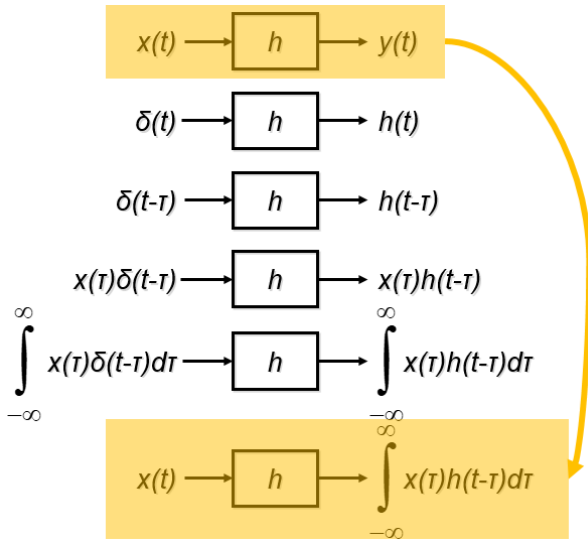


Figure 2.1-1. Derivation of the continuous convolution formula by the properties of the continuous shift-invariant linear system.

The convolution operation is often denoted by the operator “*”, and has the following properties:

Identity:

$$f(t) = \delta(t) * f(t); \quad (2.2.3)$$

Note that you can easily verify the following:

$$f(t-c) = \delta(t-c) * f(t); \text{ for a constant } c. \quad (2.2.4)$$

Commutativity:

$$g(t) = h(t) * f(t) = f(t) * h(t). \quad (2.2.5)$$

Associativity:

$$[f_1(t) * f_2(t)] * f_3(t) = f_1(t) * [f_2(t) * f_3(t)]. \quad (2.2.6)$$

Distribution:

$$f_1(t) * [f_2(t) + f_3(t)] = f_1(t) * f_2(t) + f_1(t) * f_3(t). \quad (2.2.7)$$

Note that these properties also hold for a regular multiplication operation. In a good sense (to become clearer when we learn the convolution theorem), the convolution operation is an extension of multiplication. Also, this convolution operation and all its properties can be readily extended to multi-dimensional cases.

Let us look at an example how to compute the convolution in the 1D case. Suppose that we have two Gaussian functions:

$$f_1(x) = G_1(x, \mu_1, \sigma_1) = \frac{1}{\sqrt{2\pi}\sigma_1} e^{-\frac{(x-\mu_1)^2}{2\sigma_1^2}}$$

$$f_2(x) = G_2(x, \mu_2, \sigma_2) = \frac{1}{\sqrt{2\pi}\sigma_2} e^{-\frac{(x-\mu_2)^2}{2\sigma_2^2}}$$

Then, the convolution of $f_1(x)$ and $f_2(x)$ can be directly expressed as

$$\begin{aligned} f(x) &= f_1(x) * f_2(x) \\ &= \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma_1} e^{-\frac{(\tau-\mu_1)^2}{2\sigma_1^2}} \frac{1}{\sqrt{2\pi}\sigma_2} e^{-\frac{(x-\tau-\mu_2)^2}{2\sigma_2^2}} d\tau \\ &= \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma_1 \sqrt{2\pi}\sigma_2} e^{-\frac{\sigma_2^2(\tau-\mu_1)^2 + \sigma_1^2(x-\tau-\mu_2)^2}{2\sigma_1^2\sigma_2^2}} d\tau \end{aligned}$$

Let $\sigma = \sqrt{\sigma_1^2 + \sigma_2^2}$ and $\mu = \mu_1 + \mu_2$, it can be shown through several omitted steps (you can try to fill in the gap) that

$$f(x) = f_1(x) * f_2(x) = G(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}.$$

That is, the convolution of Gaussian functions is also a Gaussian function. On the other hand, according to the identity property, the convolution of a delta function and a Gaussian function is also a Gaussian function. It suggests that the Gaussian function is the stable shape in the convolution process. Then, you may correctly guess that the convolution of an infinitely many “regular” functions will approach a Gaussian function.

SECTION 2.3. DISCRETE CONVOLUTION

Our world is both continuous and discrete, either in daily life or by physical laws. The temperature varies continuously but we typically measure it at discrete points of time. According to quantum mechanics, matter can act as both waves and particles. Up to now, we have focused on continuous functions. The counterpart of a continuous function is a discrete function. Modern computers are discrete by nature and work by digital logic. Hence, it is important to work with discrete functions.

For a discrete linear system, we denote its input by $x(n)$ and its output by $y(n)$, where n is the index for discrete points such as in space and/or time. The superposition principle or

the additivity and homogeneity properties should hold for the discrete system to be called linear. Similarly, we can define a shift-invariant linear system with an impulse response $h(n)$ when a discrete impulse $\delta(n)$ defined as

$$\delta(n) \equiv \begin{cases} 1, & n = 0; \\ 0, & n \neq 0. \end{cases} \quad (2.3.1)$$

Just like in the case of a continuous shift-invariant linear system, a key question is how to find the system output $y(n)$ from the impulse response $h(n)$ and an input $x(n)$?

Not surprisingly, the same steps used in the preceding section apply. Let us work this process through step by step. Since for the input $\delta(n)$ the output of the system is $h(n)$, for the input $\delta(n-k)$ the output of the system is $y(n) = h(n-k)$ by the shift-invariability; for the input $x(k)\delta(n-k)$ the output of the system is $y(n) = x(k)h(n-k)$ by the scaling property; and finally, for the input $x(n) = \sum_{k=-\infty}^{\infty} x(k)\delta(n-k)$ the output must be

$$y(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k) \quad (2.3.2)$$

by the additivity property.

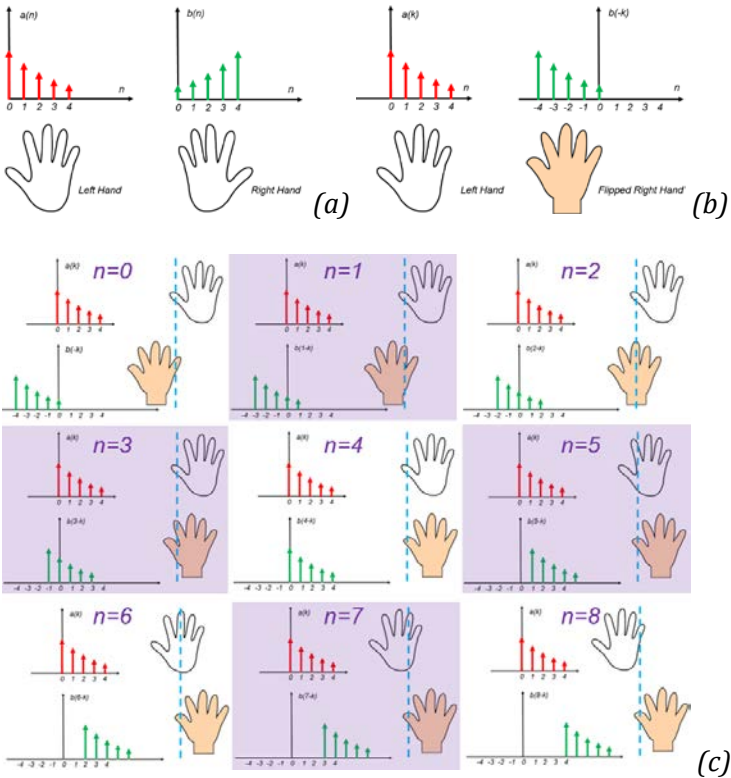


Figure 2.2-2. Example of the discrete convolution. (a) Two functions $a(n)$ and $b(n)$, (b) $b(n)$ is flipped, corresponding to “ $-k$ ” in $b(n-k)$ of the convolution formula, and (c) $b(n)$ is right shifted for 9 different n values respectively, and after shifting the corresponding elements in $a(n)$ and $b(n)$ are multiplied to obtain partial products respectively and summed together to find the convolution value for each n .

To visualize this process, let us perform a “hands-on” discrete convolution. Suppose that we have two discrete functions,

which are two sequences of data each of which contains five numbers. Let us denote them as $a(n)$ and $b(n)$ respectively, and each function is conveniently viewed as being associated with the five fingers of your left and right hands respectively. In this case, the convolution operation is defined by

$$c(n) = \sum_{k=-\infty}^{\infty} a(k)b(n-k), \text{ and visualized in Figure 2.2-2.}$$

To be more specific, let $a(n) = (5, 4, 3, 2, 1)$ and $b(n) = (1, 2, 3, 4, 5)$. MATLAB Their convolution result can be manually computed or obtained using a MATLAB code, with the result shown in Figure 2.2-3.

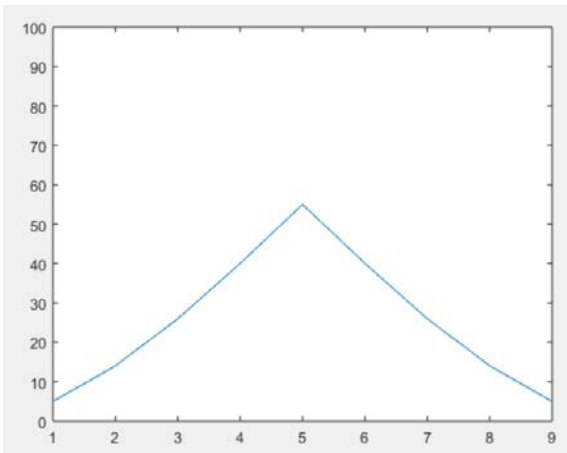


Figure 2.2-3. Discrete convolution performed using MATLAB for two discrete functions $a(n) = (5, 4, 3, 2, 1)$ and $b(n) = (1, 2, 3, 4, 5)$.

SECTION 2.4. REMARKS

For discrete shift-invariant linear systems and discrete convolutions, all the properties in the continuous case have straightforward counterparts. In particular, the discrete shift-invariant linear systems and discrete convolutions can be readily extended to 2D or 3D cases, which are most common settings for biomedical imaging. In other words, a 2D or 3D imaging system can be often viewed as a spatially invariant linear system characterized with an impulse response or point spread function $p(\vec{r})$ where the location vector \vec{r} can be 2D or 3D, and the output image $y(\vec{r})$ of the imaging system can be expressed as the convolution of the point spread function $p(\vec{r})$ and the input scene $x(\vec{r})$ to the system: $y(\vec{r}) = p(\vec{r}) * x(\vec{r})$.

In the case of 2D CT imaging, the point spread function of a CT scanner can be approximated as a Gaussian function $p(\vec{r}) = g(\vec{r}, 0, \sigma)$ (with mean zero and standard deviation σ), which can be analytically estimated or experimentally measured. An interesting question is to estimate $x(\vec{r})$ from a reconstructed $y(\vec{r})$ based on $y(\vec{r}) = g(\vec{r}, 0, \sigma) * x(\vec{r})$. This is to undo the effect of convolution, which is called a **deconvolution** or **deblurring** process. Previously, it was mentioned that convolution is a type of multiplication. Thus, deconvolution must be a type of division. This analogy will be rigorously justified with the **convolution theorem** to be learned later on. Note that when a convolution or blurring process is in 3D, deconvolution or deblurring can be done in

either 3D, 2D, or even just 1D. An example of longitudinal (1D) deconvolution is shown in Figure 2.2-4, in which longitudinal/vertical blurring was dominant, and significantly improved after deconvolution using **Wiener filtering** (Schlueter, Wang et al. 1994).

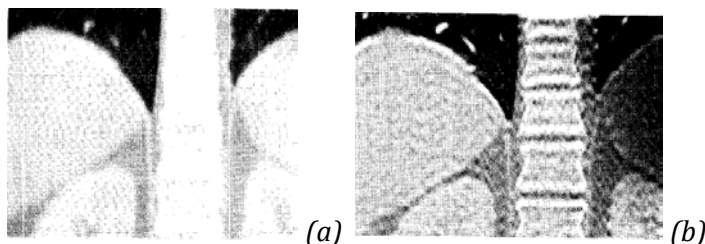


Figure 2.2-4. Example of spiral CT image deblurring. (a) A coronal slice of spiral CT with 5mm collimation and unit pitch and (b) a de-blurred image after Wiener filtering with $\alpha=1$ (Schlueter, Wang et al. 1994).

Also, we would like to mention that convolution and **cross-correlation** are closely related concepts. For two discrete functions $a(n)$ and $b(n)$, the convolution is defined by

$$c(n) = \sum_{k=-\infty}^{\infty} a(k)b(n-k),$$

" $-k$ " in $b(n-k)$ signifies that the discrete function $b(n)$ is flipped before shifting by n . Nevertheless, $b(n)$ is an arbitrary function; if we make $b(n)$ the flipped version of its original version in the first place, the flipping operation indicated by the minus sign in " $-k$ " would be unnecessary.

In this case, we have $c(n) = \sum_{k=-\infty}^{\infty} a(k)b(n+k)$. To make the difference clear between " $-k$ " and " $+k$ ", we call

$$c(n) = \sum_{k=-\infty}^{\infty} a(k)b(n+k) \quad (2.4.1)$$

the cross-correlation between discrete functions $a(n)$ and $b(n)$, instead of the convolution between them. However, there is no essential difference between convolution and cross-correlation as far as the capability of information processing is concerned.

Suppose that a discrete function $b(n)$ is of a finite length, we view its shifted version is simply another discrete function, then either the convolution or cross-correlation is in the following form:

$$c(n) = \sum_{k=-\infty}^{\infty} a(k)b_n(k), \quad (2.4.2)$$

or without loss of generality $c = \sum_{k=1}^K a(k)b(k)$ where K is a positive integer.

Recall the definition of **inner product** (if you forget it, see the next chapter). Clearly, the sum $c = \sum_{k=1}^K a(k)b(k)$ is an inner

product. For the inner product we have the well-known **Cauchy-Schwarz inequality**

$$\left(\sum_{k=1}^K a(k)b(k) \right)^2 \leq \sum_{k=1}^K a^2(k) \sum_{k=1}^K b^2(k) \quad (2.4.3)$$

while the equality holds for $a(k) = \eta b(k)$ where η is a fixed constant. This inequality can be proved as follows.

To prove the Cauchy-Schwarz inequality, let us solve for $\sum_{k=1}^K (a(k)x - b(k))^2 = \sum_{k=1}^K (a^2(k)x^2 - 2a(k)b(k)x + b^2(k)) = 0$.

We must have $4 \sum_{k=1}^K (a(k)b(k))^2 - 4 \sum_{k=1}^K a^2(k) \sum_{k=1}^K b^2(k) \leq 0$.

Clearly, $4 \sum_{k=1}^K (a(k)b(k))^2 - 4 \sum_{k=1}^K a^2(k) \sum_{k=1}^K b^2(k) = 0$ holds if $a(k) = \eta b(k)$ for a constant η . Please prove that the equality holds only if $a(k) = \eta b(k)$.

An important insight from the Cauchy-Schwarz inequality is that the cross-correlation between two discrete functions $a(n)$ and $b(n)$ will be maximized if $a(k) = \eta b_n(k)$ for a

constant η when both $\sum_{k=1}^K a^2(k)$ and $\sum_{k=1}^K b^2(k)$ are given. The

maximized response indicates a matching of the two signal shapes or a detection of a signal (for example, $b(n)$) hidden in a background function (for example, $a(n)$). In the 2D case, the cross-correlation operation can be used to extract various

local features such as edges. To detect each type of local features, such as edges at a particular orientation, we need a corresponding matching function, which is also referred to as a matching filter.

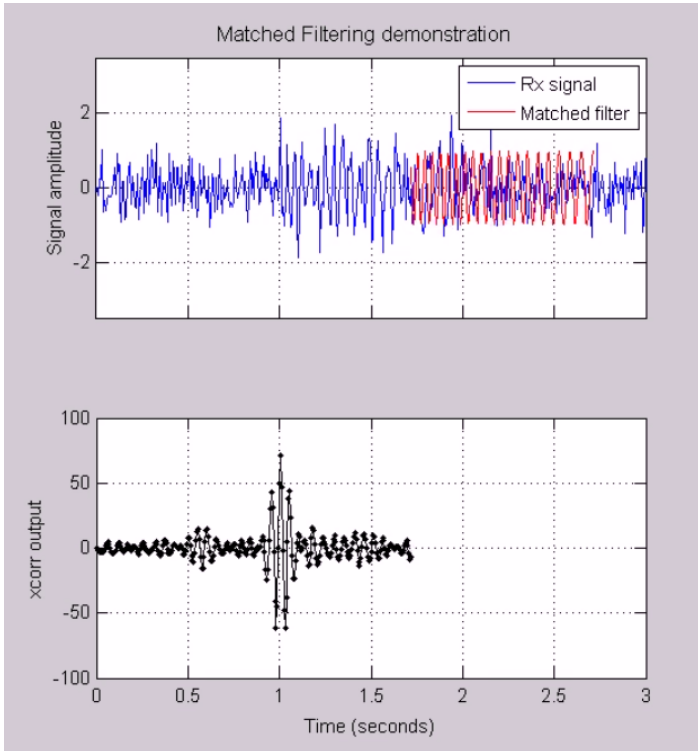


Figure 2.2-5. Cross-correlation for signal detection (Wray 2018). (a) A background signal and a target signal are colored in blue and red respectively, and (b) the cross-correlation peak indicating the location of the target signal buried in the background signal.

This utility for signal matching is shown in Figure 2.2-5 (Wray 2018). The cross-correlation operation can be extended to 2D or spaces of even higher dimensionalities. Feature detection operations, namely matching filtrations, can be performed in a fashion similar to what in Figure 2.2-5.

CHAPTER 3. FOURIER SERIES

In the previous chapter, we have shown that a continuous function $f(t)$ can be represented with infinitely many shifted

and scaled Dirac delta functions: $f(t) = \int_{-\infty}^{\infty} \delta(\tau - t) f(\tau) d\tau$.

In other words, the shifted delta functions are building blocks, which can be scaled and summed up to any continuous function. Each of the building blocks is *extremely narrow* and *highly peaked*.

In this and next chapters, we will show that a periodic or non-periodic continuous function can be represented as a **Fourier series** or **Fourier transform** respectively. In the Fourier analysis, the building blocks are sinusoidal curves with different frequencies and phases. These sinusoidal components can be also scaled and summed up to approximate a continuous or piece-wise continuous function. However, unlike delta functions, sinusoidal functions are *infinitely long* and *very smooth*.

SECTION 3.1. HIGH DIMENSIONAL SPACE

It is a mystery why we live in space and time, and why space is 3D (there are some arguments for the rationale of 3D, for example (Atkinson 2018)). From 0D to 3D, we have point, line, plane, and space respectively. In a more general / extended space R^N of **finite dimensionality** N , an arbitrary **point** is specified by a vector $\vec{A} = \{a_i\}$, $i = 1, \dots, N$. An

arbitrary **line** in the same space can be determined by a non-zero vector $\vec{B} = \{b_i\}$, $i = 1, \dots, N$, and the zero vector $\vec{O} = \{0\}$, and denoted as

$$L(k) = \{kb_i\}, k \in R \quad (3.1.1)$$

In addition to points and lines, hyperplanes and other geometric objects can be also defined in R^N . Of fundamental important are how to define the **distance** and **angulation** in R^N . Let us consider two vectors $\vec{A} = \{a_i\}$ and $\vec{B} = \{b_i\}$, $i = 1, \dots, N$.

The **distance** in R^N between these two points is defined as

$$D(\vec{A}, \vec{B}) = \sqrt{\sum_{i=1}^N (a_i - b_i)^2}, \quad (3.1.2)$$

and as a special case the distance between a vector $\vec{A} = \{a_i\}$ and the zero vector $\vec{O} = \{0\}$ is the length of \vec{A} , also referred to as its $L2$ norm, and defined as

$$\|\vec{A}\| \equiv D(\vec{A}, \vec{O}) = \sqrt{\sum_{i=1}^N a_i^2}. \quad (3.1.3)$$

A special type of **angulation** in R^N is particularly important: **orthogonality** (i.e., being perpendicular to each other). With the **inner product** operation we mentioned in the previous

chapter, the concepts of angulation and orthogonality can be defined in R^N so that geometric intuitions we have in 3D can be fully carried over to higher dimensions. Mathematically, the **inner product** of two vectors $\vec{A} = \{a_i\}$ and $\vec{B} = \{b_i\}$ are defined as

$$\vec{A} \cdot \vec{B} = \sum_{i=1}^N a_i b_i . \quad (3.1.4)$$

Next, let us see *how the inner product is intrinsically related to angulation and orthogonality.*

You can assume $N = 2$ for easy visualization. Imagine that we want to project an N -dimensional vector \vec{A} onto the line L associated with an N -dimensional vector \vec{B} as defined by Eq. (3.1.1). What do we mean by projecting an N -dimensional vector \vec{A} onto the line L ? We mean to find a point on L so that the distance between the point and the tip of the vector \vec{A} is minimized. Mathematically, this is to optimize the following functional with respect to k :

$$D^2(\vec{A}, k\vec{B}) = \sum_{i=1}^N (a_i - kb_i)^2$$

Setting the first derivative of the squared distance to zero, we immediately have $k = \frac{\vec{A} \cdot \vec{B}}{\|\vec{B}\|^2}$. Then, we can compute the

distance P from the zero vector $\vec{O} = \{0\}$ to the projected

point (i.e., the projected tip of the vector \vec{A}) determined by $k = \frac{\vec{A} \cdot \vec{B}}{\|\vec{B}\|^2}$ as $P = k \|\vec{B}\| = \frac{\vec{A} \cdot \vec{B}}{\|\vec{B}\|}$. The distance P can be

geometrically considered as the length of the vector \vec{A} (computed according to Eq. (3.1.3)) multiplied by a perceived angle θ between the vectors \vec{A} and \vec{B} . That is,

$$P = k \|\vec{B}\| = \frac{\vec{A} \cdot \vec{B}}{\|\vec{B}\|} = \|\vec{A}\| \cos \theta. \text{ In other words, the angle } \theta$$

should be defined as

$$\cos \theta = \frac{\vec{A} \cdot \vec{B}}{\|\vec{A}\| \|\vec{B}\|} \quad (3.1.5)$$

Thanks to the above geometric interpretation of the inner product, the **Euclidian geometry** can be extended from 2D/3D to high dimensionality.

A high dimensional space is also called a vector space. Several basic properties are in order:

Non-negativity:

$$\vec{A} \cdot \vec{A} \geq 0 \text{ and } \vec{A} \cdot \vec{A} = 0 \text{ if and only if } \vec{A} = \vec{0}; \quad (3.1.6)$$

Symmetry:

$$\vec{A} \cdot \vec{B} = \vec{B} \cdot \vec{A}; \quad (3.1.7)$$

Homogeneity:

$$\overline{\alpha \vec{A}} \cdot \vec{B} = \alpha (\vec{A} \cdot \vec{B}) \text{ for any scale } \alpha; \quad (3.1.8)$$

Additivity:

$$\vec{A} \cdot (\vec{B} + \vec{C}) = \vec{A} \cdot \vec{B} + \vec{A} \cdot \vec{C}. \quad (3.1.9)$$

The most natural building blocks in the space R^N is N unit vectors:

$$\vec{e}_1 = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad \vec{e}_2 = \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix}, \quad \dots \quad \vec{e}_N = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix}. \quad (3.1.10)$$

Such a set of building blocks or unit vectors is called an **orthonormal basis**, since any two different unit vectors are orthogonal as evidenced by their inner product being zero, and the length of each vector is normalized to one. Then, an arbitrary vector $\vec{A} = \{a_i\}$ in R^N can be expressed using the orthonormal basis as

$$\vec{A} = a_1 \vec{e}_1 + a_2 \vec{e}_2 + \dots + a_N \vec{e}_N \quad (3.1.11)$$

Each of the coefficients of the unit vectors can be computed with an inner product:

$$a_i = \vec{A} \cdot \vec{e}_i, \quad i = 1, 2, \dots, N. \quad (3.1.12)$$

Note that the orthonormal basis is not unique; any orthonormal basis can be rotated to have a different but equivalent presentation of any vector.

Just for fun, imagine to rotate a unit vector with respect to the origin defined by the zero vector in all possible ways in the space R^N , and the tip of the unit vector will trace a N -dimensional unit sphere

$$S \equiv \{ \vec{x} \in R^N : \|\vec{x}\| = 1 \}.$$

Note that all the spaces we have focused on so far are of finite dimensionality. However, a space can certainly have an infinite dimensionality. Why not to stretch our imagination into infinite dimensional spaces? First, we can try to dream about an infinite dimensional unit sphere. In an infinite dimensional space, point, line, plane, inner product, distance, angle, orthonormal basis, projection of one vector onto another vector, representation of a vector in terms of an orthonormal basis, and other relationships should still make sense. As an example of transition from a finite dimensional space to an infinite dimensional space, vectors $\vec{A} = \{a_i\}$ and $\vec{B} = \{b_i\}$ become functions $A(t)$ and $B(t)$, the **inner product** $\vec{A} \cdot \vec{B} = \sum_{i=1}^N a_i b_i$ become $A \cdot B = \int_{-\infty}^{\infty} A(t)B(t)dt$ (or

without loss of generality $\langle A, B \rangle = \int_0^1 A(t)B(t)dt$ over a unit interval $[0,1]$), and $A(t)$ is normalized when $\int_{-\infty}^{\infty} A^2(t)dt = 1$

(or $\int_0^1 A^2(t)dt = 1$ if $A(t)$ is defined over $[0,1]$). It should be noted that there are fundamental differences between finite and infinite dimensional spaces, just like 1 and ∞ are rather

different. At your leisure time, you can look at the so-called “Hilbert Hotel” to delve more into the use of infinity (Wikipedia-Hilbert-Hotel 2018).

SECTION 3.2. FOURIER SERIES IN REAL FORM

Since vectors can be generalized into functions, functions can be viewed as vectors in an infinite dimensional space over the whole number axis or a finite interval such as $[0, 1]$ without loss of generality. For all real-valued functions that are **square-integrable** over $[0, 1]$, which form the space $L2([0,1], \mathbb{R})$, an orthonormal basis is given as follows:

$$1, \sqrt{2} \cos(2\pi nt), \sqrt{2} \sin(2\pi nt), \quad n = 1, 2, \dots \quad (3.2.1)$$

In other words, any function $f(t)$ in this space can be, in a “good sense” (We will know more about the “good sense” in the last section, but stay assured that for the continuous parts of a function, its Fourier series expression is indeed accurate.), expressed as the following summation referred to as the **Fourier series**:

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos(2\pi nt) + \sum_{n=1}^{\infty} b_n \sin(2\pi nt) \quad (3.2.2)$$

where the use of $\frac{a_0}{2}$ is for convenience (to be seen later).

Although we focus on the representation of $f(t)$ over $[0, 1]$,

the sinusoidal terms on the right hand side of Eq. (3.2.2) work well for any real value of t . That is, the function defined over $[0, 1]$ can be extended over the whole number axis. Hence, $f(t)$ can be viewed as a periodic function for $t \in (-\infty, \infty)$, with a unit period.

The claimed orthonormality can be established by evaluating the following six inner products:

$$\begin{aligned}
 \langle 1, 1 \rangle &= 1; \\
 \langle \cos(2\pi n t), 1 \rangle &= 0, \quad n = 1, 2, \dots; \\
 \langle \sin(2\pi n t), 1 \rangle &= 0, \quad n = 1, 2, \dots; \\
 \langle \cos(2\pi m t), \sin(2\pi n t) \rangle &= 0, \quad m = 1, 2, \dots, \quad n = 1, 2, \dots; \\
 \langle \cos(2\pi m t), \cos(2\pi n t) \rangle &= \begin{cases} 0, & m \neq n, \\ \frac{1}{2}, & m = n; \end{cases} \\
 \langle \sin(2\pi m t), \sin(2\pi n t) \rangle &= \begin{cases} 0, & m \neq n, \\ \frac{1}{2}, & m = n. \end{cases} \quad (3.2.3)
 \end{aligned}$$

The evaluation is straightforward, and can be performed as your exercise.

With the representation in the form of Eq. (3.2.2) for $t \in (-\infty, \infty)$, we have three types of terms on the right hand side. The first is an overall background offset, also referred to as the **direct component**. The second part summing up cosine

components reflects the even part of $f(t)$. The third part combining sine components gives the odd part of $f(t)$. Note that the direct component is a special case of an even function, and can be grouped into the even part of the function. This is consistent to our knowledge that any function can be always decomposed into its even and odd parts:

$$f(t) = f_{\text{even}}(t) + f_{\text{odd}}(t) = \left[\frac{f(t) + f(-t)}{2} \right] + \left[\frac{f(t) - f(-t)}{2} \right].$$

For a given function $f(t)$ over $[0, 1]$, how shall we compute the coefficients a_n and b_n of its Fourier series? This can be done easily utilizing the orthonormality relationships of the sinusoidal functions. Specifically, for $n = 1, 2, \dots$ we have

$$\langle 1, f(t) \rangle = \frac{a_0}{2} = \int_0^1 f(t) dt;$$

$$2 \langle \cos(2\pi n t), f(t) \rangle = 2 \int_0^1 f(t) \cos(2\pi n t) dt = a_n;$$

$$2 \langle \sin(2\pi n t), f(t) \rangle = 2 \int_0^1 f(t) \sin(2\pi n t) dt = b_n.$$

That is,

$$\begin{aligned} a_n &= 2 \int_0^1 f(t) \cos(2\pi n t) dt, \quad n = 0, 1, \dots \\ b_n &= 2 \int_0^1 f(t) \sin(2\pi n t) dt, \quad n = 1, 2, \dots \end{aligned} \tag{3.2.4}$$

where a_0 is computed with the same formula as for $a_{n \neq 0}$, which is the convenience we mentioned earlier.

As an example, let us analytically expand the triangular function into the Fourier series. The triangular function is a function shaped like a triangle with the following formula:

$$f(t) = \begin{cases} t + 0.5, & t \in [-0.5, 0), \\ -t + 0.5, & t \in [0, 0.5]. \end{cases} \quad (3.2.5)$$

Using Eq. (3.2.4) for the coefficients a_n and b_n , the Fourier series can be written as

$$f(t) = 0.25 + \sum_{n=1}^{\infty} \frac{2}{\pi^2 (2n+1)^2} \cos[2\pi(2n+1)t].$$

For your exercise, please verify this Fourier expansion analytically.

Figure 3.2-1 helps visualize the representation of a function in its Fourier series, in which a continuous function is decomposed into a number of sinusoidal components. As your exercise, please do a similar rendering for the triangular function we just expanded.

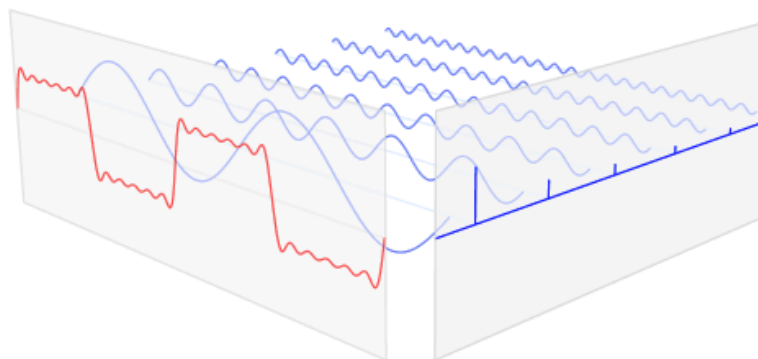


Figure 3.2-1. Decomposition of a continuous function (in red) into a number of sinusoidal components (in blue).

SECTION 3.3. FOURIER SERIES IN COMPLEX FORM

In the preceding section, we have described the Fourier series in the real form. In this section, we will convert the real form into a complex counterpart. While these two forms are mathematically equivalent, the complex form has certain merits. You will see quickly that the complex form is significantly more compact. The length of the Fourier series expression in the complex form is about 1/3 of the real counterpart. Thus, *it is often more convenient to use the complex form in derivation and analysis*. In addition to this convenience, the building block / basic unit of the complex Fourier analysis is the complex exponential function $Ae^{i\omega t}$, which plays a fundamental role in analyzing numerous

problems such as solving physical equations governing mechanical, electromagnetic, and quantum phenomena.

To have a deeper understanding of this perspective, let us briefly review **complex analysis**, a branch of mathematics dealing with complex numbers, complex variables, and complex-valued functions. In human civilization, the concept of numbers has been continuously evolving. The real number axis was established through several steps, including positive integers, zero, negative integers, rational and irrational numbers. While rational numbers are countable, irrational numbers are not. Whether it is rational or irrational, a real number always has a clear geometric meaning, such as $\sqrt{2}$ is the length of a hypotenuse of a right triangle.

However, in the real number domain there is no number whose square is -1. Thus, it appears that $\sqrt{-1}$ makes no sense at all. It is an example of divine inspiration that $i = \sqrt{-1}$ was introduced as the unit of an imaginary number (how dare you trust the absurd relationship $i = \sqrt{-1}$ in the first place!). Subsequently, conventional arithmetic operations were extended to take complex numbers. A **complex number** is expressed as $z = a + bi$, where a and b are real numbers, giving the real and imaginary parts of the complex number z respectively; i.e., $Re(z) = a$ and $Im(z) = b$. The **complex conjugate** of $z = a + bi$ is defined as $\bar{z} \equiv z^* \equiv a - bi$ where “*” denotes the **complex conjugate operation**. The two components of $z = a + bi$ can also be viewed as two coordinates of the complex number in the **complex plane**,

corresponding to the Cartesian form. The polar form of $z = a + bi$ is defined as $z = Ae^{i\theta}$, where the amplitude $A \equiv |z| \equiv \sqrt{a^2 + b^2}$ and the polar angle θ can be properly defined for $z \neq 0$.

Complex numbers can be used for addition, subtraction, multiplication and division:

$$(a + bi) \pm (c + di) = (a \pm c) + (b \pm d)i;$$

$$(a + bi)(c + di) = (ac - bd) + (bc + ad)i;$$

$$\frac{a + bi}{c + di} = \frac{a + bi}{c + di} \frac{c - di}{c - di} = \left(\frac{ac + bd}{c^2 + d^2} \right) + \left(\frac{bc - ad}{c^2 + d^2} \right) i.$$

Based on these extended rules for arithmetic operations, real functions can be generalized to complex-valued functions. For example, a Taylor expansion in the real number domain can be extended to the counterpart in the complex domain. In particular, we have the complex exponential expansion

$$e^z = \sum_{n=0}^{\infty} \frac{z^n}{n!}. \quad (3.3.1)$$

As a special case, we have

$$e^{i\theta} = \sum_{n=0}^{\infty} \frac{(i\theta)^n}{n!} = \cos \theta + i \sin \theta, \quad (3.3.2)$$

which is known as the **Euler formula**. In light of the Euler formula, $i = \sqrt{-1}$ can be viewed as a 90° -rotation operation. Indeed, the unit vector $\vec{e}_1 = (1, 0)^t$ where t indicates a **transpose operation** can be rotated by 90° twice to become $-\vec{e}_1 = (-1, 0)^t$; i.e., $i^2 = -1$ (thus, the imaginary number is not absurd at all!).

With the Euler formula, we have

$$\cos \theta = \frac{e^{i\theta} + e^{-i\theta}}{2} \quad \text{and} \quad \sin \theta = \frac{e^{i\theta} - e^{-i\theta}}{2i}. \quad (3.3.3)$$

Also of relevance to Fourier analysis in the complex form, the **inner product** operation needs to be properly defined in the complex space. An immediate consideration is that when an inner product is performed between a function/vector and itself, a real value is expected, which should be the squared length of the function/vector. Therefore, the **complex conjugate operation** is needed to produce the amplitude of a complex-valued function/vector, ignoring its phase. This requirement suggests that the **inner product** operation of two vectors in a complex space should be defined as $\vec{X} \cdot \vec{Y}^*$ due to the fact that $i^2 = -1$, instead of the simple-minded version $\vec{X} \cdot \vec{Y}$.

Similarly, the **inner product** of two complex-valued functions can be defined as

$$\langle f(t), g(t) \rangle = \int_{-\infty}^{\infty} f(t)g^*(t)dt \quad (3.3.4)$$

or in a special case over the unit interval $[0, 1]$ we have

$$\langle f(t), g(t) \rangle = \int_0^1 f(t)g^*(t)dt. \quad (3.3.5)$$

As a further argument for the complex conjugate operation in the inner product, without loss of generality let us consider two vectors each of which has only two complex components.

$$\vec{X} = \begin{pmatrix} a_1 + b_1i \\ a_2 + b_2i \end{pmatrix} \text{ and } \vec{Y} = \begin{pmatrix} c_1 + d_1i \\ c_2 + d_2i \end{pmatrix}.$$

The lengths of these two vectors are

$$|\vec{X}| = \sqrt{a_1^2 + b_1^2 + a_2^2 + b_2^2} \quad \text{and} \quad |\vec{Y}| = \sqrt{c_1^2 + d_1^2 + c_2^2 + d_2^2},$$

respectively. These two vectors can be summed to form a new vector

$$\vec{Z} = \vec{X} + \vec{Y} = \begin{pmatrix} a_1 + b_1i \\ a_2 + b_2i \end{pmatrix} + \begin{pmatrix} c_1 + d_1i \\ c_2 + d_2i \end{pmatrix} = \begin{pmatrix} a_1 + c_1 + (b_1 + d_1)i \\ a_2 + c_2 + (b_2 + d_2)i \end{pmatrix}.$$

If \vec{X} and \vec{Y} are orthogonal, the length of \vec{Z} is expected to be

$$|\vec{Z}| = \sqrt{(a_1 + c_1)^2 + (b_1 + d_1)^2 + (a_2 + c_2)^2 + (b_2 + d_2)^2}.$$

Given the assumption that \vec{X} and \vec{Y} are orthogonal, we should have $|\vec{Z}|^2 = |\vec{X}|^2 + |\vec{Y}|^2$, which is equivalent to

$$(a_1c_1) + (b_1d_1) + (a_2c_2) + (b_2d_2) = 0,$$

where again requires a complex conjugate operation in the inner product.

Note that even if we do not take the conjugate operation in the complex inner products (3.3.4) and (3.3.5), the point-wise multiplication and subsequent integration are still meaningful in terms of “measuring” one function with the other. With abuse of notation $\langle \cdot, \cdot \rangle$, we can have **un-conjugated inner products** as follows:

$$\langle f(t), g(t) \rangle = \int_{-\infty}^{\infty} f(t)g(t)dt, \text{ and} \quad (3.3.6)$$

$$\langle f(t), g(t) \rangle = \int_0^1 f(t)g(t)dt. \quad (3.3.7)$$

When functions are real, un-conjugated inner products are just the same as the previously-defined inner products. However, when functions are complex-valued, un-conjugated inner products are different from the complex inner products (3.3.4) and (3.3.5). We will use the un-conjugated inner product Eq. (3.3.6) in Section 5.4 when we discuss the Fourier transform of a series of delta functions.

Now, we are ready to produce the complex form of the Fourier series Eq. (3.2.2). According to Eq. (3.3.3), we have

$$\begin{aligned} \cos(2\pi nt) &= \frac{e^{i2\pi nt} + e^{-i2\pi nt}}{2} \text{ and} \\ \sin(2\pi nt) &= \frac{e^{i2\pi nt} - e^{-i2\pi nt}}{2i} \end{aligned} \quad (3.3.8)$$

An elegant fact is that $e^{i2\pi nt}$ for n being all integers form an orthonormal basis. That is,

$$\langle e^{i2\pi mt}, e^{i2\pi nt} \rangle = \begin{cases} 1, & m = n; \\ 0, & m \neq n. \end{cases} \quad (3.3.9)$$

This claim can be easily verified based on our previously established orthonormal basis (3.2.1). Then, we immediately have the Fourier series in the complex form:

$$f(t) = \sum_{n=-\infty}^{\infty} c_n e^{i2\pi nt}, \quad \text{and} \quad (3.3.10)$$

$$\begin{aligned} c_n &= \langle f(t), e^{i2\pi nt} \rangle = \int_0^1 e^{-i2\pi nt} f(t) dt \\ &= \int_0^1 \cos(2\pi nt) f(t) dt - i \int_0^1 \sin(2\pi nt) f(t) dt, \end{aligned}$$

and we have

$$c_n = \begin{cases} \frac{a_n}{2} - i \frac{b_n}{2}, & n \neq 0; \\ \frac{a_0}{2}, & n = 0. \end{cases} \quad (3.3.11)$$

For a real function $f(t)$, we have

$$c_n^* = \langle f(t), e^{i2\pi nt} \rangle^* = \int_0^1 f(t) e^{i2\pi nt} dt = c_{-n}. \quad (3.3.12)$$

Note that this equation shows the symmetry of the coefficients of the Fourier series in the complex domain.

As an exercise, let us expand the following function:

$$f(t) = \begin{cases} 1, & t \in [0, 0.5], \\ -1, & t \in [0.5, 1], \end{cases}$$

into the Fourier series in the complex form. Using Eq. (3.3.11), we have

$$c_n = \int_0^1 f(t) e^{-i2\pi nt} dt = \int_0^{0.5} e^{-i2\pi nt} dt - \int_{0.5}^1 e^{-i2\pi nt} dt = \begin{cases} 0, & n = 0; \\ \frac{1}{i\pi n} (1 - e^{i\pi n}), & n \neq 0. \end{cases}$$

Therefore,

$$f(t) = \sum_{n=-\infty}^{\infty} c_n e^{i2\pi nt} = \frac{4}{\pi} \sum_{n=0}^{\infty} \frac{1}{2n+1} \sin(2\pi(2n+1)t).$$

Figure 3.3-1 shows that the more the Fourier components we use, the more accurately the original function can be approached.

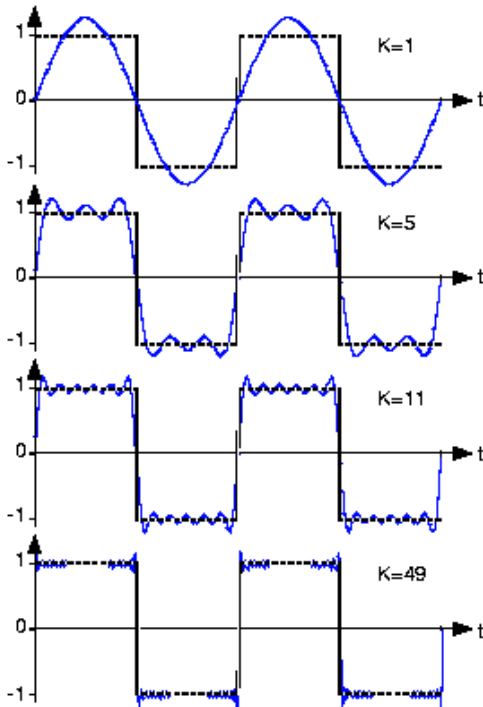


Figure 3.3-1 illustrates the fact that $f(t)$ can be approximated more and more closely with an increasing number of sinusoidal components.

If $f(t)$ is defined on a general finite interval $[a, b]$, how could we expand the function into a Fourier series in the complex form? This is actually a trivial problem. We can introduce a mapping $t = (b-a)x + a$ from $[0, 1]$ to $[a, b]$, and let $T=b-a$, then $g(x) = f(t)$ can be directly put into the

Fourier series $g(x) = \sum_{n=-\infty}^{\infty} c_n e^{i2\pi nx}$ where for any integer n ,

$$c_n = \langle g(x), e^{i2\pi nx} \rangle = \int_0^1 e^{-i2\pi nx} g(x) dx. \text{ In other words,}$$

$$f(t) = g(x) = \sum_{n=-\infty}^{\infty} c_n e^{i2\pi nx} = \sum_{n=-\infty}^{\infty} c_n e^{i2\pi n \left(\frac{t-a}{T}\right)}, \text{ that is,}$$

$$f(t) = \sum_{n=-\infty}^{\infty} c_n e^{i2\pi n \left(\frac{t}{T}\right)}, \quad t \in \left[-\frac{T}{2}, \frac{T}{2}\right], \quad (3.3.13)$$

where

$$c_n = \frac{1}{T} \int_a^b e^{-i2\pi n \left(\frac{t-a}{T}\right)} g\left(\frac{t-a}{T}\right) dt,$$

that is,

$$c_n = \frac{1}{T} \int_a^b e^{-i2\pi n \left(\frac{t-a}{T}\right)} f(t) dt. \quad (3.3.14)$$

Since the largest common period of the complex-valued orthonormal basis is T , the function $f(t)$ defined on the interval $[a, b]$ is periodically extended towards $-\infty$ and $+\infty$. Since it is periodic, we can put the origin of the coordinate system at the starting point of a period or the mid-point of the period so that the corresponding copy of $f(t)$ is defined on

$[0, T]$ and $\left[-\frac{T}{2}, \frac{T}{2}\right]$ respectively.

Although we have most focused on a function in a single interval/period, whatsoever features we have in this interval/space exist simultaneously in other

spaces/universes, which reminds us of the multiverse theory or multi-worlds interpretation (Wikipedia-Multiverse 2018).

SECTION 3.4. REMARKS

Given an orthonormal basis of a space (say, 3D), any vector in this space can be uniquely represented in terms of the unit vectors in the orthonormal basis. The coefficients for a given vector can be simply computed as the inner products of the given vector and each of the unit vectors in the orthonormal basis. As shown in Figure 3.4-1, the representation with a given orthonormal basis is unique not only in a 3D space but also in any N -dimensional space R^N .

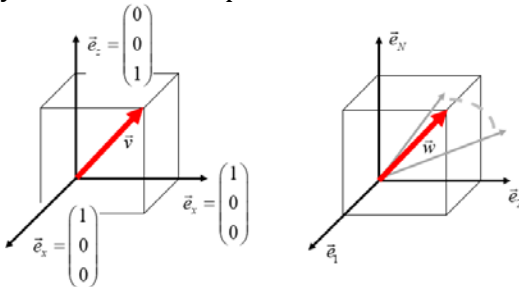


Figure 3.4-1. A vector can be uniquely represented with an orthonormal basis, and the coefficients are inner products of the vector and each member of the orthonormal basis. This way is valid in either a 3D or N -dimensional space.

In the 3D case, we have

$$\vec{v} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} = (\vec{v} \cdot \vec{e}_x) \vec{e}_x + (\vec{v} \cdot \vec{e}_y) \vec{e}_y + (\vec{v} \cdot \vec{e}_z) \vec{e}_z,$$

and in the N -dimensional space we have

$$\vec{w} = \sum_{n=1}^N (\vec{w} \cdot \vec{e}_n) \vec{e}_n. \quad (3.4.1)$$

In the case of the Fourier series representation of a function, the number of members in the sinusoidal orthonormal basis (or the complex-valued exponential orthonormal basis) is **countable**, and each member in the orthonormal basis is a continuous function that can be viewed as a vector with numerous components corresponding to the continuous change of the involved variable (**uncountable**; i.e., no way to index all the points over an interval (a, b) if $a \neq b$). The Fourier series of an arbitrary practical function (for example, square integrable and piecewise continuous) is also a generalized vector (again, with uncountably many components). Hence, *the computation of the Fourier series coefficients is nothing but performing inner products between the function to be expanded into the Fourier series and each member of the sinusoidal orthonormal basis.*

Geometrically, the inner product $c_n = \frac{1}{T} \int_0^T e^{-i2\pi n \left(\frac{t}{T}\right)} f(t) dt$ is

nothing but the projection of the function onto a member $\frac{1}{T} e^{i2\pi n \left(\frac{t}{T}\right)}$ of the orthonormal basis, as shown in Figure 3.4-2.

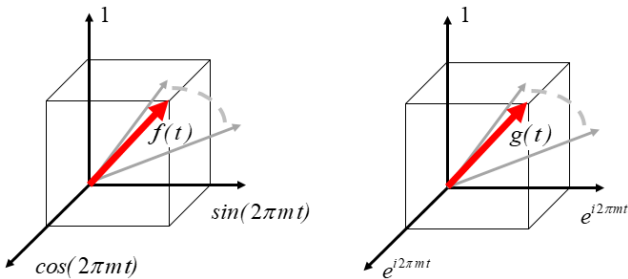


Figure 3.4-2. A function has a unique representation in an orthonormal sinusoidal or complex-valued exponential basis, with the coefficients determined by projecting the function onto each member of the orthonormal basis, which can be either a 1D sine or cosine function or even a complex-valued exponential basis function.

Whether the Fourier series of a periodic function converges to the original function is not a trivial question, and has been an important topic in the field of harmonic analysis, a branch of pure mathematics. Such convergence analysis exemplifies mathematical sophistication, involving pointwise convergence, uniform convergence, absolute convergence, and so on. Here we only mention a fundamental theorem:

Convergence Theorem: *If a function $f(t)$ is piecewise continuous, then its Fourier series at a discontinuous point t_0 will produce the average of the functional values on both sides, which is $\frac{1}{2}(f(t_0^-) + f(t_0^+))$.*

Hence, it is not surprising that the Fourier series exhibits the so-called Gibbs effect in which the convergence is pointwise

but non-uniform as evidenced by substantial ripples. The mid-point convergence and the Gibbs effects are shown in Figure 3.4-3.

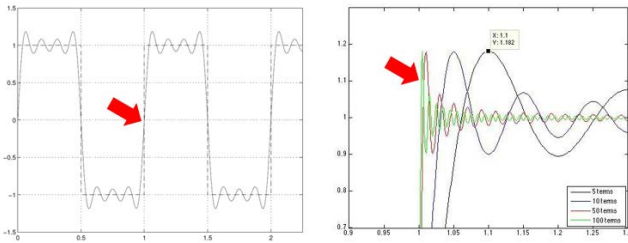


Figure 3.4-3. Convergence of the Fourier series of a piecewise continuous function. While at continuous points the values of the original function can be accurately reproduced, at each discontinuous point the Fourier series gives the average of the left and right limits of the function.

To conclude this chapter, we mention that the Fourier series can be readily extended to the cases of multi-dimensional functions, be it real-valued or complex-valued.

CHAPTER 4. FOURIER TRANSFORM

What we have learned in the preceding chapter is summarized as Eqs. (3.3.13) and (3.3.14), which is reproduced as follows:

$$f(t) = \sum_{n=-\infty}^{\infty} c_n e^{i2\pi n\left(\frac{t}{T}\right)}, \quad t \in \left[-\frac{T}{2}, \frac{T}{2}\right],$$

Where $c_n = \frac{1}{T} \int_{-T/2}^{T/2} e^{-i2\pi n\left(\frac{t}{T}\right)} f(t) dt$. This Fourier series

represents a well-behaved periodic function (such as a piecewise continuous function) with a period T as a sum of

infinitely many harmonic components $\frac{1}{T} e^{i2\pi n\left(\frac{t}{T}\right)}$. A natural

problem is how to extend this Fourier analysis method from the special case of periodic functions to the general case of non-periodic functions. The solution to this problem is the central topic of this chapter.

SECTION 4.1. FROM SERIES TO TRANSFORM

To derive the Fourier transform from the Fourier series, our basic idea is to focus on one copy of a periodic function $f(t)$

over a symmetric interval $\left[-\frac{T}{2}, \frac{T}{2}\right]$, then let T go to infinity.

Certainly, the function should be kept square-integrable over this increasingly enlarged interval so that all involved mathematical operations are meaningful. For our purpose, a

periodic function with an infinite period is nothing but a non-periodic function, since the function would be of our interest only within our reach, and we cannot go beyond $-\infty$ or $+\infty$.

Over this interval $[-\frac{T}{2}, \frac{T}{2}]$, $f(t) = \sum_{n=-\infty}^{\infty} c_n e^{i2\pi n(\frac{t}{T})}$ with

$c_n = \frac{1}{T} \int_{-T/2}^{T/2} e^{-i2\pi n(\frac{t}{T})} f(t) dt$. Inserting the coefficients into the

Fourier series, we have

$$f(t) = \sum_{n=-\infty}^{\infty} \left(\int_{-T/2}^{T/2} e^{-i2\pi(\frac{n}{T})t} f(t) dt \right) e^{i2\pi(\frac{n}{T})t} \frac{1}{T}. \quad (4.1.1)$$

The integrals on the right hand side of Eq. (4.1.1) give inner products at infinitely many discrete frequency points

$u = \frac{n}{T}$, $n = \dots, 0, 1, 2, \dots$, and for a sufficiently large T , for all

integer n the interval for u is dense on the whole number axis, and the distance between adjacent frequencies is

infinitesimally small ($\Delta u = \frac{1}{T}$). Hence, for T sufficiently

large, discrete coefficients is for the Fourier series becomes a continuous spectrum as follows:

$$F(f(t)) = \hat{f}(u) \equiv \int_{-\infty}^{\infty} e^{-i2\pi ut} f(t) dt,$$

and $f(t)$ can be recovered using the Fourier series (again in

the aforementioned good sense) as follows:

$$f(t) = F^{-1}(\hat{f}(u)) = \sum_{n=-\infty}^{\infty} \hat{f}(u) e^{i2\pi ut} \frac{1}{T} = \int_{-\infty}^{\infty} \hat{f}(u) e^{i2\pi ut} du.$$

In summary, we have the **forward and inverse Fourier transforms**:

$$\begin{cases} \hat{f}(u) = F(f(t)) = \int_{-\infty}^{\infty} e^{-i2\pi ut} f(t) dt, \\ f(t) = F^{-1}(\hat{f}(u)) = \int_{-\infty}^{\infty} e^{i2\pi ut} \hat{f}(u) du. \end{cases} \quad (4.1.2)$$

A well-behaved function in terms of t and its Fourier transform in terms of u form a **Fourier transform pair**; one can be computed from the other, denoted as $f(t) \Leftrightarrow \hat{f}(u)$.

As an example, let us compute the Fourier transform of the **rectangular function** (also known as a **gate function**):

$$\Pi(t) \equiv \begin{cases} 1, & |t| < 1/2; \\ 0, & |t| \geq 1/2. \end{cases} \quad (4.1.3)$$

The Fourier transform of the rectangular function is directly computed according to Eq. (4.1.2):

$$\hat{\Pi}(u) = \int_{-\infty}^{\infty} e^{-i2\pi ut} f(t) dt = \int_{-1/2}^{1/2} e^{-i2\pi ut} dt = \frac{\sin(\pi u)}{\pi u}.$$

In terms of a **sinc function** $\text{sinc}(x) \triangleq \frac{\sin(x)}{x}$, we have

$$\hat{\Pi}(u) = \text{sinc}(\pi u). \quad (4.1.4)$$

A **general rectangular function** is as follows:

$$\Pi_T(t) \equiv \begin{cases} 1, & |t| < T/2; \\ 0, & |t| \geq T/2. \end{cases} \quad (4.1.5)$$

As an exercise, please find the Fourier transform of $\Pi_T(t)$ directly or use the scaling property of the Fourier transform given in the following section. A few more examples are as follows:

$$e^{-\pi t^2} \Leftrightarrow e^{-\pi u^2} \quad (\text{Direct computation}), \quad (4.1.6)$$

$$\delta(t) \Leftrightarrow 1 \quad (\text{Via the Fourier transform}), \quad (4.1.7)$$

$$1 \Leftrightarrow \delta(u) \quad (\text{Via the inverse Fourier transform}). \quad (4.1.8)$$

Please read Section 5.4 for more rigorous explanation on Eqs. (4.1.7) and (4.1.8).

SECTION 4.2. FOURIER TRANSFORM PROPERTIES

While there are multiple variant definitions of the **Fourier transform**, we keep using the following forms:

$$\left\{ \begin{array}{l} \hat{f}(u) = \int_{-\infty}^{\infty} f(t) e^{-i2\pi ut} dt, \\ f(t) = \int_{-\infty}^{\infty} \hat{f}(u) e^{i2\pi ut} du. \end{array} \right. \quad (4.2.1)$$

The **Fourier transform pair** opens a door to elegant relationships. Next we present some of the basic properties, then introduce several important theorems for Fourier transform.

The following basic properties are straightforward for the Fourier Transform:

Linearity:

For two functions $f(t)$ and $g(t)$ with their Fourier transforms $\hat{f}(u)$ and $\hat{g}(u)$ respectively, if $h(t) = \alpha f(t) + \beta g(t)$, then $\hat{h}(u) = \alpha \hat{f}(u) + \beta \hat{g}(u)$ where α and β are scalars. (4.2.2)

Translation:

If $h(t) = f(t - t_0)$, then $\hat{h}(u) = e^{-i2\pi t_0 u} \hat{f}(u)$, where t_0 is a constant. (4.2.3)

Modulation:

If $h(t) = e^{-i2\pi u_0 t} f(t)$, then $\hat{h}(u) = \hat{f}(u - u_0)$, where u_0 is a constant. (4.2.4)

Scaling:

If $h(t) = f(\alpha t)$, then $\hat{h}(u) = \frac{1}{|\alpha|} \hat{f}\left(\frac{u}{|\alpha|}\right)$, where $\alpha \neq 0$ is a constant. (4.2.5)

Conjugation:

If $h(t) = f^*(t)$, then $\hat{h}(u) = \hat{f}^*(-u)$. (4.2.6)

Evenness/Oddness:

For a real function $f(t)$, if it is even, $\hat{f}(u)$ is real and even; if it is odd, $\hat{f}(u)$ is imaginary and odd. (4.2.7)

The linearity property shows that the Fourier transform can be viewed as a linear system. Would you consider this system shift-variant or not? The answer is not, as evidenced by the translation property. Interestingly, any sinusoidal function (either sine or cosine) can be obtained by translating and scaling another sinusoidal function, and all these self-similar sinusoidal components can be combined to represent almost

all functions in the Fourier series (for a periodic function) or through the Fourier transform (for a non-periodic function). In a good sense, the translation and scaling, as symmetric operations, generate everything.

Next, let us re-visit the concept of convolution by interpreting the convolution operation in the Fourier space. As it turns out, a convolution operation in a conventional space is equivalent to multiplication in the Fourier space. Mathematically, we have the following deep relationship.

Convolution Theorem:

For transform pairs $f(t) \Leftrightarrow \hat{f}(u)$ and $g(t) \Leftrightarrow \hat{g}(u)$, we have $f(t) * g(t) \Leftrightarrow \hat{f}(u) \hat{g}(u)$. (4.2.8)

Proof:

Let us start with the product in the Fourier space

$$\begin{aligned} P &= \hat{f}(u) \hat{g}(u) = \left(\int_{-\infty}^{\infty} f(s) e^{-i2\pi us} ds \right) \left(\int_{-\infty}^{\infty} g(t) e^{-i2\pi ut} dt \right) \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(s) g(t) e^{-i2\pi u(s+t)} ds dt \\ &= \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} f(s) e^{-i2\pi u(s+t)} ds \right) g(t) dt. \end{aligned}$$

Let $l = s + t$, then $dl = ds$, we have

$$P = \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} f(l-t) e^{-i2\pi ul} dl \right) g(t) dt$$

$$\begin{aligned}
&= \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} f(l-t)g(t)dt \right) e^{-i2\pi ul} dl \\
&= \int_{-\infty}^{\infty} (f(l) * g(l)) e^{-i2\pi ul} dl.
\end{aligned}$$

Note that the convolution theorem is *only* valid for Fourier transform (i.e., a convolution is a multiplication *only* in the Fourier space). Could you figure out why?

Recall that the image **deconvolution** or **deblurring** problem that was mentioned before is to estimate an image $x(\vec{r})$ from its burred version $y(\vec{r})$ based on $y(\vec{r}) = g(\vec{r}) * x(\vec{r})$, where $g(\vec{r})$ is the impulse response of an imaging system. According to the convolution theorem, now we know that the convolution or blurring in the image domain is the multiplication in the Fourier domain, and the deconvolution or deblurring in the image domain is, conceptually, the division in the Fourier space. Note that this becomes tricky when amplitudes of some Fourier components are small or zero, since divisions with small denominators are unstable. Hence, it is fair to say that the key to deconvolution or deblurring is to *regularize* this type of inverse problems with prior knowledge in the form of constraints on features of images that we expect to have.

Parseval Theorem:

With a Fourier transform pair $f(t) \Leftrightarrow \hat{f}(u)$, we have

$$\int_{-\infty}^{\infty} |f(t)|^2 dt = \int_{-\infty}^{\infty} |\hat{f}(u)|^2 du. \quad (4.2.9)$$

Proof:

Let start from the original domain:

$$\begin{aligned}
 I &= \int_{-\infty}^{\infty} |f(t)|^2 dt = \int_{-\infty}^{\infty} f(t) f^*(t) dt \\
 &= \int_{-\infty}^{\infty} f(t) \left(\int_{-\infty}^{\infty} \hat{f}(s) e^{i2\pi st} ds \right)^* dt \\
 &= \int_{-\infty}^{\infty} f(t) \left(\int_{-\infty}^{\infty} \hat{f}^*(s) e^{-i2\pi st} ds \right) dt \\
 &= \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} f(t) e^{-i2\pi st} dt \right) \hat{f}^*(s) ds = \int_{-\infty}^{\infty} \hat{f}(s) \hat{f}^*(s) ds .
 \end{aligned}$$

Given the fact that Fourier transform is a presentation of a function in the sinusoidal or complex-valued exponential orthonormal basis, Parseval's identity is not surprising at all, as it geometrically means that the norm of the function (the length of the vector, which in this case with un-countably many components) in the original space remains the same if you measure it on the orthonormal basis. In simple words, the length of a ruler will remain the same regardless which Euclidian coordinate system (rotated or not) you use.

Earlier we mentioned that amplitudes of some Fourier components are small or zero, and do you know why? According to the Parseval theorem, $\int_{-\infty}^{\infty} |\hat{f}(u)|^2 du$ is a finite number, and therefore $|\hat{f}(u)|^2$ must decay more rapidly than

$\frac{1}{|u|}$ for $|u| \rightarrow \infty$; otherwise, $\int_{-\infty}^{\infty} |\hat{f}(u)|^2 du$ will not converge. More detailed analysis shows that there is a correlation between the decay rate of the Fourier spectrum and the differentiability of the original function. The smoother a function is, the faster its Fourier spectrum decays.

SECTION 4.3. HIGH-DIMENSIONAL EXTENSION

So far all the discussions on the Fourier transform are in the 1D case, but the essential ideas and properties hold in the N -dimensional case. The extension from the 1D to N -dimensional case is not challenging. For example, the **2D Fourier transform pair** can be defined as follows:

$$\left\{ \begin{array}{l} \hat{f}(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-i2\pi(xu+yv)} dx dy, \\ f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \hat{f}(u, v) e^{i2\pi(xu+yv)} du dv. \end{array} \right. \quad (4.3.1)$$

Suppose that we have a 2D rectangular function centralized at the origin of the 2D Cartesian coordinate system:

$$f(x, y) = \begin{cases} 1, & |x| < A/2 \quad \text{and} \quad |y| < B/2; \\ 0, & \text{Otherwise,} \end{cases}$$

Its Fourier transform can be directly computed as

$$\begin{aligned}\hat{f}(u, v) &= \int_{-B/2}^{B/2} \int_{-A/2}^{A/2} e^{-i2\pi(xu+yv)} dx dy \\ &= AB \operatorname{sinc}(\pi Au) \operatorname{sinc}(\pi Bv).\end{aligned}$$

A common utility of the 2D Fourier transform is to remove noise in an image as shown in Figure 4.3-1. A noisy image is first transformed into the Fourier space, in which high frequency components are mostly noise while low and intermediate frequency components form structures of interest. A filtering mask is placed in the Fourier space that keeps low and intermediate frequency components intact and zero out all high frequency components. This filtered Fourier spectrum is then inversely transformed back to the original space, producing a smooth image without strong noise.

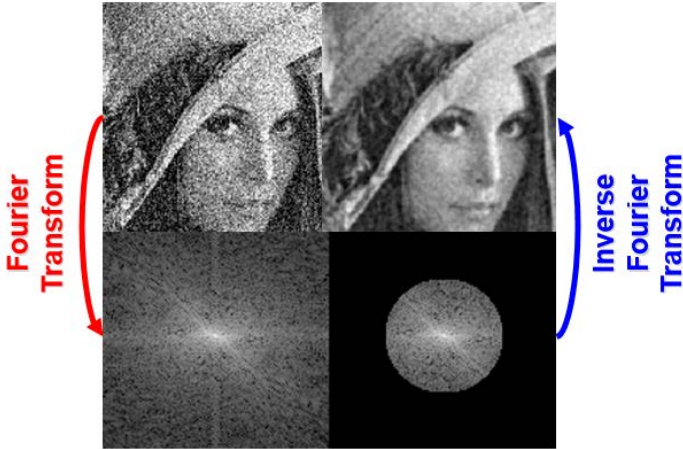


Figure 4.3-1. Image noise suppression using a Fourier transform based filtering operation.

The above 2D formulation of the Fourier transform can be generalized into **the N -dimensional Fourier transform pair** as follows:

$$\begin{cases} \hat{f}(\vec{u}) = \int_{R^N} f(\vec{r}) e^{-i2\pi(\vec{r}\cdot\vec{u})} d\vec{r}, \\ f(\vec{r}) = \int_{R^N} \hat{f}(\vec{u}) e^{i2\pi(\vec{r}\cdot\vec{u})} d\vec{u}, \end{cases} \quad (4.3.2)$$

where \vec{r} and \vec{u} are vectors denoting locations in the original and Fourier spaces respectively. The key agent here is the single variable $t = \vec{r} \cdot \vec{u}$, with which the effect of two N -dimensional vectors are summarized as a single-valued 1D function. Geometrically, \vec{u} is a principal vector representing a particular N -dimensional wave, whose direction is the

direction of \vec{u} and whose frequency is specified by the amplitude of \vec{u} in the N -dimensional space. At any location \vec{r} , the value of this wave function is totally determined by the inner product $\vec{r} \cdot \vec{u}$, which is the projection of the location vector \vec{r} onto the vector \vec{u} in the N -dimensional space. Again, the forward Fourier transform does nothing but computing the projection of an N -dimensional function $f(\vec{r})$ onto an N -dimensional complex-valued wave $e^{i2\pi(\vec{r} \cdot \vec{u})}$, a member in the N -dimensional orthonormal basis.

Rotation Properties:

For $f(\vec{r}) \Leftrightarrow \hat{f}(\vec{u})$, if $\vec{r}' = R_{\vec{\theta}} \vec{r}$ where $R_{\vec{\theta}}$ is an orthonormal rotation matrix specified by the angular vector $\vec{\theta}$, then we have $f(R_{\vec{\theta}} \vec{r}) \Leftrightarrow \hat{f}(R_{\vec{\theta}} \vec{u})$. (4.3.3)

Proof:

$$\hat{f}(\vec{u}) = \int_{R^N} f(\vec{r}) e^{-i2\pi(\vec{r} \cdot \vec{u})} d\vec{r} = \int_{R^N} f(\vec{r}) e^{-i2\pi(\vec{r}' \cdot \vec{u})} d\vec{r}.$$

$$\hat{f}(R_{\vec{\theta}} \vec{u}) = \int_{R^N} f(\vec{r}) e^{-i2\pi((\vec{r}' R_{\vec{\theta}}) \cdot \vec{u})} d\vec{r} = \int_{R^N} f(\vec{r}) e^{-i2\pi((R_{\vec{\theta}}^t \vec{r}') \cdot \vec{u})} d\vec{r}.$$

Let $\vec{s}' = R_{\vec{\theta}}^t \vec{r}'$, the superscript "t" denotes the transport operation, we have $\vec{r}' = R_{\vec{\theta}} \vec{s}'$ ($R_{\vec{\theta}}^{-1} = R_{\vec{\theta}}^t$ for the orthonormal matrix $R_{\vec{\theta}}$) and

$$\hat{f}(R_{\vec{\theta}} \vec{u}) = \int_{R^N} f(R_{\vec{\theta}} \vec{s}') e^{-i2\pi((\vec{s}') \cdot \vec{u})} d\vec{s}'.$$

Replacing \vec{s}' with \vec{r} , we have

$f(R_{\theta}\vec{r}) \Leftrightarrow \hat{f}(R_{\theta}\vec{u})$ (If you are confused, please set $N=2$ and go through the process again).

Note that the rotation property makes a sense only in the space with a dimensionality greater than 1. Actually, the rotation property is geometrically self-evident, as shown in Figure 4.3-2. Intuitively, the Fourier spectrum of an image is a linear combination of many waves whose directions and amplitudes are specified by the Fourier spectrum of the image. Hence, a rotation of the original image implies that all of its component waves (as specified by the Fourier spectrum) must be rotated by the same angle. Indeed, images are objective, and the selection of coordinate systems is subjective. Any rotated coordinate system can be used, but we must have the same intrinsic features of the images regardless of the rotation angle.

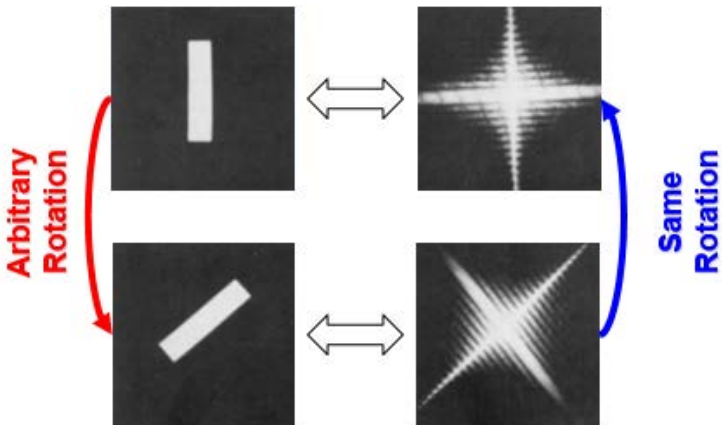


Figure 4.3-2. Rotation of an image means the same rotation of its Fourier spectrum.

SECTION 4.4. REMARKS

Since a well-behaved function is essentially equivalent to its Fourier transform, if we know one of them, we have almost all information about the other. Why then would we bother with the Fourier transform when it is essentially a redundant copy of the original function? If you read the above text carefully, you should have had a feeling about the answer.

Fourier analysis gives us a different perspective of the original function. An amazing fact as presented from this section was that a discontinuous function such as a rectangular function can be viewed as a sum of many continuous simple sinusoidal functions. On the other hand, a continuous function can be viewed as a sum of many delta functions that are extremely discontinuous. In other words, continuity and discontinuity can represent each other very well. Often times the complexity of a problem depends on perspectives. As an impressive example, the convolution theorem reveals that a complicated convolution process is a straightforward multiplication in the Fourier space. Just like we need to appreciate a sculpture from different viewing angles, whenever we have different perspectives of an important subject such as a function, our understanding of it becomes comprehensive, and our capability of analyzing it is enhanced.

Fourier analysis exemplifies the divide and conquer strategy. With the Fourier series or Fourier transform, a function is decomposed into sinusoidal building blocks. For a linear system, if we know how the system responds to any sinusoid

stimulus, we can determine the response of the system to any input that is a linear combination of sinusoidal stimuli. This is normally easier than a direct solution for the system response to a general input.

Fourier analysis is beautiful. In the Fourier basis, we see self-similarity among sinusoidal building blocks, which are related via translation and scaling operations. A function and its transform demonstrate impressive duality and conservation. As an example, if a function is well localized in one domain, its counterpart will be quite spread out in the other domain, governed by the **uncertainty principle**.

Uncertainty Theorem:

For $\int_{-\infty}^{\infty} |f(t)|^2 dt = \int_{-\infty}^{\infty} |\hat{f}(u)|^2 du = 1$, let

$$\sigma_t^2 = \int_{-\infty}^{\infty} t^2 |f(t)|^2 dt \quad \text{and} \quad \sigma_u^2 = \int_{-\infty}^{\infty} u^2 |\hat{f}(u)|^2 du = 1,$$

we have $\sigma_t \sigma_u \geq \frac{1}{4\pi}$.

As an exercise, you can try to prove the uncertainty theorem if you are curious enough. When will the product $\sigma_t \sigma_u$ reach the minimum? We learned previously in Eq. (4.1.6) that $e^{-\pi t^2} \Leftrightarrow e^{-\pi u^2}$, which is to say that

$$\frac{1}{\sqrt{2\pi\sigma_t^2}} e^{-\frac{t^2}{2\sigma_t^2}} \Leftrightarrow \frac{1}{\sqrt{2\pi\sigma_u^2}} e^{-\frac{u^2}{2\sigma_u^2}} \quad \text{and} \quad \sigma_t = \sigma_u = \frac{1}{\sqrt{2\pi}}.$$

case, we have $\sigma_t \sigma_u = \frac{1}{4\pi}$.

As another example of the uncertainty principle, if a function is a delta, its Fourier spectrum is a constant, and vice versa. Yet another example, you can also try to prove the following theorem on the Fourier transform of a function with a finite support (i.e., non-zero functional values are defined on a finite interval(s)):

Finite Support Theorem:

If both $f(t)$ and $\hat{f}(u)$ have finite support, then $f(t) = 0$ almost everywhere, which is equivalent to say that the Fourier transform of a function with a finite support must have infinite support. (4.3.4)

Fourier analysis is useful in many fields for various reasons. The ubiquity of Fourier analysis is a key feature of medical imaging. As to be learned later, computed tomography (CT) is based on the **Fourier slice theorem**, magnetic resonance imaging (MRI) takes samples in the Fourier space, and so on.

CHAPTER 5. SIGNAL PROCESSING

With either the Fourier series or the Fourier transform, we typically deal with a continuous or piecewise continuous function. In this chapter, we discuss (1) how to transform a continuous function into a **discrete function** (which is a function defined on a discrete set such as all integers), (2) under which conditions the information carried by the continuous function is fully preserved in its discrete version, (3) how to recover the continuous function from the discrete counterpart, and (4) what the **discrete Fourier transform** and **Fast Fourier Transform** are, and why we need them.

Briefly speaking, we are in the era of big data and supercomputing, and computers only take digital data and work according to digital logic. Therefore, to be digitally processed, an analog signal should be sampled at discrete points, and recorded in the binary format. Fourier analysis is both the theoretical foundation and a powerful tool for **digital signal processing**.

SECTION 5.1. SERIES OF DELTA FUNCTIONS

When we introduced the delta function, we mentioned its sampling property: $\int_{-\infty}^{\infty} \delta(t-t_0) f(t) dt = f(t_0)$. Evidently, to represent $f(t)$ we will need to sample it densely enough. Since we do not know where in the domain of the function needs to be sampled more densely, we should sample it uniformly. Mathematically, *this sampling process can be*

modeled as an integral of the product of an analog signal and a series of delta functions.

Precisely, the series of delta functions is defined as

$$s(t) = \sum_{k=-\infty}^{\infty} \delta(t-k), \quad (5.1.1)$$

or in the general case of a sampling step Δ ,

$$s_{\Delta}(t) = \sum_{k=-\infty}^{\infty} \delta(t-k\Delta). \quad (5.1.2)$$

Integrating the product of a function to be sampled and a series of delta functions yields the sampling result:

$$d(t) = \int_{-\infty}^{\infty} f(t)s_{\Delta}(t)dt = \sum_{k=-\infty}^{\infty} f(k\Delta)\delta(t-k\Delta). \quad (5.1.3)$$

Note that the above integral is an inner product between the continuous function of interest and the series of delta functions. In other words, the outcome of the measurement process is a new function $d(t) = \langle f(t), s_{\Delta}(t) \rangle$.

To see what such a sampling process means in the Fourier domain, we need to know the Fourier transform of $s(t)$ or $s_{\Delta}(t)$. Actually, we have

$$s(t) \Leftrightarrow s(u), \text{ or generally, } s_{\Delta}(t) \Leftrightarrow \frac{1}{\Delta} s_{1/\Delta}(u). \quad (5.1.4)$$

A simple but imprecise way to show the elegant symmetry $s_{\Delta}(t) \Leftrightarrow \frac{1}{\Delta} s_{1/\Delta}(u)$ is to perform the following Fourier analysis “loosely” without considering the convergence issue (in case you are curious about a more rigorous way, please read the Section 5.4).

With a Fourier series expansion, we have

$$s_{\Delta}(t) = \sum_{n=-\infty}^{\infty} \delta(t - n\Delta) = \sum_{n=-\infty}^{\infty} c_n e^{i2\pi \frac{n}{\Delta} t}, \text{ where}$$

$$c_n = \frac{1}{\Delta} \int_{-\Delta/2}^{\Delta/2} s_{\Delta}(t) e^{-i2\pi \frac{n}{\Delta} t} dt = \frac{1}{\Delta} \int_{-\Delta/2}^{\Delta/2} \delta(t) e^{-i2\pi \frac{n}{\Delta} t} dt = \frac{1}{\Delta}.$$

Hence,

$$s_{\Delta}(t) = \sum_{n=-\infty}^{\infty} \delta(t - n\Delta) = \frac{1}{\Delta} \sum_{n=-\infty}^{\infty} e^{i2\pi \frac{n}{\Delta} t} = \frac{1}{\Delta} s_{1/\Delta}(u),$$

where in the last step the modulation property Eq. (4.2.4) of the Fourier transfer is used. That is, $s_{\Delta}(t) \Leftrightarrow \frac{1}{\Delta} s_{1/\Delta}(u)$.

The series of delta functions and the convolution theorem are instrumental for us to link the four important objects: a continuous function, its continuous Fourier transform/spectrum, as well as the sampled copies of the function and the spectrum.

Computers can store and process discrete quantized samples only, which are known as **digital signals**, instead of continuous data. Hence, we must know how to process discrete data from the sampling process Eq. (5.1.3) and how

to interpret processed results in the continuous domains. For these purposes, **digital signal processing** has been developed as a branch of engineering that is dedicated to computerized analysis of digital signals. As a prerequisite, in the next two sections we first present the relationship between a continuous function and its sampled version and under what condition they are equivalent, and then we motivate the discrete Fourier transform that transforms discrete data into a sampled version of the Fourier transform of the continuous function. A big picture is in Figure 5.1-1.

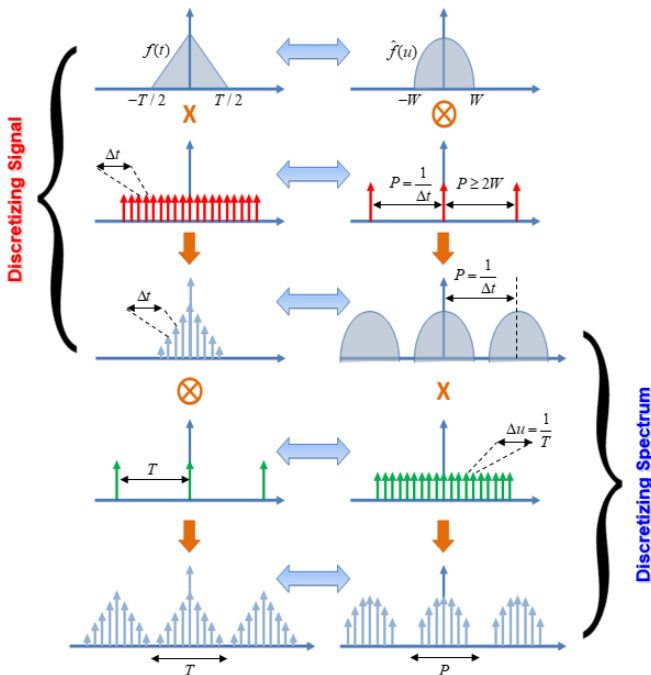


Figure 5.1-1. Steps from a continuous function and its Fourier spectrum to their discrete counterparts, aided by the series of delta functions and the convolution theorem.

SECTION 5.2. SAMPLING THEOREM

Now, let us reveal an equivalency between a continuous function and its discrete/sampled version. As shown in Figure 5.1-1, we start with a continuous function and its continuous Fourier spectrum. First, we use a series of delta functions (in red) to sample the continuous function in the t -domain. This red series of delta functions has its Fourier spectrum as a series of delta functions as well (recall Eq. (5.1.4)). Both the red trains of impulses are similar in shape (both look like combs with an infinitely long length) with their periods being reciprocally related. By the convolution theorem, the multiplication for sampling, as defined by Eq. (5.1.3), means the convolution of the Fourier spectrum of the continuous function with the red series of delta functions in the Fourier space. Hence, we have the middle row of Figure 5.1-1, showing that the sampled function has a periodically repeated Fourier spectra, each of which has the same shape as the original spectrum, as long as P is sufficiently large to avoid any significant overlap between different copies of the Fourier spectrum. Such an overlap is also referred to as **aliasing**. This sampling rate P is called the **Nyquist sampling frequency**, which is no less than twice of the **maximum bandwidth** W of the Fourier spectrum of the original function (the Fourier transform must be zero or insignificant at $-W$ or W if $P = \frac{1}{2W}$ to avoid an aliasing problem!).

By the finite support theorem, *a function and its Fourier transform cannot be of a finite support at the same time unless both being zero*. However, for a well-behaved function/signal of a finite length, its Fourier transform will rapidly decay so that it can be practically treated as a spectrum of a finite support. After the period of the series of delta functions is

appropriately selected for sampling in the t -domain, there will be no significant overlapping in the Fourier space. In this case, we can recover the original function from its sampled version as follows:

$$\begin{aligned} f(t) &= F^{-1} \left[\Pi_p(u) \left(\hat{f}(u) * S_p(u) \right) \right], \\ &= F^{-1} \left[\Pi_p(u) \right] * F^{-1} \left[\hat{f}(u) * S_p(u) \right], \\ &= F^{-1} \left[\Pi_p(u) \right] * \left[F^{-1} \left[\hat{f}(u) \right] F^{-1} \left[\hat{S}_p(u) \right] \right]. \end{aligned}$$

Let us go back to the t -domain, we have

$$\begin{aligned} f(t) &= P \operatorname{sinc}(\pi Pt) * \left[f(t) \left(\frac{1}{P} S_{1/P}(t) \right) \right] \\ &= \operatorname{sinc}(\pi Pt) * \left[\sum_{k=-\infty}^{\infty} f(t) \delta \left(t - \frac{k}{P} \right) \right]. \end{aligned}$$

By Eq. (2.2.4), we finally have

$$f(t) = \sum_{k=-\infty}^{\infty} f \left(\frac{k}{P} \right) \operatorname{sinc} \left(\pi P \left(t - \frac{k}{P} \right) \right). \quad (5.2.1)$$

The above key result can be summarized as

Nyquist/Shannon Sampling Theorem:

If the Fourier spectrum of a continuous function $f(t)$ is non-zero only over $(-W, W)$, $f(t)$ can be perfectly recovered from its discrete values sampled at the Nyquist

sampling rate or a greater rate (no less than $2W$) according to Eq. (5.2.1).

Such a continuous function is called **band-limited**, and can be fully represented by its sampled version using Eq. (5.2.1). This means that the appropriately sampled continuous function does not suffer from loss of information on the original continuous function. Hence, *digital signal processing on discrete data is functionally equivalent to the process in which we directly process the corresponding original continuous data*. This is comforting that under a rather reasonable condition (band-limitedness) a sampled function/digital signal contains essentially the same content of information as that carried by the continuous original.

Next, we explain how to discretize the Fourier transform, which corresponds to the rest part of Figure 5.1-1. To convert the continuous Fourier spectrum/spectra into a discrete version for use in a computer, we need the second series (in green) of delta functions for sampling the continuous periodic Fourier transform in Figure 5.1-1. This green series of delta functions has the counterpart as a series of delta functions in the t -domain. Similarly, both the green trains of delta impulses have their periods being reciprocally related. The effect of the green trains of delta functions is not only to convert the continuous periodic Fourier transform into a discrete periodic function but also to make the discrete function sampled before in the t -domain becoming periodic.

By the sampling theorem, we can recover the continuous function from its sampled version according to Eq. (5.2.1), as long as the sampling rate is sufficiently high.

SECTION 5.3. DISCRETE FOURIER TRANSFORM

Suppose that we have a continuous function/signal $f(t)$ whose support is $(-T/2, T/2)$ or $(0, T)$ with a continuous Fourier spectrum $\hat{f}(t)$ that is significant only over $(-W, W)$ (in other words, the Fourier spectrum can be ignored outside $(-W, W)$). Due to the first sampling process (with the red trains of impulses), we have a periodic continuous spectrum in the Fourier space, and we must use the second sampling process (with the green trains of impulses) to turn the periodic continuous Fourier transform into a periodic discrete function in the Fourier domain.

Similar to what we did in the first sampling process, we now need to sample the periodic continuous function in the Fourier space at a sampling rate no less than $1/T$. According to the sampling theorem again, there will be no aliasing from the sampling process as long as a sufficiently high sampling rate is applied (no less than $1/T$, and the functional values must be zero or insignificant at the boundary of the support interval $(-T/2, T/2)$ or $(0, T)$ if the sampling rate equals $1/T$ to avoid an aliasing in the t -domain). Specifically, the number N of samples in the t -domain is $N = T / \frac{1}{P} = PT$, and

symmetrically the number M of samples in the Fourier domain is

$$M = P / \frac{1}{T} = PT = N, \text{ or}$$

$$\frac{1}{N} = \frac{1}{PT} = \Delta t \Delta u \quad (5.3.1)$$

Again, there should be no information loss in each of the two sampling processes. That is, the set of discrete samples ($N=PT$) in the t -domain and the set of discrete samples ($M=N=PT$) in the Fourier domain are intrinsically linked together in the sense that each of which is equivalent to its original continuous counterparts.

In the t -domain, sampling $f(t)$ gives a list of N samples $\ddot{f}(t) = f(t)s_{\Delta t}(t)$ with $\Delta t = \frac{1}{P}$, which are $f(t_0), f(t_1), \dots, f(t_{N-1})$ where $t_n = n\Delta t, n = 0, 1, \dots, N-1$. Mathematically, we have

$$\ddot{f}(t) = f(t)s_{\Delta t}(t) = \sum_{n=0}^{N-1} f(t_n)\delta(t-t_n), \text{ and}$$

its Fourier transform $\hat{\hat{f}}(u) = \sum_{n=0}^{N-1} f(t_n)e^{-i2\pi u \frac{n}{P}}$, by the translation property Eq. (4.2.3) of the Fourier transform.

On the other hand, in the Fourier domain the sampled $\hat{f}(u)$ also gives a list of N samples $\hat{\hat{f}}(u) = \hat{f}(u)s_{\Delta u}(u)$ with

$\Delta u = \frac{1}{T}$, which are $\hat{f}(u_0), \hat{f}(u_1), \dots, \hat{f}(u_{N-1})$ where $u_m = m\Delta u, m = 0, 1, \dots, N-1$. In a discrete version of the Fourier transform (up to a scaling factor), we have

$$\hat{f}(u_m) \approx \sum_{n=0}^{N-1} f(t_n) e^{-i2\pi \frac{nm}{PT}} = \hat{\hat{f}}(u_m) = \sum_{n=0}^{N-1} f(t_n) e^{-i2\pi \frac{mn}{N}}.$$

Hence, the **discrete Fourier transform** is now well motivated as

$$\hat{f}(u_m) = \sum_{n=0}^{N-1} f(t_n) e^{-i \frac{2\pi mn}{N}}, \quad (5.3.2)$$

which is to compute the continuous Fourier transform at discrete points $u_m = m\Delta u, m = 0, 1, \dots, N-1$, in the Fourier domain from sampled values of the original continuous function $f(t)$ at discrete points $t_n = n\Delta t, n = 0, 1, \dots, N-1$, in the t -domain.

Let us use the integer indices in the bracket to denote sampling points in the t - and Fourier domains respectively. That is, we define $f[0] = f(t_0), f[1] = f(t_1), \dots, f[N-1] = f(t_{N-1})$, and similarly $\hat{f}[0] = \hat{f}(u_0), \hat{f}[1] = \hat{f}(u_1), \dots, \hat{f}[N-1] = \hat{f}(u_{N-1})$. Then, we can neatly define the **discrete Fourier transform** as follows:

$$\hat{f}[m] = \sum_{n=0}^{N-1} f[n] e^{-i \frac{2\pi mn}{N}}, \quad m = 0, 1, \dots, N-1.$$

Once you have these discrete Fourier components $\hat{f}[m]$ ($m=0,1,\dots,N-1$), the original functional samples $f[n]$ ($n=0,1,\dots,N-1$) can be recovered using the **inverse discrete Fourier transform**:

$$f[n] = \frac{1}{N} \sum_{m=0}^{N-1} \hat{f}[m] e^{-i \frac{2\pi mn}{N}}, \quad n=0,1,\dots,N-1.$$

Then, we have **the discrete Fourier transform pair**:

$$\left\{ \begin{array}{l} \hat{f}[m] = \sum_{n=0}^{N-1} f[n] e^{-i \frac{2\pi mn}{N}}, \quad m=0,1,\dots,N-1; \\ f[n] = \frac{1}{N} \sum_{m=0}^{N-1} \hat{f}[m] e^{-i \frac{2\pi mn}{N}}, \quad n=0,1,\dots,N-1. \end{array} \right. \quad (5.3.3)$$

There is the weighting factor $1/N$ in the inverse discrete Fourier transform, which can be understood from the numerical computation of the continuous Fourier transform and its inversion, in which the sampling steps in the t -domain and Fourier space must be used, and whose overall effect is summarized as $1/N$. To appreciate this point, we mentioned

that $\frac{1}{N} = \frac{1}{P} \frac{1}{T} = \Delta t \Delta u$. In other words, an original continuous

function is approximated as a collection of rectangular functions, each of which has a height $f[n]$ and a width Δt , while the continuous Fourier spectrum is viewed as a collection of rectangular functions with a height $\hat{f}[m]$ and a width Δu .

The orthonormal basis is used in the discrete Fourier transform, because we can show that

$$\sum_{k=0}^{N-1} \frac{1}{\sqrt{N}} e^{i2\pi mk/N} \frac{1}{\sqrt{N}} e^{i2\pi nk/N} = \frac{1}{N} \sum_{k=0}^{N-1} e^{i2\pi(m-n)k/N},$$

i.e.,

$$\sum_{k=0}^{N-1} \frac{1}{\sqrt{N}} e^{i2\pi mk/N} \frac{1}{\sqrt{N}} e^{i2\pi nk/N} = \begin{cases} 1, & m = n; \\ 0, & m \neq n. \end{cases} \quad (5.3.4)$$

Remember that when we discussed the Fourier series, we already mentioned the orthonormal property. Based on the just-formulated orthonormality Eq. (5.3.4), we can directly define the discrete Fourier transform and its inversion in several equivalent forms, including Eq. (5.3.3) or a totally symmetric version:

$$\begin{cases} \hat{f}[m] = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} f[n] e^{-i\frac{2\pi mn}{N}}, & m = 0, 1, \dots, N-1; \\ f[n] = \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} \hat{f}[m] e^{-i\frac{2\pi mn}{N}}, & n = 0, 1, \dots, N-1. \end{cases}$$

By now, the linkages become very clear among a continuous function, its continuous Fourier transform, the sampled version of the function, and its Fourier spectrum. The definition of the discrete Fourier transform is very similar to that of the inverse transform. For either of them, it appears that the number of the basic operations (multiplication and addition) is on an order of N^2 , where N is the number of data.

Importantly, there are fast algorithms for the discrete Fourier transform and its inversion, with the computational complexity on an order of $N \cdot \log(N)$. The most popular **fast Fourier transform (FFT)** algorithm and **inverse FFT (IFFT)** algorithms were designed by Cooley & Tukey in 1965. In 1969, the Fourier transform of a 2048-point signal took over 13 hours using a directly implemented program but using the FFT algorithm the same signal processing task only took 2 seconds (Wikipedia-FFT 2018). We will not explain how FFT works here, as it is technically tedious in our context.

An exemplary utility of FFT is to compute the discrete convolution. For two discrete signals $f[n]$ and $g[n]$ each of which is of a length N , the convolved signal has a length $2N-1$, and the computational cost is on an order of N^2 for a directly implemented convolution program. Alternatively, since FFT is much faster, we can quickly ($N \cdot \log(N)$) transfer the two signals into the discrete Fourier domain, perform a multiplication there (N operations), and convert the product back to the signal space using IFFT ($N \cdot \log(N)$).

Performing a discrete convolution using FFT/IFFT means that we have to deal with periodic functions in the t - and Fourier domains. The multiplication in the Fourier space is equivalent to the convolution of two periodical extended versions corresponding to $f[n]$ and $g[n]$ respectively. Hence, what we compute is a **circular convolution**, instead of the conventional **linear convolution**. With a trick called “**zero padding**” which means to increase the length of a signal by adding zeros to it, circular and linear convolutions can give the same results; see a simple example (MATLAB-

CircularConvolution 2018) on the MATLAB website. Another example is to use FFT for spectral analysis (MATLAB-SpectralAnalysis 2018). The key idea to improve the spectral resolution is to use the zero padding technique in the t -domain. Based on Eq. (5.3.1), $\frac{1}{N} = \Delta t \Delta u$, by increasing N with zero padding, for a given Δt we can effectively reduce Δu . The smaller Δu becomes, the better the spectral resolution will be in the Fourier space, which means that we can estimate the Fourier components of a signal more accurately. If you feel difficult to understand this paragraph, please do play with the two cited MATLAB examples.

SECTION 5.4. REMARKS

We have learned two fundamental ways to decompose/represent a function/signal. First, when we introduced the convolution, we learned that a function can be a convolution of the function itself and a delta function. This is to say that the original function can be viewed as a linear combination of many appropriately shifted and scaled delta functions. Then, with Fourier analysis, a function, either continuous or discrete, can be viewed as a sum of various Fourier components. By Eqs. (4.1.7) and (4.1.8) respectively, a delta function in the t -domain and a constant in the Fourier domain form a Fourier transform pair, so do a constant in the t -domain and a delta function in the Fourier domain.

An interesting and important extension of Fourier analysis is so-called **wavelet analysis**. A generic **wavelet** is locally

concentrated, which is much “shorter” than a sinusoidal function. The key mechanism for wavelet analysis is that the basis functions are formed from the generic or “mother” waveform through scaling its support and translation in the t -domain, without altering the shape of the mother waveform. Compressed wavelet shapes in the t -domain correspond to enhanced high frequency contents in the Fourier space, more efficiently representing signals through linear combination in many practical applications such as signal/image compression. Two mother wavelets are in Figure 5.4-1.

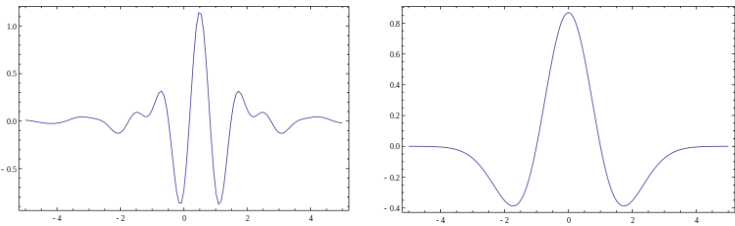


Figure 5.4-1. Two mother wavelets: (Left) The Meyer wavelet, infinitely differentiable, and (Right) the Mexican hat, which is the negative normalized second derivative of a Gaussian function.

A major task of signal processing is to reconstruct a signal from discrete measurements. In an unconstrained space, this is an impossible mission since the signal could take any value when it is not measured between two adjacent sampling points. However, with a reasonable signal model such as the band-limited signal model, it becomes feasible to recover the signal faithfully from its sampled version, as well demonstrated by our derivation of the Nyquist sampling theorem.

Just like the representation of a signal is not unique (for example, it can be expressed as a series of translated and weighted delta functions or a sum of sinusoidal functions), the signal model is not unique. Different from the band-limited signal model, the signal sparsity model has attracted a major attention since this millennium. As a result, *a signal may be reconstructed from a number of samples far less than what is requested by the Nyquist sampling theorem*. This area of research is known as **compressed sensing** or **compressive sensing**. Among many potential signals that fit into sampled data, a recovered signal is requested to be sparse in a good sense; for example, its **total variation (TV)** (Han, Yu et al. 2009, Yu and Wang 2009, Yu, Yang et al. 2009). In a 1D case,

TV can be defined as
$$TV = \sum_{n=1}^{N-1} |f[n] - f[n-1]|.$$
 The

minimization of TV implies that the recovered signal tends to be piecewise constant. While TV encourages sparsity in terms of edginess, there are other sparsifying transforms such as high-order TV measures (Yang, Yu et al. 2010) and dictionary representation (Xu, Yu et al. 2012).

Both the band-limited assumption and the sparsity constraint are quite mathematical. A superior way to represent data features is to have a big dataset of relevant samples. In a fundamental sense, such **big data** should be our best signal model! For example, the **ImageNet** project is a large image database to test visual object recognition algorithms (ImageNet 2018). ImageNet contains multi-millions of URLs pointing to annotated images. A signal/image recovery algorithm can be trained on these images for the best

performance, and this **data-driven** approach is characteristic of the rapidly-developing field known as **machine learning**.

Finally, let me comment on an open question related to Eq. (5.1.4), i.e., why $s(t) \Leftrightarrow s(u)$ or $s_{\Delta}(t) \Leftrightarrow \frac{1}{\Delta} s_{1/\Delta}(u)$? This question arises from the need to perform Fourier analysis on $d(t) = \langle f(t), s_{\Delta}(t) \rangle$. Hence, we require a reasonable $d(t)$ so that its Fourier analysis will make sense. For example, we do not want to have $d(t) = \sum_{k=-\infty}^{\infty} \delta(t - k\Delta)$, which is the case when $f(t) = 1$; otherwise, the Fourier integral will diverge.

Such a problem can be avoided if we require that $f(t)$ is in the Schwartz space. The name might sound too fancy but we can simply consider it as a space consisting of all “good” functions that will make the Fourier transform of $d(t)$ converge. More rigorously speaking, the Schwartz space consists of all smooth functions whose derivatives decay faster than any power; i.e., $t^{\alpha} \frac{\partial^{\beta} f(t)}{\partial t^{\beta}} \rightarrow 0$ as $|t| \rightarrow 0$ for any non-negative integers α and β . Basically, the citizenship of a function in the Schwartz space means that the function allows Fourier analysis. Although $s_{\Delta}(t)$ as a generalized function (a series of generalized functions) is a strange animal not living in the Schwartz space, $d(t) = \langle f(t), s_{\Delta}(t) \rangle$ can be analyzed using the Fourier transform if $f(t)$ is in the

Schwartz space, thanks to the extremely nice character of $f(t)$. Since practical functions/signals are of finite extent and can be well approximated with smooth curves if needed, they are in the Schwartz space as a good playground for our engineers.

First, let us consider the Fourier transform of $s(t)$, which is tricky since the Fourier transform of $s(t)$ does not converge:

$$\hat{s}(u) = \sum_{k=-\infty}^{\infty} e^{-i2\pi ku}. \text{ However, for our measurement purpose, we}$$

only need the Fourier transform of the inner product $d(t) = \langle f(t), s(t) \rangle$. *As long as the inner product and its Fourier transform make sense, we can do our job without any problem. That is to say, we should interpret the Fourier transform in the sense of such inner products.* It is in this sense that it can be rigorously proven that Eq. (5.1.4) is valid. The rough ideas are as follows.

Geometrically, if a function is in the Schwartz space, its Fourier transform is also in that space, since the Fourier transform is to represent the original function in another orthonormal coordinate system and does not change its intrinsic properties. Thus, both $\langle \hat{s}, f \rangle$ and $\langle s, \hat{f} \rangle$ exist in the sense of the unconjugated inner product Eq. (3.3.6) (with abuse of notation, please read (Osgood 2018) for more rigorous treatment, which is an outstanding book but it is too long for our purpose):

$$\langle \hat{s}, f \rangle = \langle s, \hat{f} \rangle. \quad (5.4.1)$$

Proof:

$$\begin{aligned} \langle \hat{s}, f \rangle &= \int_{-\infty}^{\infty} f(u) \left(\int_{-\infty}^{\infty} s(v) e^{-i2\pi uv} dv \right) du \\ &= \int_{-\infty}^{\infty} s(v) \left(\int_{-\infty}^{\infty} f(u) e^{-i2\pi uv} du \right) dv = \langle s, \hat{f} \rangle. \end{aligned}$$

Poisson Summation Formula: If we let $f(t)$ be in the Schwartz space, then

$$\sum_{l=-\infty}^{\infty} \hat{f}(l) = \sum_{k=-\infty}^{\infty} f(k). \quad (5.4.2)$$

Proof:

First, we extend $f(t)$ into a periodic function

$$F(t) = f(t) * s(t) = \sum_{k=-\infty}^{\infty} f(t-k).$$

Next, we expand it into the Fourier series

$$F(t) = \sum_{l=-\infty}^{\infty} c_l e^{i2\pi lt}$$

Where

$$c_l = \sum_{k=-\infty}^{\infty} \int_0^1 e^{-i2\pi lt} f(t-k) dt = \sum_{k=-\infty}^{\infty} \int_{-k}^{-k+1} e^{-i2\pi l(t+k)} f(t) dt.$$

Since $e^{-i2\pi lk} = 1$, we have

$$c_l = \sum_{k=-\infty}^{\infty} \int_{-k}^{-k+1} e^{-i2\pi lt} f(t) dt = \int_{-\infty}^{\infty} e^{-i2\pi lt} f(t) dt = \hat{f}(l).$$

If we evaluate $F(0)$ in both the originally extended form and the Fourier series, then the Poisson summation formula is immediately obtained.

In the sense of the un-conjugated inner product in the Schwartz space, we can find the Fourier transform of $s(t)$ as follows. Without loss of generality, let us assume that $f(t)$ is a real-valued function. By Eq. (5.4.1),

$\langle \hat{s}, f \rangle = \langle s, \hat{f} \rangle = \sum_{l=-\infty}^{\infty} \hat{f}(l)$. By the Poisson summation

formula (5.4.2), we have

$$\begin{aligned} \langle \hat{s}, f \rangle &= \sum_{l=-\infty}^{\infty} \hat{f}(l) = \sum_{k=-\infty}^{\infty} f(k) = \sum_{k=-\infty}^{\infty} \langle \delta(t-k), f(k) \rangle \\ &= \left\langle \sum_{k=-\infty}^{\infty} \delta(t-k), f(k) \right\rangle = \langle s, f \rangle. \end{aligned}$$

Therefore, $\langle \hat{s}, f \rangle = \langle s, f \rangle$ or $\hat{s} = s$ as far as our intended measurement process is concerned. That is, $\hat{s}(u) = s(u)$ or $s(t) \Leftrightarrow s(u)$, and it can be similarly shown that

$\hat{s}_{\Delta}(u) = \frac{1}{\Delta} s_{1/\Delta}(u)$ or $s_{\Delta}(t) \Leftrightarrow \frac{1}{\Delta} s_{1/\Delta}(u)$. In other words, a

series of delta functions is still a train of impulses in the reciprocal space. The longer the period in one space, the shorter the period in the other space.

Very interestingly, we saw that $e^{-\pi t^2} \Leftrightarrow e^{-\pi u^2}$, and now we have $\hat{s} = s$. It was mentioned that in the context of the Fourier

transform pair, the wider a function in one domain, the narrower its counterpart in the other domain. For a constant function in one domain, we have a delta function in the other domain. As the delta function is more and more spread out, the constant function becomes more and more localized. The Gaussian function is the sweet balanced point so we have $e^{-\pi t^2} \Leftrightarrow e^{-\pi u^2}$. You might have been convinced that the Gaussian function is the only symmetric solution, but now we have $\hat{s} \Leftrightarrow s$, which emerged from a different perspective. Because s is both widely spread out and highly localized, by the duality of the Fourier transform so should \hat{s} , providing another balancing point $\hat{s} \Leftrightarrow s$. Do you have any additional solutions of this type?

CHAPTER 6. NETWORK

A **network** is an important type of complex systems, and consists of many interconnected components, modules, and sub-systems. Due to the interconnections in the network, it can be modeled as a composition function with multiple inputs, outputs, and couplings. Typical cases of networks are **electrical networks**, **biological neural networks**, and **artificial neural networks** that recently attract a major attention. In this chapter, we will discuss basic concepts and techniques solving for unknowns in electrical networks, also referred to as **circuit analysis**. Then, we will introduce biological and artificial neural networks, touching on the emerging field of **machine learning** as it pertains to medical imaging.

SECTION 6.1. ELECTRICAL NETWORK

An electrical network, or a circuit, consists of electrical components such as power **sources**, **resistors**, **capacitors**, **inductors**, and **operational amplifiers** that are linked to different **nodes** and form various **loops**. Each electrical component is described by a specific relationship between the voltage across the component and the current through it. In every circuit, there should be one or more voltage and/or current **sources** to provide driving force. Circuit analysis is to determine the voltage and current for each electrical component.

These systems can be easy or very hard to analyze depending on the complexity of the circuit in question. First, let us introduce

Ohm's law for a resistor, and then generalize it for a capacitor and an inductor. Ohm's law in its generalized form can be directly used to find the voltage across an electrical component from the current through it, or vice versa.

You may recall that Ohm's law is the voltage-current relationship for a resistor,

$$V = RI \quad (6.1.1)$$

where V denotes the voltage which is a driving force to generate a current, I is the current which is the flow of electrons (I), and R is the **resistance** reflecting the capability of a **resistor** to resist or restrict the current. The voltage $V=RI$ can also be interpreted as the voltage the resistor of resistance R takes when the current is I . Assuming a constant voltage drop across a resistor, increasing resistance will reduce the current through the resistor.

However, we cannot build many interesting circuits without the use of other components, such as capacitors and inductors. A **capacitor** consists of parallel metallic plates where charges can be stored. It works by having the plates accumulate opposite charges due to a driving voltage applied across the capacitor. These charges will act against the driving voltage and eventually counteract the voltage to the point that no more current flows. Note that capacitors have different capabilities to store charges or electrical energy based upon their geometry and dielectric material between the plates of the capacitor. This energy-storing capability is known as **capacitance**. The voltage, the amount of stored charges, and capacitance are related by the equation

$$Q(t) = CV(t), \quad (6.1.2)$$

where Q is the amount of charges (positive or negative), C is the capacitance of the capacitor, and V is the driving voltage across the capacity. Since the current is the first derivative of the amount of charges with respect to time, we have

$$I(t) = \frac{dQ}{dt} = C \left(\frac{dV}{dt} \right), \quad (6.1.3)$$

where I is the current through the capacitor, and $\frac{dV}{dt}$ is the change in the voltage over time. Alternatively, we have the following relationship:

$$V(t) = \frac{1}{C} \int_{t_0}^t I(t) dt + V(t_0), \quad (6.1.4)$$

where $V(t_0)$ is the initial voltage.

As a counterpart to a capacitor characterized by $I = C \left(\frac{dV}{dt} \right)$, we have another component known as an **inductor** characterized by

$$V(t) = L \left(\frac{dI}{dt} \right). \quad (6.1.5)$$

An inductor is a coiled wire structure that generates an electromagnetic field when a current I flows through the coil,

with the voltage across the inductor being proportional to the rate of change of the current. The proportional coefficient, L , is known as **inductance**. The inductance depends on not only the geometry of the current path but also the magnetic permeability of nearby materials. In other words, the inductor opposes any change in the current, and the voltage must be applied to increase the current. The differential relationship between the voltage and current for an inductor can be also put in the integral form:

$$I = \frac{1}{L} \int_{t_0}^t V(t) dt + I(t_0), \quad (6.1.6)$$

where $I(t_0)$ is the initial current. Similar to Eq. (6.1.2) in the case of a capacitor, in the case of an inductor we have

$$\Phi(t) = LI(t), \quad (6.1.7)$$

where Φ is the magnetic flux which reflects the magnetic energy stored in the coil.

As we have discussed, resistors, capacitors and inductors can be used to make up circuits. A circuit made up of power sources and only these three types of components is referred to as an **RCL circuit**. Since each of these components is linear (you can verify), an **RCL network** is also linear.

Recall that we mentioned earlier that the convolution theorem is only valid for Fourier transform. *This means that a stable response of an RCL circuit to a sinusoidal signal remains*

sinusoidal at the same frequency. That is, in the steady state, a resistor, a capacitor, an inductor, or an RCL network will not change the waveform of a sinusoidal input/excitation, and the output/response will be a sinusoidal function at the same frequency with a scaled amplitude and a changed phase. Thus, once we have the amplitude and the phase of the response wave, we will know the wave exactly.

Mathematically, a sinusoidal wave $F(t) = A \cos(\omega t + \phi)$ can be put in a complex form $\tilde{F}(t) = A e^{-i(\omega t + \phi)}$, and $F(t) = \text{Re}(\tilde{F}(t))$. For a stable response of an RCL circuit driven by a power source at a given frequency, the voltage and current for any component of the circuit will vary at the same frequency, and only A and ϕ matter. Hence, we can simplify the complex notation into the phasor notation $F = A \angle \phi$.

With this in mind, let us reanalyze the voltage-current relationship for a capacitor. For a voltage source $V(t) = A \cos(2\pi ft) = A \cos(\omega t)$, we have

$$I(t) = C \frac{dV(t)}{dt} = -C\omega A \sin(\omega t) = C\omega A \sin(\omega t + \pi).$$

That is,

$I(t) = C\omega A \cos(\omega t + \pi - \frac{\pi}{2}) = C\omega A \cos(\omega t + \frac{\pi}{2})$. In the complex notation, $\tilde{V}(t) = A e^{-i(\omega t)}$, $\tilde{I}(t) = C\omega A e^{-i(\omega t + \frac{\pi}{2})}$, which motivate the definition of the **capacitance**

$$Z_c = \frac{\tilde{V}}{\tilde{I}} = \frac{1}{iC\omega} \quad (6.1.8)$$

so that

$$I(t) = \operatorname{Re} \left(\frac{\tilde{V}(t)}{Z_c} \right) = C\omega A \cos(\omega t + \frac{\pi}{2}).$$

In the **phasor** form (a simplest notation), we have $V = A\angle 0^\circ$ and $I = C\omega A\angle 90^\circ$.

In the case of an inductor, the derivation is very similar. For a current source $I(t) = B \cos(2\pi ft) = B \cos(\omega t)$, we have

$$V(t) = L \frac{dI(t)}{dt} = -L\omega B \sin(\omega t) = L\omega B \sin(\omega t + \pi), \text{ or}$$

$$V(t) = L\omega B \cos(\omega t + \pi - \frac{\pi}{2}) = L\omega B \cos(\omega t + \frac{\pi}{2}).$$

In the complex notation, $\tilde{I}(t) = B e^{-i(\omega t)}$, $\tilde{V}(t) = L\omega B e^{-i(\omega t + \frac{\pi}{2})}$, and then we define the **inductance**

$$Z_L = \frac{\tilde{V}}{\tilde{I}} = iL\omega \quad (6.1.9)$$

so that

$$V(t) = \operatorname{Re} \left(Z_L \tilde{I}(t) \right) = L\omega B \cos(\omega t + \frac{\pi}{2}).$$

In the phasor form, we have $I = B\angle 0^\circ$ and $V = L\omega B\angle 90^\circ$. While the current is phased 90° ahead of the voltage in the case of a capacitor, the current is phased -90° behind the voltage in the case of an inductor.

Now, we are ready to extend Ohm's law, $V = IR$, which is the case where there is neither phase advance nor delay, to the general case with R being replaced with Z , which is called **impedance**. Generally speaking, impedance means resistance in a generalized form that can be applied to both **direct current (DC)** and **alternating current (AC)** circuits assuming that the involved voltage and current are constant or sinusoidal. Specifically, Z can be computed in a way determined by the type of components: For a resistor $Z_R = R$,

for a capacitor $Z_C = \frac{1}{j\omega C}$, and for an inductor $Z_L = j\omega L$.

Since impedance links the voltage across a component and the current through the component in the same way as resistance does in Ohm's law $V = IR$, we can generalize Ohm's law as

$$\tilde{V} = Z\tilde{I} \quad (6.1.10)$$

where Z is the impedances.

In a general case, impedance is a complex number reflecting the resisting effect of a single or multiple components. Indeed, components can be linked together in series or in parallel. The **equivalent impedance** Z_e can be computed in each case by treating impedance components Z_1 and Z_2 in series or parallel as a whole block with an overall voltage across the block and a total current through the block. The resultant formulas for series and parallel connections are respectively:

$$Z_e = Z_1 + Z_2 \text{ and} \quad (6.1.11)$$

$$\frac{1}{Z_e} = \frac{1}{Z_1} + \frac{1}{Z_2}, \quad (6.1.12)$$

as shown in Figure 6.1-1. As your exercise, please prove Eqs. (6.1.11) and (6.1.12).

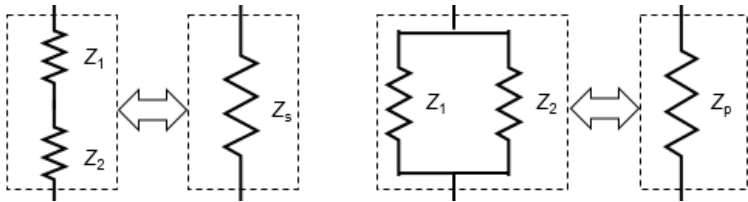


Figure 6.1-1: Equivalent impedance of two impedance components in series and parallel connections respectively.

When linking more components including voltage and current sources into a more complicated network, it becomes more difficult to analyze the voltage and current for each component. Now, let us introduce a few concepts that are essential to analyze a circuit systematically. The first is a **node**, which is the point where circuit elements are connected. The second is called a **branch** which is the path between two nodes. The last is a **loop** which is a closed path through involved branches only once. To practice, please count how many nodes, branches, and loops in Figure 6.1-2. Using these concepts, we can explain two **Kirchhoff's laws** to setup a system of linear equations in terms of voltage across, current through, and impedance of any component.

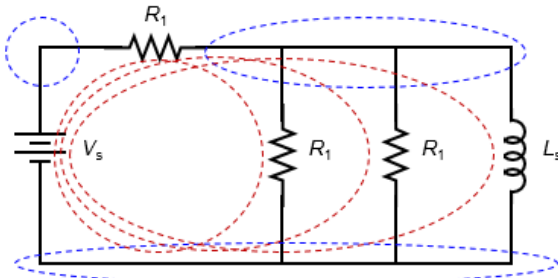


Figure 6.1-2. Three nodes (blue) and three loops (red) in the circuit.

The first is **Kirchhoff's current law (KCL)**, which states that the sum of the currents into and out of a node equals zero. In other words, charges can be moved around but they cannot be created or destroyed. KCL means that the total amount of charges must be conserved, much like mass conservation. The second is **Kirchhoff's voltage law (KVL)**, which states that the sum of the potential differences around a loop equals zero. In other words, the total voltage drop around the loop should equal the total voltage due to the sources, just like that the total change in potential energy must be zero after you return to your destination from climbing a hill. KCL and KVL are illustrated in Figure 6.1-3.

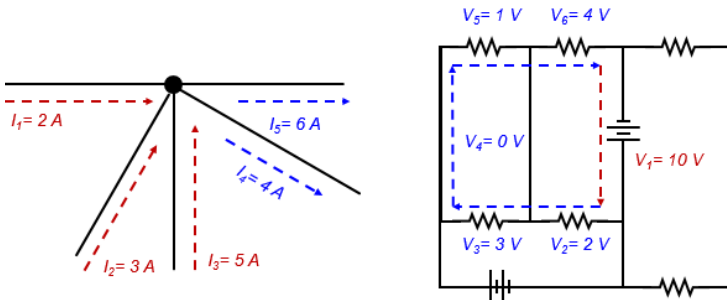


Figure 6.1-3. KCL (Left) and KVL (Right) reflect conservation of charge and energy respectively.

The use of KCL and KVL as needed will guarantee that an electrical network can be solved for a unique solution. This statement can be appreciated heuristically as follows. Suppose that a basic network can be solved with a unique solution (indeed, for a single branch we can just use Eq. (6.1.10), $V=ZI$, to find the solution). Then, as shown in Figure 6.1-4, we can do three kinds of enhancement to the basic network: (1) adding a loop without adding any node (the red line forms a new loop), (2) adding a node without increasing the number of loops (the green node is new), and (3) adding a loop and a node (the blue node is new, and the blue line forms a new independent loop). In the first case, applying KVL to the new loop gives us an additional independent equation to solve for the voltage across the red branch. In the second case, the potential differences from the new node to each of the adjacent nodes can be easily determined since by assumption we can find the potential difference between the two adjacent nodes. In the third case, we can apply KCL and KVL to the new node and the new loop to solve for the voltage across the blue branch and the current division at the blue node along the pre-existing branch.

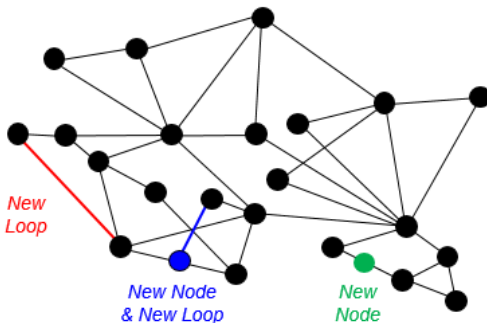


Figure 6.1-4. Basic network (Black) and enhanced one (Colored components) by adding a loop without adding any node (the red line), adding a node without increasing the number of loops (the green node), and adding a node and a loop (the blue node and line).

Using KCL and KVL at convenience, we should be able to set up a system of sufficiently many independent linear equations to solve for unknown voltage and current variables. For example, Figure 6.1-5 shows how linear equations are listed for a circuit.

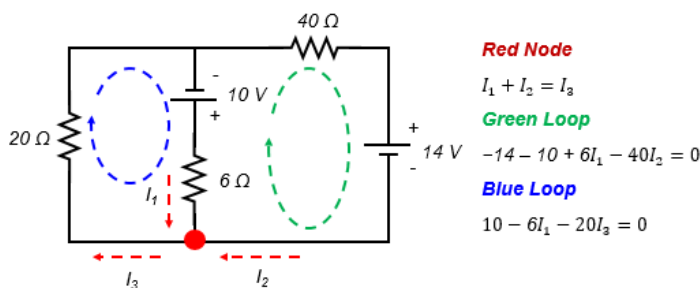


Figure 6.1-5. Three independent linear equations obtained using KCL and KVL at convenience.

Although the above circuit only uses resistors and batteries, it should be noted that KCL and KVL can be applied to AC circuits as well, since the charge conservation and the energy conservation hold regardless the type of circuits, DC or AC. For the steady state response of an RCL circuit, the system of linear equations will be in the complex space, and can be similarly solved.

SECTION 6.2. EXEMPLARY CIRCUITS

Using resistors, capacitors and inductors, we can create useful devices. An analog **low-pass filter** is a good example, which works by filtering out higher frequency signals using a combination of a resistor and either a capacitor or an inductor, as shown in Figure 6.2-1. Note that one could similarly create other types of filters, such as a **high-pass filter** that suppresses low frequency signals, and a **band-pass filter** that makes a certain frequency range more transparent to an input signal.

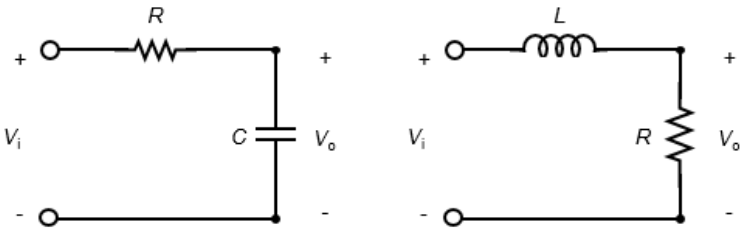


Figure 6.2-1. Low-pass filters using a combination of a resistor and either a capacitor (Left) or an inductor (Right).

Taking the RC circuit in Figure 6.2-1 as an example, we have

$$\tilde{V}_i(t) = Z_e \tilde{I}(t) = \left(R + \frac{1}{i\omega C} \right) \tilde{I}(t) \text{ and}$$

$$\tilde{V}_o(t) = Z_c \tilde{I}(t) = \frac{1}{i\omega C} \tilde{I}(t).$$

Then, the **transfer function** is defined as

$$T(\omega) = \frac{\tilde{V}_o(t)}{\tilde{V}_i(t)} = \frac{\frac{1}{i\omega C}}{R + \frac{1}{i\omega C}} = \frac{1}{1 + i\omega RC} = \frac{1}{1 + i\frac{\omega}{\hat{\omega}}},$$

where $\hat{\omega} = \frac{1}{RC}$ is the bandwidth of the low-pass filter.

An input signal can be decomposed using the Fourier transform into many sinusoidal components at different frequencies. Because the transfer function depends on the frequency of a sinusoidal stimulation, the output response will be quite close to the sinusoidal input when its frequency is small relative to the bandwidth, and the output response will be nearly zero when its frequency is much greater than the bandwidth.

As another type of examples, let us discuss circuits for weak signal detection. First, let us look at a **voltage divider** and a **Wheatstone bridge**. A resistive **voltage divider** is a simple circuit comprised of two resistors in series with a connection coming out between the two resistors. This setup can be manipulated so that a certain fraction of an input voltage is transferred through the circuit to be picked up at the next stage. This amount of voltage to be found is as follows:

$$V_{Out} = \frac{R_2}{R_1 + R_2} V_{In}. \quad (6.2.1)$$

Any small perturbation to either the input voltage or the resistor will lead to a change of the output voltage. Typically, such a change is the signal to be detected, which deviates from the nominal output voltage. Since the signal is small, we need an **amplifier** (to be discussed below) to increase the signal by a large factor. This will magnify the nominal output background as well, which could yield a very high voltage and

become impractical. For signal magnification, it is highly desirable to design a circuit with a zero output unless there is a signal. This is the purpose of the **Wheatstone bridge**, which consists of two voltage dividers as shown in Figure 6.2-2. Under the balancing condition for the Wheatstone bridge:

$$\frac{R_1}{R_2} = \frac{R_3}{R_x}, \quad (6.2.2)$$

the voltage between the midpoints of the two voltage dividers will be zero. If there is any change in one of the resistors (called the **reference resistor** or a **sensor**) the measured voltage will be a non-zero signal.

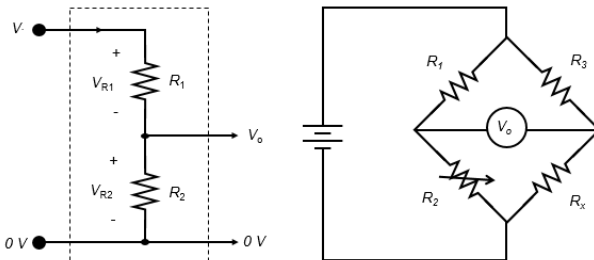


Figure 6.2-2: Single voltage divider (Left) and the Wheatstone bridge (Right) consisting of two voltage dividers for weak signal detection.

While a resistor is our sensor in Figure 6.2-2, a capacitor or an inductor can be used for sensing as well. For example, with a capacitive displacement sensor, any relative displacement of the plates in the capacitor will change the capacitance. Hence, the displacement can be recorded as the resultant voltage change (say, in the phasor form).

Because weak signal detection is fundamentally important to medical imaging, let us have a deep look at the voltage divider in Figure 6.2-2. From the input perspective, the sensor should have a very large resistance or impedance so that the voltage signal can be effectively captured by the sensor, instead of being wasted on non-sensing components. However, from the output perspective, the sensor should have a very small resistance or impedance so that the voltage signal can be effectively relayed to the next stage, instead of being kept on the sensor. Again, we want to magnify the voltage signal by a large factor either across the sensor in the voltage divider or between the midpoints of the two voltage dividers in the Wheatstone bridge. *These requirements suggest a need for a dreamed component that has a large input impedance, a small output impedance, and a huge magnification factor.* This dream came to true when the **operational amplifier** was invented!

An operational amplifier is a nonlinear component that has large input impedance and small output impedance, and serves to increase a weak voltage to a very high level, as summarized in Figure 6.2-3. Such an amplifier consists of many transistors driven by a power source. Each transistor is a fusion of two diodes. A diode is a nonlinear semiconductor component through which a current flow easily along one direction and much harder in the other direction. Despite the physical complexity of an operational amplifier, its ideal and typical characteristics can be summarized in Table 6.2-1, and represented in a simple diagram in Figure 6.2-3.

<i>Parameter</i>	<i>Ideal Value</i>	<i>Typical Value</i>
<i>Open-loop Gain</i>	∞	10^5 - 10^9

<i>Common Mode Gain</i>	0	10^{-5}
<i>Bandwidth</i>	∞	1-20 MHz
<i>Input Impedance</i>	∞	10^6 - $10^{12} \Omega$
<i>Output Impedance</i>	0	100-1,000 Ω

Table 6.2-1. Key features of an operational amplifier.

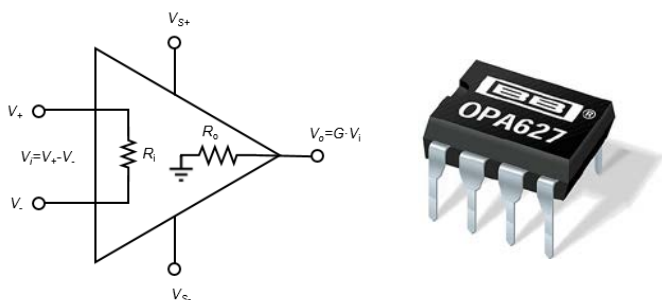


Figure 6.2-3. Symbol and rendered views of an operational amplifier, which is often packed in an integrated circuit chip.

When it comes to working with operational amplifiers, it is important to note two rules. First, the two input terminals are at the same voltage (otherwise, the output voltage would be huge). Second, no current flows into either of the two input terminals (otherwise, the two input terminals would not be at the same voltage). Using these two rules, one will be able to analyze circuits involving an operation amplifier. To practice, please analyze the first circuit module (and more if you like) in Figure 6.2-4 to see if you can find the relationship between input and output voltages.

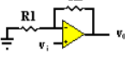

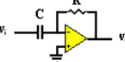
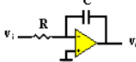
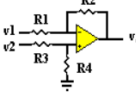
<p style="text-align: center;">Non-Inverting</p>  $v_o = \left(1 + \frac{R_2}{R_1}\right) \times v_i$	<p style="text-align: center;">Inverting</p>  $v_o = -\frac{R_2}{R_1} \times v_i$	
<p style="text-align: center;">Differentiating</p>  $v_o(t) = -RC \frac{dv_i(t)}{dt}$ <p>At any single frequency the gain is:</p> $v_o = -2\pi fRC \times v_i = -\omega RC \times v_i$	<p style="text-align: center;">Integrating</p>  $v_o(t) = -\frac{1}{RC} \int_0^t v_i(t) dt$ <p>At any single frequency the gain is:</p> $v_o = -\frac{v_i}{2\pi fRC} = -\frac{v_i}{\omega RC}$	<p style="text-align: center;">Differential</p>  $v_o = \frac{1 + \frac{R_2}{R_1}}{1 + \frac{R_3}{R_4}} \times v_2 - \frac{R_2}{R_1} \times v_1$

Figure 6.2-4. Typical circuit modules constructed around an operational amplifier.

To make a digital computer, nonlinear components must be used to form logical gates where the output voltage will be high (“1”) or low (“0”) depending on wherever certain logical requirements are met at the input ports (for example, an “AND” gate will output a high voltage if and only if all input variables are at a high voltage. Arithmetic and symbolic operations can be implemented as logic operations, and that is why a digital computer is so powerful.

SECTION 6.3. NEURAL NETWORK

Networks can come in many different forms other than the electrical networks we have discussed. In particular, there exist fascinating networks within the realm of biology, namely biological neural networks. These networks are also kind of electrical networks because signals are generated and

transmitted as electrical pulses. The nervous systems of organisms are primarily made up of **biological neurons**. These neurons themselves can be very complex, and the exact mechanisms by which they function are beyond the scope of this book. However, we can simplify the neuron as shown in Figure 6.3-1. Note that **inputs** to a neuron are not treated in the same way; i.e., **weighting factors** associated with the stimuli are generally different, with certain pathways encouraging stimulation and the others suppressing it. After the weight processes, all the input stimuli are accumulated together inside the neuron. If the overall stimulation is too weak, the neuron will ignore it. However, if the total effect is strong relative to a certain **threshold**, an electrical impulse will be generated as a non-zero **output** of the neuron, and transmitted along neural fiber to other neurons. While each neuron works based upon fairly standard rules, numerous neurons are interconnected to form a biological neural network from which **intelligence** emerges!

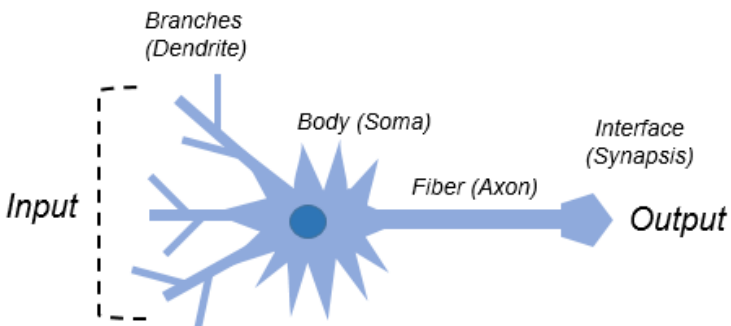
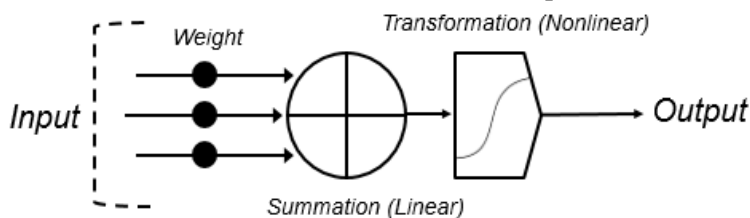


Figure 6.3-1. Biological neuron with multiple inputs via dendrites and a single output in response to an overall effect of

all weighted stimuli, whose output will be sent along the axon to other neurons if the response is over a threshold.

Inspired by this amazing insight, **artificial neurons** and **artificial neural networks** were proposed decades ago. Over past several years, this biomimetic approach, as the mainstream of **machine learning** and **artificial intelligence**, has achieved astonishing achievements! This is the very technology behind auto-driving cars, natural language understanding and translation systems, and game-playing programs such as *Alpha-Go*, as a few examples.

The main idea of an artificial neural network is to map the inner working of a biological neuron into a mathematical operator and interconnect such operators into a system for various computational tasks. *An artificial neuron is a counterpart of a biological neuron.* As an operator, an artificial neuron works in two steps: (1) compute a **linear combination** by which all inputs are individually weighted and then summed into a single number; and (2) perform a **nonlinear activation** in which the inner product is fed into a nonlinear function to produce an output. The artificial neuron is illustrated in Figure 6.3-2. If we compare Figures 6.3-1 and 6.3-2, we can see the correspondences between



between the biological and artificial neuron components.

Figure 6.3-2. Artificial neuron as a counterpart of a biological neuron.

To describe an artificial neuron mathematically, the inner product part is expressed as the following linear function of the input variables $\mathbf{x} = \{x_i\}$, x_i , $i = 0, 1, 2, \dots, n$:

$$f(\mathbf{x}) = \sum_{i=0}^n w_i x_i, \quad (6.3.1)$$

where $\mathbf{w} = \{w_i\}$, $i = 0, 1, 2, \dots, n$, $w_0 = b$ is an offset, and $x_0 = 1$. Then, $f(\mathbf{x})$ will be nonlinearly processed, such as by a **sigmoid function**

$$\sigma(x) = \frac{1}{1 + \exp(-\beta x)}, \quad (6.3.2)$$

where β is a constant.

Clearly, the single neuron can separate two sets of inputs that are linearly separable. In contrast, for linearly inseparable tasks, a single neuron is subject to classification errors. For example, a single neuron is incapable of simulating the **XOR gate** which is a **binary function** defined as follows:

$$f(x, y) = \begin{cases} 0, & x = y; \\ 1, & x \neq y. \end{cases} \quad (6.3.3)$$

With artificial neurons as building blocks, an artificial neural network can be built of various architectures. The **feedforward network** uses **layers** of artificial neurons and passes data from low to high levels to extract from local to global features. In this setting, each neuron in the previous layer will provide its output as an input to the next layer, and such a network is potentially very powerful for signal processing, image reconstruction and analysis.

The potential of an artificial neural network can be maximized when its architecture is appropriate, its activation functions are effective, and its parameters are optimized. Currently, there are neither rigorous theory nor surely-working guidelines in specifying a network topology and an activation function. Fortunately, this field is extremely hot, with new results being published constantly. Based on collective wisdom and experience, we do have a good sense what kinds of network architectures and activation functions should be used for trial.

Given a network architecture and a nonlinear activation mechanism, the rest of the task is to optimize all the parameters including both weights and offsets. Starting from a random guess, we can keep improving the parameters according to a **loss function** that measures the discrepancy between the desirable output and the current output of the network. This **optimization** problem can be addressed using a **gradient descent search method**, or an alternative.

Gradient descent search is a first-order optimization algorithm to find the minimum of an objective function

iteratively along the gradient direction which gives the greatest improvement when the current parametric setting is improved by a small amount. In other words, you need to take a small step each time proportional to the negative of the gradient (or of the approximate gradient) of the function from the currently guessed parameters. Mathematically, we have

$$\mathbf{w}_{k+1} := \mathbf{w}_k - \eta \frac{\partial E(\mathbf{w}_k)}{\partial \mathbf{w}} \quad (6.3.4)$$

where E is the loss function between the output $h(\mathbf{x})$ of the network given the current parameters and the ideal output \mathbf{y} :

$$E(\mathbf{w}) = \sum_{i=0}^n (h(\bar{x}_i) - y_i)^2, \quad (6.3.5)$$

and η is the learning rate which is a scaling factor typically between zero and one and can be adaptively changed. In the optimization process, the weight values will naturally follow the gradient direction which is spatially varying until the derivative term vanishes, indicating that the parameters are very close to an optimal solution.

For a single neuron, we compute the gradient vector as follows:

$$\frac{\partial E(\mathbf{w})}{w_i} = \frac{\partial}{w_i} \sum_{j=0}^N (h(\bar{x}_j) - y_j)^2 \quad (6.3.6)$$

$$\begin{aligned}
&= \sum_{j=0}^N 2 \left(h(\bar{x}_j) - y_j \right) \frac{\partial}{\partial w_i} \left(h(\bar{x}_j) - y_j \right) \\
&= 2 \sum_{j=0}^N \left(h(\bar{x}_j) - y_j \right) \sigma'(\mathbf{x}_j \cdot \mathbf{w}) \frac{\partial}{\partial w_i} (\mathbf{x}_j \cdot \mathbf{w}) \\
&= 2 \sum_{j=0}^N \left(h(\bar{x}_j) - y_j \right) \sigma'(\mathbf{x}_j \cdot \mathbf{w}) x_{i,j}.
\end{aligned}$$

Along the gradient direction, we can optimize the neuron gradually to produce the desired output as closely as possible.

Take this further, multi-layer neural networks can be similarly optimized. However, the exact steps are not shown here and can be found online if you are curious. Indeed, the basic idea remains the same as that for the single neuron optimization but the **chain rule** is needed to derive a dedicated algorithm known as **back propagation**.

SECTION 6.4. REMARKS

The artificial neuron is a mathematical function, and can be built as an electrical network. Specifically, input variables can be voltage or current values, and weighting parameters can be implemented with resistors connecting to a common node. Then, the nonlinear activation of the total signal at the node can be implemented using a nonlinear component as simple as a diode. From the electrical perspective, it is very natural to consider **complex-valued networks**, in which inputs and weights are complex numbers, and so can be the output. For inputs, we take AC signals, and for weights we use impedance

components instead of resistors alone. *Complex-valued networks remain an active area of research.*

A scientific philological view is that the advancement of science is through the emergence of fundamental methodologies referred to as **paradigms**. Thousands of years ago, the scientific paradigm was empirical only, meaning that scientists simply described natural phenomena, such as observations of the night sky. Then, the theoretical paradigm came, thanks to Euclid, Newton, Einstein, and others. A few decades ago, we progressed to the computational era, and we began simulating complex processes. Today, recognized as the fourth paradigm, we heavily rely on machine learning and big data so that new knowledge can be automatically found from big data. The point is that when machines are able to learn and do, many jobs of ours can be done by machines!

In 2017, I participated a public debate in support of the statement that “machine learning will transform radiology significantly within the next 5 years”, which was published in Medical Physics as a “Point/Counterpoint” article (Wang, Kalra et al. 2017), which was ranked as Top 10 in terms of the number of downloads for the journal in 2017. I believe that machine learning and artificial intelligence has a transformative potential in the medical physics field, valid for both medical imaging and treatment planning. You may just read that debate article.

CHAPTER 7. IMAGE QUALITY

This chapter is the last piece of the foundation (necessary knowledge on mathematical/signal analysis), based on which we are well prepared to learn major tomographic modalities and their combinations. Since any imaging modality is intended to produce images that are supposed to be sharp, fine, and informative, **image quality assessment (IQA)** is an important aspect of medical imaging. Defining quality objectively is not straightforward, since quality in general is a context-sensitive concept. That being said, we have certain ways of defining image quality appropriately. While some ways are applicable to all types of images, others are more system-specific or task-specific. These will be explained in the following sections.

SECTION 7.1. GENERAL MEASURES

The simplest way to access image quality is to compute a **distance**, such as the **Euclidian distance**, between a medical image and the **ground truth**. The common measure is the **mean squared error (MSE)** between an image in question \mathbf{x} and a gold-standard reference \mathbf{y} and defined as

$$\text{MSE}(\mathbf{x}, \mathbf{y}) = \frac{1}{n} \sum_i^n (x_i - y_i)^2 \quad (7.1.1)$$

where i is the index through n pixels. MSE measures how different an image is from the reference. A lower MSE is expected to indicate a better image quality. It should be noted

that this measure has slightly different variants such as the **root mean squared error (RMSE)**:

$$\text{RMSE}(\mathbf{x}, \mathbf{y}) = \sqrt{\frac{1}{n} \sum_i^n (x_i - y_i)^2}, \quad (7.1.2)$$

the **mean absolute error (MAE)**:

$$\text{MAE}(\mathbf{x}, \mathbf{y}) = \frac{1}{n} \sum_i^n |x_i - y_i|, \quad (7.1.3)$$

the **mean absolute percentage error (MAPE)**:

$$\text{MAPE}(\mathbf{x}, \mathbf{y}) = \frac{1}{n} \sum_i^n \frac{|x_i - y_i|}{x_i}, \quad (7.1.4)$$

Note that these variations are often more desirable since the squared difference can be immensely large when the error is large.

The MSE is quite reasonable, as it is essentially a measure of the area between two curves or the measure of the volume between the two surfaces, which are a 2D image of interest and the reference image respectively. This measure is compatible to error analysis in a transformed/channelized space using coefficients of an orthonormal linear transform. Recall that Parseval's identity states that the squared area under a function is the same as the squared area of the

Fourier-transformed function:
$$\int_{-\infty}^{\infty} |f(t)|^2 dt = \int_{-\infty}^{\infty} |\hat{f}(s)|^2 ds.$$

Although all the above errors are useful, they often fail to agree with our visual perception, and cannot perform well in many practical applications. As shown in Figure 7.1-1, relative to the ground truth, all the compromised images suffer from the same amount of MSE but they give very different visual impressions.



Figure 7.1-1. Relative to the standard image (in the red box), the five compromised images (in the green box) all have a MSE of 225 but look very differently. On the other hand, the SSIM measure (in yellow to be explained below) effectively reflects the visual quality.

It can be seen in Figure 7.1-1 how all the compromised images differ from the standard in different ways. For example, the top two images in the green box are visually pleasing with minor variations. On the other hand, the bottom three images have poorer quality due to their blocky, blurring, or noise

appearances respectively. Therefore, *we cannot rely on MSE alone to access image quality.*

This brings us to a biomimetic perspective of image quality assessment based on the **human visual system (HVS)**. HVS extracts structural information and is adapted for contextual changes. This is why all the green-boxed images look so different despite the same MSE measure. This structurally-oriented assessment is a **top-down approach** in contrast to the pixel-wise comparison and overall error-pooling, which is a **bottom-up approach**.

Then, how to assess structural distortion in images? A classic result is the **structural similarity (SSIM)** (Wang, Bovik et al. 2004). This measure computes a structural distortion in an image that is consistent to HVS. Please look at Figure 7.1-1 again, and see how SSIM varies with visual image quality.

Figure 7.1-2 shows the mechanism by which the SSIM measurement works. SSIM incorporates three aspects of the difference between two images: **luminance**, **contrast**, and **structure**. To be clear, luminance is the measure of the average value of each image, contrast is a variability within an image. Once we normalize two images to the same luminance and contrast levels, we can compare their structures. Formally, we wish to incorporate these three aspects into a single measure:

$$S(\mathbf{x}, \mathbf{y}) = f(l(\mathbf{x}, \mathbf{y}), c(\mathbf{x}, \mathbf{y}), s(\mathbf{x}, \mathbf{y})) \quad (7.1.5)$$

where l is the luminance, c is the contrast, and s is the structure. When formulating such a similarity measure, we request the measure to satisfy the following **postulates**: symmetry, boundedness, and unique maximum:

1. **Symmetry:** $S(x, y) = S(y, x)$; (7.1.6)

2. **Boundedness:** $S(x, y) \leq 1$; (7.1.7)

3. **Unique Maximum:** $S(x, y) = 1$ if and only if the two images are the same. (7.1.8)

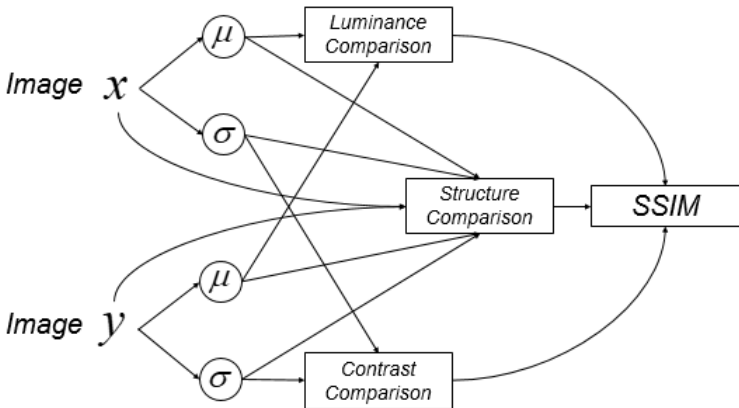


Figure 7-2. Structural Similarity (SSIM) integrates differences in luminance, contrast and structure, and forms a composition number between 0 and 1 to reflect an overall visual impression.

To find an SSIM value of an image relative to a reference image, we need to compute the mean and deviation of an image. The mean can be expressed as

$$\mu_x(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^N x_i \quad (7.1.9)$$

Note that if we remove the mean from an image, $\mathbf{x} - \mu_x$, we have a luminance-normalized image. The standard deviation can be computed as

$$\sigma_x(\mathbf{x}) = \left(\frac{1}{N-1} \sum_{i=1}^N (x_i - \mu_x)^2 \right)^{1/2} \quad (7.1.10)$$

Likewise, if we scale an image with its deviation, $\frac{(\mathbf{x} - \mu_x)}{\sigma_x}$, we have a contrast-normalized image.

First, let us perform the luminance comparison as follows:

$$l(\mathbf{x}, \mathbf{y}) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}, \quad (7.1.11)$$

where the constant C_1 is included to avoid instability when $\mu_x^2 + \mu_y^2$ is close to zero. Specifically, $C_1 = (K_1L)^2$, where K_1 is a very small constant, and L is the dynamic range of pixel values (for example, $L = 255$ for 8-bit grayscale images). To see how the above expression satisfies the postulates we have previously listed, we point out the following simple computation:

$$f(x) = \frac{2ax + c}{a^2 + x^2 + c},$$

$$f'(x) = \frac{2a(a^2 + x^2 + c) - 2x(2ax + c)}{(a^2 + x^2 + c)^2},$$

$$f'(x) = 0 \Leftrightarrow x = a.$$

Likewise, we will use a similar expression for the contrast comparison:

$$c(\mathbf{x}, \mathbf{y}) = \frac{2\sigma_x \sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}, \quad (7.1.12)$$

where $C_2 = (K_2 L)^2$, K_2 being a small constant like K_1 .

Note again how the above expression still satisfies the postulates brought up earlier. Note that for the same amount of contrast difference ($\Delta\sigma = \sigma_y - \sigma_x$), the contrast comparison will basically depend on the ratio between $\Delta\sigma$ and σ . This point can be appreciated by looking at the following estimation (for a small constant offset):

$$f(x, x + \Delta x) = \frac{2x(x + \Delta x) + c}{x^2 + (x + \Delta x)^2 + c} \approx 1 - \frac{1}{2} \left(\frac{\Delta x}{x} \right)^2.$$

In other words, a change in stimulus is measured with respect to this ratio, being consistent to HVS, which is commonly referred to as **Weber's law**, as shown in Figure 7.1-3.

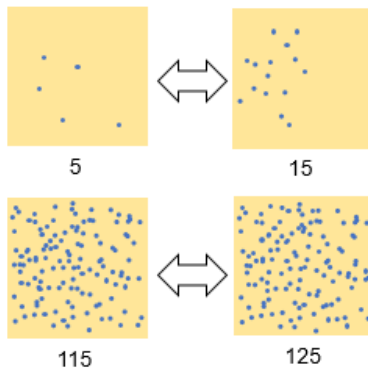


Figure 7.1-3. Example showing how the HVS interprets difference. The two images in the left column are in sharper contrast than the images in the right column even though the number of dots differs by the same amount (i.e., 10) in the two columns respectively.

Now, we are ready to compare structural similarity between two normalized images $(\mathbf{x} - \mu_x) / \sigma_x$ and $(\mathbf{y} - \mu_y) / \sigma_y$. The correlation between the two images is an effective measure for this purpose, based on the Cauchy-Schwarz Inequality Eq. (2.4.3). Since the correlation between the normalized images is the same as the **correlation coefficient** between the original images, the structural comparison can be done as follows:

$$s(\mathbf{x}, \mathbf{y}) = \frac{\sigma_{xy} + C_3}{\sigma_x \sigma_y + C_3}, \quad (7.1.13)$$

with the same analogous constant C_3 as previously discussed, and

$$\sigma_{xy} = \frac{1}{N-1} \sum_{i=1}^N (x_i - \mu_x)(y_i - \mu_y). \quad (7.1.14)$$

Note that the value of $s(\mathbf{x}, \mathbf{y})$ will be no more than 1 due to the Cauchy-Schwarz Inequality and reach 1 if and only if $\mathbf{x} = k\mathbf{y}$ where k is a constant.

Combining the above three Eqs. (7.1.11), (7.1.12) and (7.1.13) together, we obtain the SSIM as follows:

$$\text{SSIM}(\mathbf{x}, \mathbf{y}) = [l(\mathbf{x}, \mathbf{y})]^\alpha [c(\mathbf{x}, \mathbf{y})]^\beta [s(\mathbf{x}, \mathbf{y})]^\gamma, \quad (7.1.15)$$

where α , β , and γ are parameters to adjust the relative importance of the three components. This expression can be simplified by setting $\alpha = \beta = \gamma = 1$ and $C_3 = C_2 / 2$, yielding the following widely-used form of SSIM:

$$\text{SSIM}(\mathbf{x}, \mathbf{y}) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}. \quad (7.1.16)$$

The concept of SSIM has been extended in various ways, such as in the cases of color images, time-varying signals, and multi-scale analysis.

SECTION 7.2. SYSTEM-SPECIFIC INDICES

Since tomographic scanners produce images, we are interested in system specifications on various aspects of the nominal image quality. These specifications are often provided by manufactures and monitored by medical physicists for quality control. Technological advancements often lead to improved system-specific image quality indices.

First, data **noise** is what we do not want but it is unavoidable. The noise will demonstrates itself as random fluctuations, because of the probabilistic nature of the underlying physics, which is quantum mechanics in the cases of x-ray transmission and γ -ray emission. Due to data noise and imperfectness of imaging system components, images computed with noisy data necessarily contain noise as well. The **signal-to-noise ratio (SNR)** is typically defined as the ratio of the magnitude of a signal and the standard deviation of noise:

$$SNR = \frac{A_{Signal}}{\sigma_{Noise}}, \quad (7.2.1)$$

where A denotes the signal magnitude. Sometimes, RSN is also defined as the ratio between the power of the signal and the variance of the noise. Indeed, the **peak SNR (PSNR)** is defined as

$$PSNR = 10 \log \left(\frac{A_{Signal-Max}}{\sigma_{Noise}} \right)^2, \quad (7.2.2)$$

where $A_{Signal-Max}$ is the maximum signal or largest pixel value in an image. Clearly, information extraction will be more difficult with a lower SNR than in the case of a higher SNR. A noise-interfered image of “SIGNAL” is shown in Figure 7.2-1. A sister version of SNR is the contrast-to-noise ratio (CNR) defined as

$$CNR = \left| \frac{A_{Signal-1} - A_{Signal-2}}{\sigma_{Noise}} \right|, \quad (7.2.3)$$

which is focused on the difference between two signals, normalized by the noise deviation.

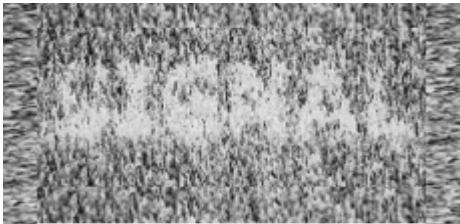


Figure 7.2-1. Signal embedded in a noisy background.

Second, various kinds of **image resolution** are used to indicate the imaging system’s resolving power. The target to be resolved by the imaging system can be structural details, shading differences, temporal changes, or spectral contents, which correspond to **spatial resolution** (or simply, resolution), **contrast resolution** (or, simply, contrast), **temporal resolution** (i.e., **time resolution**), and **spectral resolution** (or **energy resolution**) respectively. Quantitatively, resolution of any kind is a measure of the

smallest difference between two signals with which they can be differentiated.

For a point object modeled as a delta function, an imaging system will not be capable to capture it perfectly. The resultant image will be a blurred version of this point, which is called the **impulse response** or **point spread function (PSF)** of the imaging system. Then, **spatial resolution** is the distance between two bright spots at which they can be told apart and within which they become indistinguishable. Suppose that the spots are of bell shape, which is also referred to as Gaussian blurring, when they overlap too much they will be blurred together. Hence, it is heuristic to define spatial resolution as the **full width at half maximum (FWHM)**, as shown in Figure 7.2-2. When the separation is greater than FWHNM, the two spots are fused into one spot, and we cannot resolve them visually.

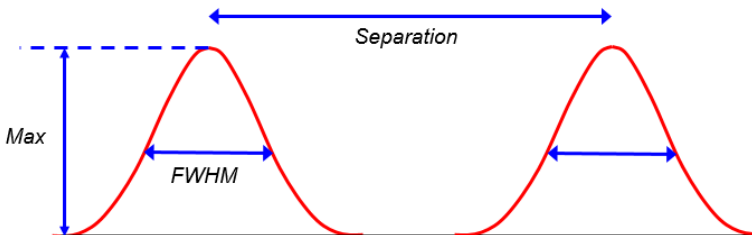


Figure 7.2-2. Image resolution defined as full width at half maximum (FWHM) of a point response function, which is often in a Gaussian form. When two points are separated less than FWHM away, they cannot be visually distinguished.

FWHM is only one of spatial resolution measures. Another good example is the **modulation transfer function (MTF)**.

With an ideal point/line object as an input, the output of the imaging system will be a blurred version of the original structure. The Fourier transform is then performed on the input and the output respectively. Their resultant Fourier spectra are $I(f)$ and $O(f)$ respectively. MTF is defined as

$$MTF(f) = \frac{|O(f)|}{|I(f)|} \quad (7.2.4)$$

where f is the spatial frequency. A typical MTF curve is shown in Figure 7.2-3. When the value of MTF goes below the noise level at a sufficiently high frequency, image details and noise fluctuations are no longer separable, suggesting the spatial resolution at that frequency; for example, in terms of line pairs per millimeter.

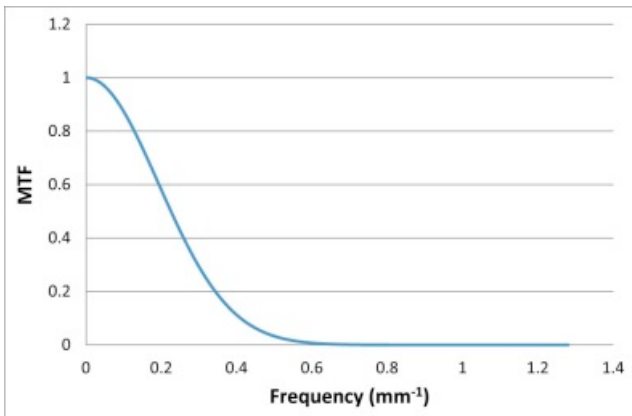


Figure 7.2-3. Normalized MTF curve.

Another important type of resolution is **contrast resolution**, which targets subtle shading differences. For example, nodules may look very similar to normal biological soft tissues in CT images, and they do not present in good contrast when image noise is strong or the imaging protocol is not optimized. Figure 7.2-4 shows significant differences between three radiograms in various contrasts.

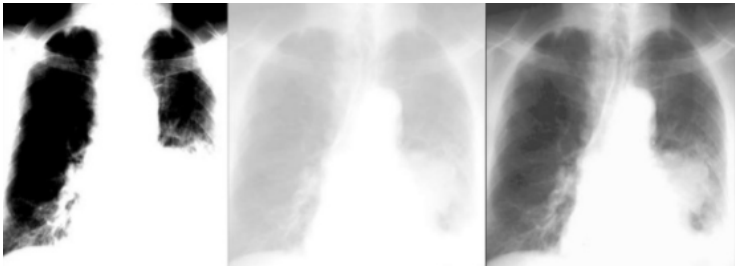


Figure 7.2-4. Three x-ray radiographic images from the same patient in high (Left), low (Middle) and optimum (Right) contrasts respectively.

Temporal resolution measures how quickly an imaging system can take a snapshot so that a moving structure, such as a beating heart, can be “frozen” to avoid motion blurring. For example, ultrasound imaging forms an image quickly, and is suitable to determine cardiac functions. In contrast, magnetic resonance imaging (MRI) cannot have both sharp spatial resolution and fast temporal resolution at the same time.

Spectral resolution is also highly relevant. For optical imaging, we can use various **fluorescent probes** to label biological biomarkers in different colors. Thus, we often produce colorful images, and do not want to be color-blinded. As a further example, x-ray images used to be on a gray-scale

but now equipped with the so-called **photon-counting detector** techniques, the **K-edge** of contrast tracers can be resolved for chemically-specific **material decomposition**.

Finally, the concept of **image artifacts** must be explained. Image artifacts are structures in images but these structures are not real. Hence, the artifacts are ghosts in the image domain. For example, when metallic implants are inside a patient, x-rays cannot penetrate them effectively, leading to poor data quality. When we reconstruct a CT image from compromised x-ray data, streaking artifacts will appear around the metal parts. These artifacts are not real but they hidden anatomical and pathological features. The sources and types of imaging modality-dependent image artifacts are many, and can be compensated for or eliminated using dedicated techniques. Figure 7.2-5 shows two cases, in each of which there are the true, uncorrected images, and images corrected using **metal artifact reduction (MAR)** algorithms including a state of the art method called “**normalized MAR**” (**NMAR**) and our in-house **convolutional neural network (CNN)** method.

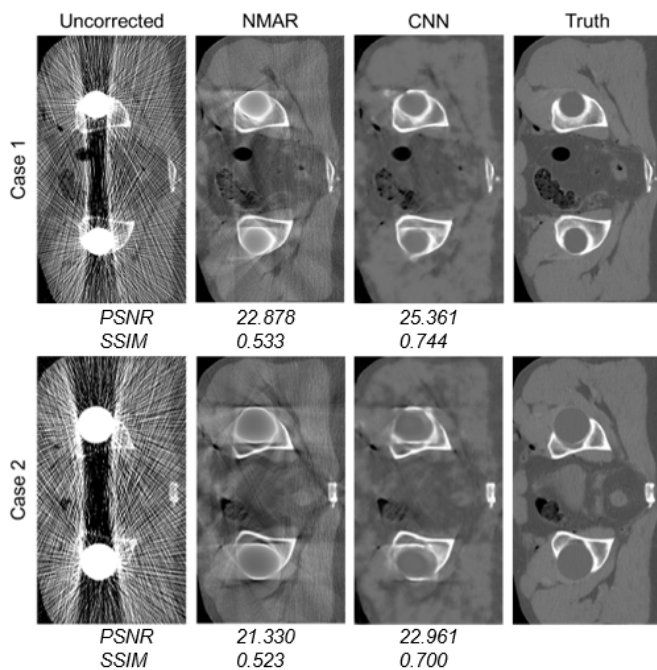


Figure 7.2-5. True, metal artifact affected, and corrected images.

SECTION 7.3. TASK-SPECIFIC PERFORMANCE

The specifications of an imaging system include multiple system-specific parameters/indices, and not all of them are equally important for a given diagnostic/interventional task. Most imaging studies are for a single disease/procedure, and image quality assessment should be task-based to optimize the diagnostic performance. In this section, we will introduce a few basic measures that are of direct clinical interest, and discuss how to optimize them.

After reading an image, an interpretation must be given, which can be correct or incorrect, depending on inherent image quality and also subsequent image analysis. As an example, let us say that we wish to detect a lung nodule in a patient. The only four scenarios are possible: (1) there is a nodule, and it is detected, known as a **true positive (TP)**; (2) there is no nodule, and it is confirmed as no nodule, known as a **true negative (TN)**; (3) there is a nodule present but is not detected and falsely reported as no nodule, known as a **false negative (FN)**, and (4) there is no nodule at all, but a tumor is falsely reported, known as a **false positive (FP)**. Depending on the task, false negatives and false positives can have varying degrees of adverse effects. For example, in the case of lung screening, a false negative is usually far worse than a false positive.

Two fundamental concepts in medicine are **sensitivity** and **specificity**. Sensitivity is defined as $TP/(TP+FN)$, where the denominator is the total number of all positive cases.

Sensitivity is the likelihood of detecting a pathological feature positively when the case is indeed positive; in other words, it is a measure of how sure we can say “**yes**”. On the other hand, **specificity** is defined as $TN/(TN+FP)$, where the denominator is the total number of all negative cases. **Specificity** is the likelihood of detecting a case a negative when the case is indeed negative; in other words, it is a measure of how sure we can say “**nope**”.

Two related concepts are **positive predictive value (PPV)** and **negative predictive value (NPV)**. PPV is defined by $TP/(TP+FP)$, where the denominator is the total number of all positively reported cases. PPV is the fraction of patients who have positive results actually are positive. On the other hand, NPV is defined as $TN/(TN+FN)$, where the denominator is the total number of all negatively reported cases. NPV is the fraction of patients who have negative results actually are negative. Furthermore, **diagnostic accuracy (DA)** is the ratio between the number of correctly reported cases and the number of patients, **prevalence (PR)** is the ratio between the number of positive cases and the number of involved subjects. To highlight how PPV and NPV are different from sensitivity and specificity, please look at Table 7.3-1. In this lung CT screening study, sensitivity is 22/30 (73.3%), specificity is 1,739/1,790 (97.2%), PPV is 22/73 (30.1%), NPV is 1,739/1,747 (99.5%), DA is (22+1,739)/1,820 (96.8%), and PR is 30/1,820 (1.6%).

	<i>Lung Nodule</i>		
CT	Yes	No	Total
Positive	22	51	73
Negative	8	1739	1747
Total	30	1790	1820

Table 7.3-1. Diagnostic analysis on lung CT screening results from 1,820 subjects.

Now, we are ready to discuss a practically important but slightly tricky tool: **receiver operating characteristic (ROC)** analysis. This tool relies on a curve using data of specificity and sensitivity to depict the **diagnostic performance** of an imaging study. The plot focuses on sensitivity as a function of “1-specificity” which is the rate of false positives or false alarms, as shown in Figure 7.3-1.

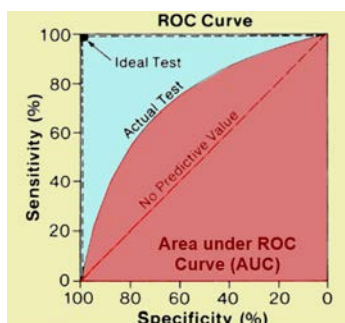


Figure 7.3-1. Receiver operating characteristic (ROC) curve with the horizontal and vertical axes for “1-specificity” (false positives or false alarms) and sensitivity respectively.

To understanding Figure 7.3-1 better, let us consider two extreme cases. First, what if we blindly claim all the subjects have diseases? In this case, all healthy patients are called diseased, and the rate of false positives will be 100%

(specificity will be 0%). On the other hand, sensitivity will be 100%, since no patient with disease will be called healthy. This case is the point at the top-right corner. Second, what if we blindly claim all patients are healthy? In this case, all diseased patients are called healthy, and the rate of false positives will be 0% (specificity will be 100%), but sensitivity will be 0%, since no healthy patient is called diseased. This case is the point at the bottom-left corner. Third, we can just flip a coin to decide if we call a subject healthy or diseased. Then, both sensitivity and specificity will be 50%.

Generally, a population is categorized into two groups: healthy and diseased. In an ideal case, there is a large gap between the readings from the two groups, making a clear distinction between those who are healthy and those who are diseased. However, often times the distribution has a significant overlap, resulting in no clear cut between the two classes. Due to this overlap, different thresholds can be set to achieve varying levels of sensitivity and specificity. In other words, many possible combinations of sensitivity and specificity are possible.

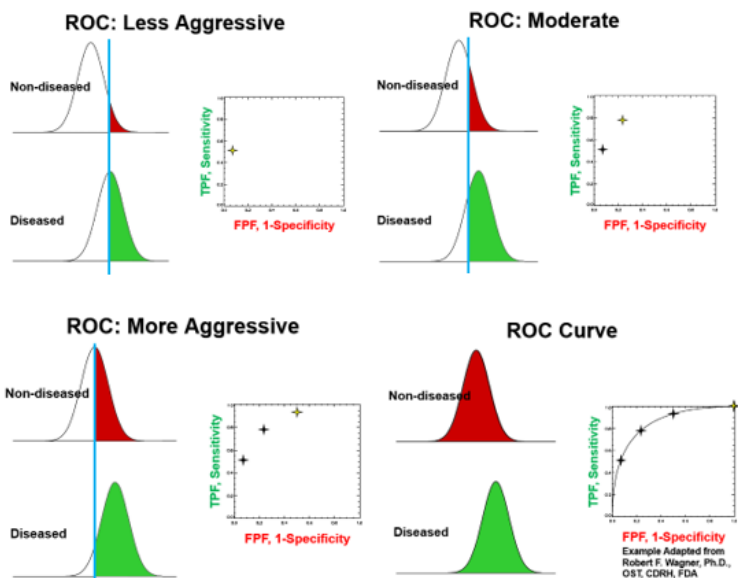


Figure 7.3-2. Area under the ROC curve as a good measure of the diagnostic performance.

As shown in Figure 7.3-2, as different thresholds are chosen, the diagnostic decision will lead to corresponding selectivity and specificity. Each decision rule (threshold) will define a unique point in the sensitivity-specificity plane. All such points form the ROC curve as a representation of the diagnostic performance. Better ROC curves will be those bulging farther to the upper left corner, having greater areas under the ROC curve.

It should be underlined that ROC analysis can be applied to not only assess image quality but also evaluate doctors' expertise or the capability of image analysis software in extracting

diagnostic information from images. In this way, we can measure individual performance with regard to detecting diseases, since each doctor has his/her own interpretation and therefore his/her own ROC curve. Similarly, we can compare computer-aided diagnostic programs with trained radiologists. In all these cases, better doctors or higher quality software packages will follow a ROC curve closer to the upper left corner. It is widely believed that the day is near that artificial intelligence will outperform average radiologists in most diagnostic tasks (Wang, Kalra et al. 2017)!

To appreciate how a machine can do a radiologist's job, we should briefly describe an algorithmic mechanism called a **model observer**. Let us start with a **linear imaging system** model, which is a good approximation for a majority of imaging systems and typically in a discrete form:

$$\mathbf{g} = \mathbf{H}\mathbf{f} + \mathbf{n} \quad (7.3.1)$$

where \mathbf{f} is an object being imaged, \mathbf{H} is an imaging operator that represents an imaging system, \mathbf{n} is a noise background from the imaging process, and \mathbf{g} is an image vector. Then, we say that there are two cases to be reported based on the image produced by the imaging system: either positive or negative (for example, there is a nodule or not). This **binary classification** task can be formulated as a statistical **hypothesis testing** problem:

$$\begin{aligned} \mathbf{H}_0 &= \mathbf{H}\mathbf{f}_b + \mathbf{n}; \\ \mathbf{H}_1 &= \mathbf{H}(\mathbf{f}_b + \mathbf{f}_s) + \mathbf{n}, \end{aligned} \quad (7.3.2)$$

where \mathbf{H}_0 is the outcome from a negative case, \mathbf{H}_1 is the outcome from a positive case, \mathbf{f}_b is the background image (or the features of an image that do not determine if the outcome is positive or negative), and \mathbf{f}_s is a signal (or the features of interest whose presence indicates a positive result, such as a nodule).

In an ideal case, the **ideal observer** utilizes all statistical information available regarding the diagnostic task to maximize the diagnostic performance. The ideal observer should extract all relevant features, utilize all prior knowledge, and achieve the best diagnostic performance as measured by the area under the ROC curve among all the observers, based on the same data \mathbf{g} . It is conceptually clear that the ideal observer should make its decision according to the likelihood ratio:

$$\Lambda(\mathbf{g}) = \frac{f(\mathbf{g} // \mathbf{H}_1)}{f(\mathbf{g} // \mathbf{H}_0)}, \quad (7.3.3)$$

where $f(\mathbf{g} // \mathbf{H})$ is probability density functions of the image under a hypothesis.

For simplicity, we can assume our decision process to be a linear system targeting the maximum SNR. The resultant optimal observer is called the **Hotelling observer**. This is important as often times an ideal observer is not possible as one does not have access to all the required statistics. A **channelized Hotelling observer**, on the other hand, will

perform the same task as the human observer but it will only utilize certain features of an image instead of the whole image.

SECTION 7.4. REMARKS

We have discussed three aspects of image quality assessment. All the measures are defined to quantify how well an imaging system performs but in different settings. For example, the least squared measures tell how close it is between an acquired image and the reference, SSIM shows structural similarity as the name SSIM indicates, and the area under the ROC curve is specific to clinical tasks.

From the information theoretic perspective, when an imaging system produces an image. A measurement is done on an underlying object, and inherent information in the image needs to be extracted by a radiologist, a pathologist, or a computer program. Given the rapid development of machine learning techniques, computerized intelligent image readers are being actively developed, and will gradually replace human experts. This whole workflow includes multiple steps from data acquisition, through image reconstruction, image processing and analysis, to the final report. Each of the steps should be optimized to maximize the information content and/or minimize the information loss.

Information theory is an important branch of modern science, which was pioneered by Claude E. Shannon to find the fundamental capability and limit on signal processing and communication (Shannon 1997). Random variables or processes are often used as mathematical models of information theory.

Among important concepts in information theory, **entropy** and **mutual information** are particularly relevant. The material in this sub-section describes the essential idea of the information theory, and demands some abstract thinking and substantial efforts to digest well.

The entropy for a discrete random variable is defined as

$$H_x = \sum_{x \in X} p(x) \log_2 \frac{1}{p(x)}, \quad (7.4.1)$$

where the discrete random variable x is defined in its space X with a probability $p(x)$. The meaning of the second factor

$\log_2 \frac{1}{p(x)}$ on the right-hand side of Eq. (7.4.1) is the number

of **bits** necessary to represent permissible values of the variable x . If you toss a coin, you have two possible events, and you only need 1 bit to record your outcome (0 for head, and 1 for tail). If you play a dice in the shape of a regular octahedron, you have eight equally likely results, and you need at least 3 bits to record the outcome in this case. Of course, if you play a regular dice which has six faces, you still need 3 bits, since the bit is the minimum unit of the digital memory. With the

probability $p(x)$ as the weighting factor for $\log_2 \frac{1}{p(x)}$, the

entropy, as a statistical expectation, indicates the minimum number of bits needed to describe the random variable x . Hence, the larger the entropy, the more the uncertainty associated with the random variable, as measured by the

minimum amount of digital memory for coding all possible outcomes.

For two discrete random variables x and y in their domains \mathbf{X} and \mathbf{Y} with probabilities $p(x)$ and $p(y)$ as well as **joint probability** $p(x, y)$, we have the **conditional probability distributions** as follows:

$$p(x/y) = \frac{p(x, y)}{p(y)}, \text{ and } p(y/x) = \frac{p(x, y)}{p(x)}. \quad (7.4.2)$$

If the two random variables are independent, each conditional probability is independent of the conditioned variable. If the variables are dependent, *how should we measure their dependency?*

From the information theoretic perspective, the answer is to compute how much information we obtain on one variable after we know the value of the other variable. This consideration leads to the concepts of **conditional entropy**, **total entropy**, **mutual information**, and **Kullback-Leibler distance**.

The conditional entropy $H_{Y/X}$ and total entropy $H_{X,Y}$ are naturally defined as

$$H_{Y/X} = - \sum_{x \in X} p(x) \sum_{y \in Y} p(y/x) \log_2 p(y/x), \quad (7.4.3)$$

$$H_{X,Y} = \sum_{x \in X, y \in Y} p(x, y) \log_2 p(x, y). \quad (7.4.4)$$

If you are insightful enough, it should be understandable that the total entropy should be the sum of the conditional entropy of one variable and the entropy of the conditioned variable; that is,

$$H_{x,y} = H_x + H_{y/x}. \quad (7.4.5)$$

Indeed, if we have a composite system \mathbf{Z} that consists of two independent subsystems \mathbf{X} and \mathbf{Y} , with three corresponding random variables x, y , and z with probabilities $p(x), p(y)$ and $p(z)$, then $p(z)=p(x)p(y)$. Then, the total entropy must be the sum of the component entropies:

$$\begin{aligned} H_z &= \sum_{z \in \mathbf{Z}} p(z) \log_2 p(z) \\ &= \sum_{x \in \mathbf{X}, y \in \mathbf{Y}} p(x)p(y) \log_2 [p(x)p(y)] \\ &= \sum_{x \in \mathbf{X}} p(x) \log_2 p(x) + \sum_{y \in \mathbf{Y}} p(y) \log_2 p(y) \quad (7.4.6) \\ &= H_x + H_y. \end{aligned}$$

To prove Eq. (7.4.5), we only need to point out that $p(x, y) = p(x)p(y/x)$ from Eq. (7.4.2).

If the two variables are independent, then $H_{y/x} = H_y$; but generally $H_{y/x} < H_y$ (i.e., we would have some information about y if we know x if the two variables are statistically relevant). Hence, we can define the **mutual information** as

$$I_{x,y} = H_y - H_{y/x} = H_x - H_{x/y}, \quad (7.4.7)$$

Eq. (7.4.7) can be re-formulated as follows:

$$\begin{aligned}
 I_{X,Y} &= \sum_{y \in Y} p(y) \log_2 \frac{1}{p(y)} + \sum_{x \in X} p(x) \sum_{y \in Y} p(y/x) \log_2 p(y/x) \\
 &= H_Y + \sum_{x \in X, y \in Y} p(x, y) \log_2 p(y/x) \\
 &= H_Y + \sum_{x \in X, y \in Y} p(x, y) \log_2 \frac{p(x, y)}{p(x)} \\
 &= H_Y + H_X + H_{X,Y} \\
 &= \sum_{x \in X, y \in Y} p(x, y) \log_2 \frac{p(x, y)}{p(x)p(y)}
 \end{aligned}$$

That is,

$$I_{X,Y} = \sum_{x \in X, y \in Y} p(x, y) \log_2 \frac{p(x, y)}{p(x)p(y)}. \quad (7.4.8)$$

Geometrically, the mutual information measure how far a joint probability distribution is from an independent distribution given by the product of the marginal distributions.

Motivated by the concept of mutual information, we define the **Kullback-Leibler distance** between two probability distributions $p(x)$ and $q(x)$ in the same spirit of Eq. (7.4.8):

$$D(q // p) = \sum_{x \in X} q(x) \log \left(\frac{q(x)}{p(x)} \right), \quad (7.4.9)$$

with the convention that $0 \log(0) = 0$ and $0 \log\left(\frac{0}{0}\right) = 0$. It can be shown that $D(q//p)$ is convex with respect to $q(x)$, nonnegative, and equal to zero if and only if $p(x)$ and $q(x)$ are indistinguishable. Hence, $D(q//p)$ is qualified as a distance measure although it is not symmetric, i.e., $D(q//p) \neq D(p//q)$.

In Sub-section 7.1, we have described least squared measures as Euclidean distances. There are actually many types of distances, and by any measure the Kullback-Leibler distance is among the most important one; for example, in image processing (registration) and machine learning. Also, in Sub-section, we have introduced the SSIM, which involves a correlation coefficient Eq. (7.1.14) to measure the structural similarity. While the correlation coefficient only reflects the linear correlation, the mutual information Eq. (7.4.8) effectively captures the nonlinear dependency in the fundamental way.

ABOUT THE AUTHOR



Ge Wang is the Clark & Crossan Endowed Chair Professor and the Director of the Biomedical Imaging Center, Rensselaer Polytechnic Institute, Troy, NY, USA. He authored the first spiral cone-beam/multi-slice CT reconstruction paper in 1991. Currently, there are over 100 million medical CT scans yearly with a majority in the helical cone-beam/multi-slice mode. He and his collaborators published the first paper on bioluminescence tomography. His group published the first papers on interior tomography and omnitomography (“*all-in-one*”) to acquire diverse datasets simultaneously (“*all-at-once*”) with CT-MRI as an example. His results were featured in *Nature*, *Science*, and *PNAS*, and recognized with academic awards. He wrote over 430 peer-reviewed journal papers. His team has been in collaboration with world-class groups and continuously well-funded. He is the Lead Guest Editor of five *IEEE Transactions on Medical Imaging* Special Issues, the founding Editor-in-Chief of *International Journal of Biomedical Imaging*, and an Associate Editor of *IEEE Transactions on Medical Imaging*, *Medical Physics*, *IEEE Access*, and other journals. He is Fellow of the *IEEE*, *SPIE*, *OSA*, *AIMBE*, *AAPM*, and *AAAS*.

References

Atkinson, N. (2018). from

<https://www.seeker.com/space/astrophysics/a-new-theory-explains-why-the-universe-is-three-dimensional>.

Donoho, D. L. and C. Grimes (2003). "Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data." Proceedings of the National Academy of Sciences of the United States of America **100**(10): 5591-5596.

Einstein, A. (2005). The meaning of relativity. Princeton, Princeton University Press.

Eldar, Y. C. and G. Kutyniok (2012). Compressed sensing : theory and applications. Cambridge ; New York, Cambridge University Press.

Feynman, R. P., R. B. Leighton and M. L. Sands (2011). The Feynman lectures on physics. New York, Basic Books.

Gallager, R. (2006). Course materials for 6.450 Principles of Digital Communications I, Fall 2006. MIT OpenCourseWare(<http://ocw.mit.edu/>), MIT.

Han, W., H. Yu and G. Wang (2009). "A general total variation minimization theorem for compressed sensing based interior tomography." Int J Biomed Imaging **2009**: 125871.

ImageNet. (2018). from <http://www.image-net.org/>.

MATLAB-CircularConvolution. (2018). from <https://www.mathworks.com/help/signal/ug/linear-and->

[circular-convolution.html?requestedDomain=true&nocookie=true](#).

MATLAB-SpectralAnalysis. (2018). from <https://www.mathworks.com/help/signal/ug/amplitude-estimation-and-zero-padding.html?requestedDomain=true>.

Oppenheim, A. V., A. S. Willsky and S. H. Nawab (1997). Signals & systems. Upper Saddle River, N.J., Prentice Hall.

Osgood, B. G. (2018). from <https://see.stanford.edu/Course/EE261>.

Roweis, S. T. and L. K. Saul (2000). "Nonlinear dimensionality reduction by locally linear embedding." Science **290**(5500): 2323-+.

Schlueter, F. J., G. Wang, P. S. Hsieh, J. A. Brink, D. M. Balfe and M. W. Vannier (1994). "Longitudinal image deblurring in spiral CT." Radiology **193**(2): 413-418.

Shannon, C. E. (1997). "The mathematical theory of communication. 1963." MD Comput **14**(4): 306-317.

ShareTechNote. (2018). from http://www.sharetechnote.com/html/Handbook_EngMath_Chaos_LogisticEq.html.

Signal-Processing-Stack-Exchange. (2018). from <https://dsp.stackexchange.com/questions/19522/additive-but-not-homogeneous-continuous-system>.

Sontag, E. D. (1998). Mathematical control theory : deterministic finite dimensional systems. New York, Springer.

Suetens, P. (2009). Fundamentals of medical imaging. New York, Cambridge University Press.

Wang, G. (2016). "Perspective on deep imaging." IEEE Access.

Wang, G., M. Kalra and C. G. Orton (2017). "Machine learning will transform radiology significantly within the next 5 years." Med Phys **44**(6): 2041-2044.

Wang, Z., A. C. Bovik, H. R. Sheikh and E. P. Simoncelli (2004). "Image quality assessment: from error visibility to structural similarity." IEEE Trans Image Process **13**(4): 600-612.

Wikipedia-FFT. (2018). from https://en.wikipedia.org/wiki/Fast_Fourier_transform.

Wikipedia-Hilbert-Hotel. (2018). from https://en.wikipedia.org/wiki/Hilbert%27s_paradox_of_the_Grand_Hotel.

Wikipedia-Linear-System. (2018). from http://en.wikipedia.org/wiki/Linear_system.

Wikipedia-Multiverse. (2018). from <https://en.wikipedia.org/wiki/Multiverse>.

Wikipedia-Scalar. (2018). from [http://en.wikipedia.org/wiki/Scalar_\(mathematics\)](http://en.wikipedia.org/wiki/Scalar_(mathematics)).

Wikipedia-Vector-Space. (2018). from https://en.wikipedia.org/wiki/Vector_space.

Wray, M. (2018). from <http://www.michw.com/tag/matlab/>.

Xu, Q., H. Yu, X. Mou, L. Zhang, J. Hsieh and G. Wang (2012). "Low-dose X-ray CT reconstruction via dictionary learning." IEEE Trans Med Imaging **31**(9): 1682-1697.

Yang, J., H. Yu, M. Jiang and G. Wang (2010). "High Order Total Variation Minimization for Interior Tomography." Inverse Probl **26**(3): 350131-3501329.

Yu, H. and G. Wang (2009). "Compressed sensing based interior tomography." Phys Med Biol **54**(9): 2791-2805.

Yu, H., J. Yang, M. Jiang and G. Wang (2009). "Supplemental analysis on compressed sensing based interior tomography." Phys Med Biol **54**(18): N425-432.