LESSONS ON INFORMATION IN TERMS OF THE FOURIER TRANSFORM FOR PRIMARY-AGED STUDENTS

Ikeda, Daisuke
Faculty of Information Science and Electrical Engineering, Kyushu University

https://hdl.handle.net/2324/4495647

出版情報：INTED2018 Proceedings, pp.3734-3739, 2018-03-05. IATED
バージョン：
権利関係：

LESSONS ON INFORMATION IN TERMS OF THE FOURIER TRANSFORM FOR PRIMARY-AGED STUDENTS

D. Ikeda

Kyushu University (JAPAN)

Abstract

Computer science education for kids has received considerable attention in recent years. In fact, many visual languages for kids has recently developed, such as ScratchJr. Although computer science has a wide range of sub-fields, we assume that it consists of the following three basic fields: computational science, such as programming, and informational science, such as digitization, and artificial intelligence, such as machine learning. The recent attention mainly focuses on topics in computational science, but not those in information science, which is one of the fundamental theory of computer science. In this paper, the author shares experiences of the lessons on part of informational science for students from elementary school to university. One big feature of the lessons is that they are based on the Fourier transform, which is generally learned by university students in many departments in faculties of science, engineering, and computer science, and is difficult for young kids to understand. Despite of the difficulty, the author strongly believe that it can provide different perspectives of information theory, depending on knowledge students already have. Another merit of the Fourier transform is that we can use many hands-on materials, such as spectroscopes, and phenomena, such as rainbows, in our daily life. In this paper, we explain the outline of the proposed lectures and materials, and discuss merits of the lectures for kinds, based on the Fourier transform.

Keywords: Information Theory, Fourier Transform, JavaScript Application, Hands-on Experience

1 INTRODUCTION

Computer science education for kids has received considerable attention in recent years. In fact, the former US president Barack Obama said [3]:

> Don't just buy a new videogame; make one. Don't just download the latest app; help design it. Don't just play on your phone; program it. No one's born a computer scientist, but with a little hard work and some math and science, just about anyone can become one.

In Japan, the government said that programming would be introduced as required subjects into primary and secondary education in 2020.

To support these movements, many visual languages for kids has recently been developed, such as Scratch (https://scratch.mit.edu/) and Scratch Jr (https://www.scratchjr.org/). Google launched a similar tool, called Blockly (https://developers.google.com/blockly/), for students and teachers. LEGO's counterpart is WeDo 2.04 (https://education.lego.com/en-us/downloads/wedo-2) makes full use of its hands-on materials, Lego bricks, not only for programming but also general scientific classes. Not only programming languages, materials to teach computer science for kids have been developed, such as [1, 2] and CS Fundamentals Unplugged (https://code.org/curriculum/unplugged).

Although computer science has a wide range of sub-fields, we assume that it consists of the following three basic fields: computational science, such as programming, and informational science, such as digitization, and artificial intelligence, such as machine learning. This is because our university offers computer science education for first-year students based on the assumption. From this assumption, the recent attention mainly focuses on topics in computational science, but not those in information science, which is one of the fundamental theory of computer science.

In this paper, the author shares experiences of the lessons on part of informational science for students from elementary school to university. These lessons were provided not at the university but at events in our open campus or science fairs.

One big feature of the lessons is that they are based on the Fourier transform. From mathematical perspective, the Fourier transform decomposes a function into simpler trigonometric functions. Equally, the stimulus is represented as the superposition of infinitely many sinusoids, that is the superposition principle of physics. In general, the Fourier transform is learned by university students in many departments in faculties of science, engineering, and computer science, since it is necessary for students to understand various mathematical notions, such as trigonometric functions, infinite sequences, and integral calculus. Therefore, it is too difficult for young kids to learn the Fourier transform.

Despite of the difficulty of the Fourier transform, the author strongly believe that it can provide different perspectives of information theory, depending on knowledge students already have. For example, we can see that the DC component of the Fourier transform is the average of an input signal and the other components the other averages for different granularities. Thus, to understand this idea, it is only required to know the notion of the average, which is taught at elementary schools. As target students become older, we can add difficult parts from the Fourier transform to lessons.

Another merit of the Fourier transform is that it is deeply combined with physical phenomena in our daily life. Essentially, once we have done the Fourier transform, we obtain a frequency domain representation, which corresponds to a set of waves or sounds, in case of light or sound. Therefore, we can use many class materials to develop hands-on experiences for kids.

In this paper, first we will briefly explain the basic notion of the Fourier transform. Then we will show the outline of the proposed lecture and materials for the lecture.

2 THE FOURIER TRANSFORM

In this section, we will explain the basic idea of the Fourier series, which is a specific version of the Fourier transform, without giving the detailed mathematical notions. From mathematical perspective, the Fourier series decomposes a function into simpler trigonometric functions.

Any function $\mathrm{f(x)}$ can be decomposed in a Fourier series, like the following:

$$\mathrm{f(x)} = \frac{a_0}{2} + \sum_{k=1}^{\infty} a_k \cos kx + b_k \sin kx \ , \quad (1)$$

where a_0, a_k, b_k are called Fourier coefficients and defined as follows:

$$a_0 = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)dx,$$

$$a_k = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\cos kx\, dx, \text{ and}$$

$$b_k = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\sin kx\, dx.$$

Of course, although there exists some requirements for a function, we won't go into further details.

To understand the meaning of the Fourier series, first let us begin with the Cartesian coordinate system, or orthogonal coordinate system. In case of 1 dimension, we use just one number. Next, in case of 2 dimensions, we use a pair of numbers, such as (2.5, 4.7). That is, $R^2 = \{(x, y)|\ x, y \in R\}$, where R is the set of real numbers. Similarly, we can consider 3 dimensions, like $(\mathrm{x, y, z})$. Naturally, we consider this space as our physical space.

Beyond our physical space, we can also consider n dimensional space, like $(x_1,\ x_2, \cdots,\ x_n)$ for $\mathrm{n} \geq 4$. This is a concrete mathematical notion, but it is difficult to imagine, based on the natural correspondence between 3 dimensional space and our physical space. Instead of this, we use the block representation, where the height of each block corresponds to one element of a vector. For example, for a point (4,2,0,2) in 4 dimensional space, we obtain four blocks in Fig. 1. In other words, a vector corresponds to a shape of four blocks, using this representation. The more rows of blocks we use, the more detailed shape can be depicted.

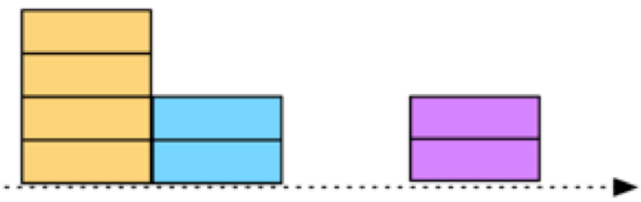

Figure 1. A block representation of (4, 2, 0, 2).

Along this line of thinking, we obtain an analog representation as the representation of infinitely many numbers, like Fig. 2. Now we reach the idea that a (continuous) function is represented as a vector in an infinite dimensional space. In this representation, a shape of the given function corresponds to the infinite sequence of numbers of functions' values.

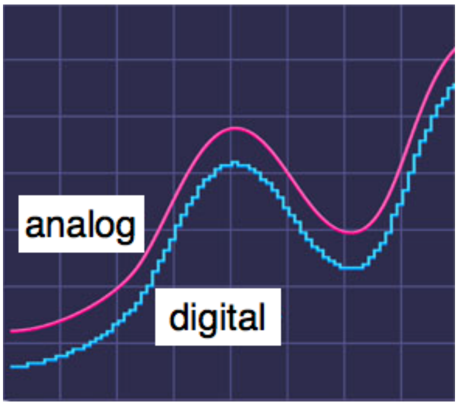

Figure 2. A continuous function is an analog representation and the corresponding digital representation can be expressed by a finite sequence of blocks.

Now we check the Fourier series, defined by equation 1. This equation transforms a function, or equally a vector with an infinite dimension, into $a_0,\ a_k, b_k$. These Fourier coefficients are a sequence of infinitely many numbers, that is, they define a vector in another infinite dimensional space. A meaning of this transformation will be explained in the next section.

You might think that it is difficult to imagine infinite dimensional spaces. But we often see real phenomena of such transformation in the real world. For example, a rainbow is the output of such a transformation, that is, given a white light, each coefficient of the Fourier series corresponds to each single light, like Fig. 3.

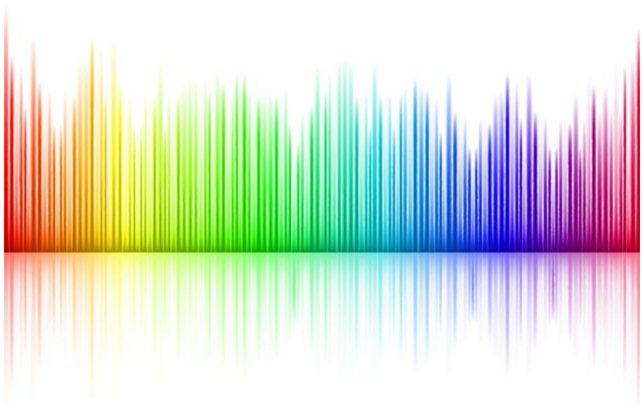

Figure 3. A series of single light with a specific frequency contained in a white light.

3 THE PROPOSED LECTURE AND MATERIALS

In this section, the proposed lecture and materials are explained.

The basic story of the lecture is that we would like to send a shape of blocks, like the shape in Fig. 1, assuming a sender and a receiver. We begin with the mechanism of sending images with a facsimile. When the sender wants to send the picture of an umbrella, the left-hand side of Fig. 4. In the facsimile, the picture is translated into a digitized image, like the right-hand side of Fig. 4, and the receiver obtained this image. This transformation is called the analog-digital transformation.

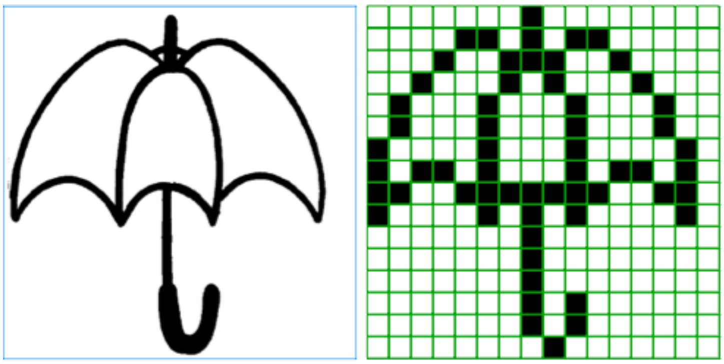

Figure 4. A source image (the left-hand side) and the ditized image (the right-hand side)

A straightforward representation of the digitized image is the block representation, like Fig. 1, or equally the vector representation, like (4, 2, 0, 2), in case of four dimension. Students can see that any shapes with the four rows of blocks can be built using four colours of blocks. For senior high school students who have already learned vectors, we also explain that a block of one colour corresponds to a base vector, such as (1, 0, 0, 0). These types of base vectors are called trivial base vectors.

Using four colours blocks or trivial base vectors, we can construct any shapes in four dimensions. The next step in this lecture is the following question: “Are there any other base vectors to represent any shape in this dimension?” But, since it is difficult for students to get an answer for that, we begin with asking students what if we can use only one digit to represent a shape in this dimension. Given a shape of (4, 2, 0, 2), we suppose 2 is the answer for that because 2 is the average of the four digits, 4 + 2 + 2 + 0/4 = 2. Therefore, the receiver can only sketch the shape in which all rows have the same height of two, (2, 2, 2, 2). like the right-hand side of Fig. 5 if the sender builds the shape of the left-hand side.

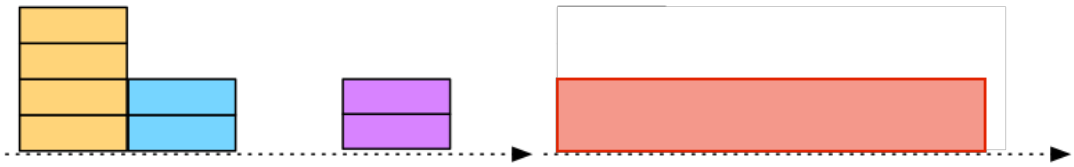

Figure 5. A source shape of blocks (the left-hand side) and (2, 2, 2, 2) showing the average of all rows, (4, 2, 0, 2) (the right-hand side)

Next, we consider what if we can use only two digits, including the average. We suppose 1 is the answer when (4, 2, 0, 2) is sent and we assume the left-hand side is positive. The receiver sketches the shape like the right-hand side of Fig. 6 when 2 and 1 are sent, where (3, 3, 1, 1) represents two averages of the left two rows (4, 2) of the source shape, and the other rows (0, 2), respectively. When the receiver receives (2, 1), it rebuilds the shape, by adding 1 to 2 at the left two rows and by subtracting 1 to 2 at the right two rows, and obtains the shape in transparent blue since we have assumed the left-hand side is positive. Compared to the sketch of (2, 2, 2, 2), (3, 3, 1, 1) can convey a more detailed shape.

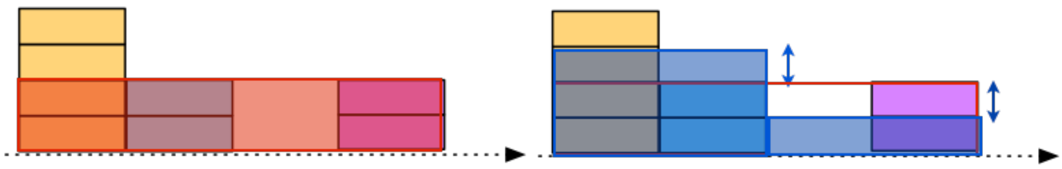

Figure 6. The average (2, 2, 2, 2) of (4, 2, 0, 2) (the left-hand side), and (3, 3, 1, 1) of two averages of two rows (4, 2) and the other rows (0, 2), respectively (the right-hand side)

Next, we consider what if we can use three digits, including the above two digits. In this case, 1 is the answer when (4, 2, 0, 2) is sent and we assume the tow outer rows are positive. The receiver sketches the shape like the right-hand side of Fig. 7, by adding 1 to the two outer rows and subtracting 1 to the two inner rows from (3, 3, 1, 1). Finally, the receiver can rebuild (4, 2, 0, 2). Although we need four digits to represent a shape in 4 dimension in general, three digits are enough for this shape.

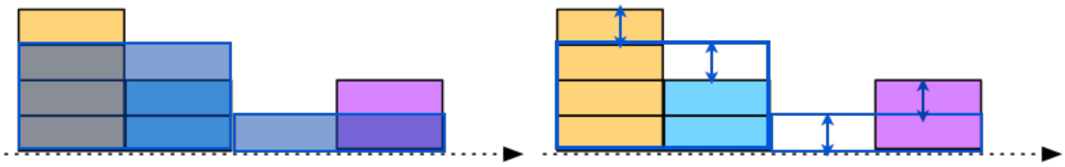

Figure 7. (4, 2, 0, 2)(the right-hand side) is obtained by adding 1 to two outer rows of (3, 3, 1, 1) (left-hand side) and subtracting 1 from two inner rows of it.

Eventually, the sender sends (2, 1, 1, 0) to the receiver, given (4, 2, 0, 2). These four elements are obtained by the inner product of (4, 2, 0, 2) with the following four vectors:

$$\frac{1}{4}(1,1,1,1), \frac{1}{4}(1,1,-1,-1), \frac{1}{4}(1,-1,-1,1), \frac{1}{4}(1,-1,1,-1)$$

For example, the inner product of (4, 2, 0, 2) with $^1/_4\,(1,1,-1,-1)$

$$\frac{1}{4}(1,1,-1,-1)\cdot(4,2,0,2) = \frac{1}{4}(4+2+0-2) = 1$$

In other words, (2, 1, 1, 0) is a vector or a point in the vector space spanned by the above four bases, where they are orthogonal with each other. Except for constants, they are equivalent to Hadamard bases.

To understand any shape in 4 dimensions can be represented in this way, the author developed a Web application using HTML, CSS, and JavaScript. You can use the application from a browser of PCs, tablets, and smart phones, accessing http://bit.ly/2AhTwq3, or equally to http://www.i.kyushu-u.ac.jp/~daisuke/ fourier.html. A screenshot of this application is shown in Fig. 8. Please note that its size is designed to fit iPad.

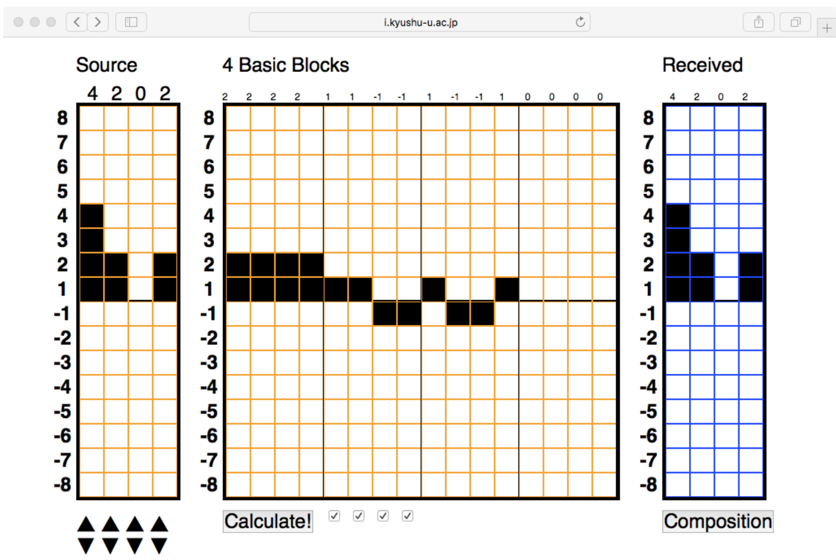

Figure 7. A screenshot of the developed Web application..

Using this application, students can learn essences of the Fourier transform, where a shape of building blocks is given at "Source", four components after the transformation are shown at "4 Basic Blocks" by clicking "Calculate!" button, and a superposed shape of checked components is shown at "Received" by "Composition" button. The more components we check, the closer to the source shape the received one is.

Using only positive numbers, we only have trivial bases. But if we are allowed to use negative numbers, we can use other bases, like the above ones. We can obtain such bases by rotating trivial bases. Consider trivial bases (1,0) and (0,1) in 2 dimensions for example. By rotating them 45 degrees counter-clockwise, (−1,1) and (1,1) are obtained, where We omit the constant of $\frac{1}{\sqrt{2}}$.

In the block representation, students can grasp the meaning of negative numbers graphically since, for a vector v, we obtain the vector with the opposite direction by multiplying (-1) to this vector. Therefore, students intuitively understand the reason why (−1) × (−1) = 1.

The derived four bases correspond to the direct current component 4 (constant), $\sin x$, $\cos x$, and $\sin 2x$. These trig functions are smooth curves, and they correspond to analog data. Students can see smooth curves of our voices or any sounds, with oscilloscopes.

The fourth vector of the derived four bases are at twice the frequency of the second and third one. In other words, bases vectors with higher frequency can depict detailed shapes. In the real world, a frequency of a light corresponds to a single light. We can have an experience of this correspondence using a prism, spectroscope, or a rainbow.

Thus, the more bases we use, the more detailed shapes can be depicted. However, we can tell a difference between two shapes only if the difference is enough large since our recognition has a valid range of frequency. Therefore, we can eliminate elements of a vector with higher frequencies. This is the basic principle of compression methods, like MP3.

ACKNOWLEDGEMENTS

This work was supported by JSPS KAKENHI Grant Number 15H02787.

REFERENCES

[1] T. Bell, I. H. Witten, and M. Fellows. “CS Unplugged: An enrichment and extension programme for primary-aged students”, Computer Science Unplugged, v3.0 edition, 2015.

[2] T. J. Cortina. “Reaching a Broader Population of Students through ’Unplugged’ Activities”, Communications of the ACM, 58(3):25–27, 2015.

[3] B. Obama. “President obama asks america to learn computer science”, https://www.youtube.com/watch?v=6XvmhE1J9PY, 12 2013.