

Computer Vision CITS4240

School of Computer Science & Software Engineering
The University of Western Australia

The Fourier Transform

Fourier transform theory

So far we have been processing images by looking at the grey level at each point in the image. These methods are known as spatial methods.

However, there are many ways of transforming image data into alternative representations that are more amenable for certain types of analysis. The most common image transform takes spatial data and transforms it into frequency data. This is done using the *Fourier transform*.

The Fourier transform is simply a method of expressing a function (which is a point in some infinite dimensional vector space of functions) in terms of the sum of its projections onto a set of basis functions. Since an image is only defined on a closed and bounded domain (the image window), we can assume that the image is defined as being zero outside this window. In other words, we can assume that the image function is *integrable* over the real line.

To see how the Fourier transform works, we will begin with a one-dimensional signal and consider a simple step function. This is equivalent to taking a horizontal slice through an image that is black on its left half and white on its right half, as shown in figure 1.

Now, a step function (or a square wave form) can be represented as a sum of sine waves of frequency $\omega, 3\omega, 5\omega, \dots$, where ω is the frequency of the square wave, and we recall that frequency = 1/wavelength. Normally, frequency refers to the rate of repetitions per unit time, that is, the number of cycles per second (Hertz). In images we are concerned with spatial frequency, that is, the rate at which brightness in the image varies across the image, or varies with viewing angle. Figure 2 shows the sum of the first few terms in a sine wave decomposition of a square wave. This sum converges to the square wave as the number of terms tends to infinity.

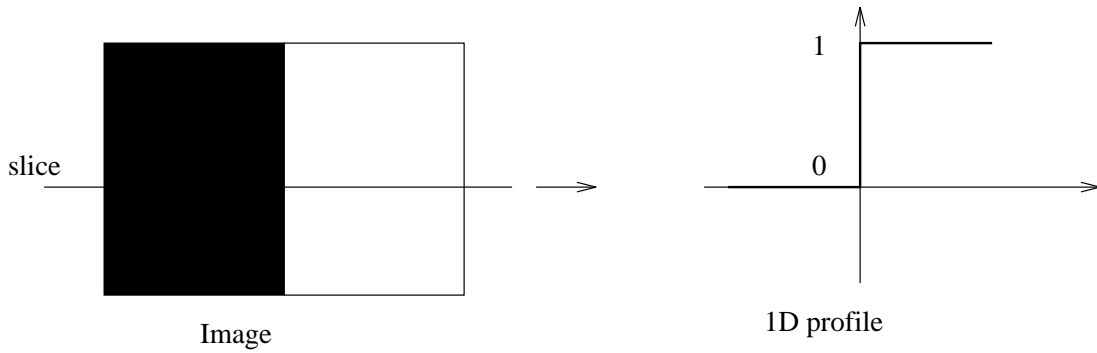


Figure 1: A step function as a slice through an image.

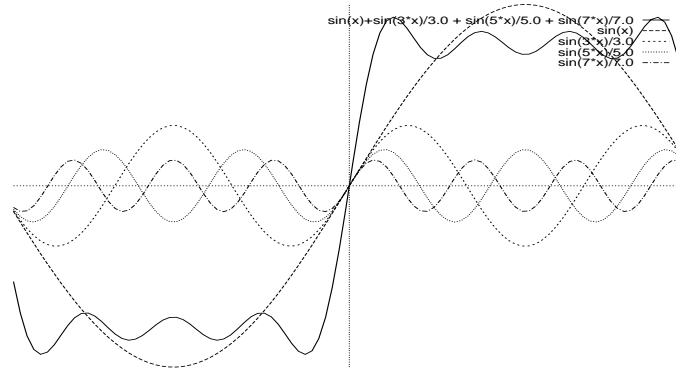


Figure 2: A step function is the sum of an infinite number of sine waves.

From the decomposition of the signal into varying sinusoidal components we can construct a diagram displaying the amplitudes of all the sinusoids for all the frequencies. A graph of such a diagram is given in figure 3 below for the square wave.

Note that we have to consider negative frequencies (whatever that might actually mean) so the sinusoidal component of frequency f and amplitude A_1 has to be split into two components of amplitude $A_1/2$ at the frequencies $+f$ and $-f$. A graph of the amplitude of the Fourier components is known as the *spectrum* of the wave form.

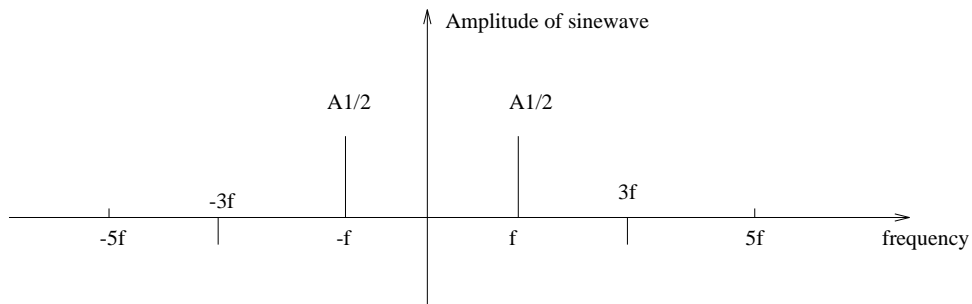


Figure 3: The amplitude of the sine waves at each frequency for a square wave.

So what does the Fourier transform really mean? When we calculate the Fourier transform of an image, we treat the intensity signal across the image as a function, not just an array

of values. The Fourier transform describes a way of decomposing a function into a sum of orthogonal basis functions in just the same way as we decompose a point in Euclidean space into the sum of its basis vector components.

For example, a vector \mathbf{v} in 3-space is described in terms of 3 orthogonal unit vectors \mathbf{i} , \mathbf{j} and \mathbf{k} , and we can write \mathbf{v} as the sum of its projections onto these 3 basis vectors:

$$\mathbf{v} = x\mathbf{i} + y\mathbf{j} + z\mathbf{k}.$$

Given the vector \mathbf{v} , we can calculate the components of \mathbf{v} in each of the \mathbf{i} , \mathbf{j} , and \mathbf{k} directions by calculating the dot product (or *inner* product or *projection*) of \mathbf{v} and each of these basis vectors. Thus

$$x = \mathbf{v} \cdot \mathbf{i}, y = \mathbf{v} \cdot \mathbf{j}, \text{ and } z = \mathbf{v} \cdot \mathbf{k}.$$

A similar process is used to calculate the Fourier transform of a function. The function is just, conceptually, a point in some vector space (although now the vector space is infinitely dimensional). Given our orthogonal basis functions, we calculate the component of our given function in each of the basis functions by calculating the inner product between the two. The standard basis functions used for Fourier transform are $\{\sin(2\pi\omega x), \cos(2\pi\omega x), \omega \in \mathbf{R}\}$ or, equivalently $\{e^{-i2\pi\omega x}, \omega \in \mathbf{R}\}$. It is the frequency ω that varies over the set of all real numbers to give us an infinite collection of basis functions. Since

$$e^{2\pi i\omega x} = \cos(2\pi\omega x) + i \sin(2\pi\omega x),$$

we see that the Fourier transform has real and imaginary components. Moreover, the exponential form of basis function allows us to represent both real and complex valued functions by their Fourier transform.

We can show that any two basis functions of different frequencies are orthogonal by calculating their inner product and showing that it is 0. For example, for the real case and considering only the cosine terms,

$$\int \cos(2\pi\omega_1 x) \cos(2\pi\omega_2 x) dx = 0$$

for $\omega_1 \neq \omega_2$, because the function being integrated is actually a cosine function itself ($2 \cos A \cos B = \cos(A+B) + \cos(A-B)$) and so it has equal areas above and below the x -axis.

Thus, we project our given function f onto our basis functions $e^{-2\pi i\omega x}$ to get the Fourier amplitudes $F(\omega)$ for each frequency ω :

$$\mathcal{F}(f(x)) = F(\omega) = \int f(x)e^{-2\pi i\omega x} dx.$$

In general, $F(\omega)$ will be complex, say of the form $a(\omega) + ib(\omega)$.

We often express F in polar form though:

$$F(\omega) = |F(\omega)| e^{i\Phi(\omega)},$$

where

$$|F(\omega)| = \sqrt{a^2 + b^2}$$

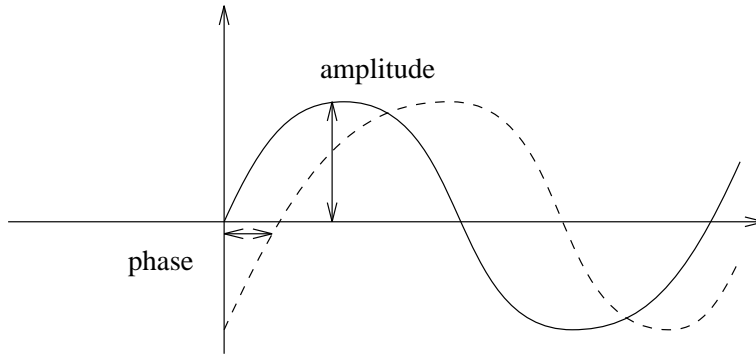


Figure 4: The amplitude and phase angle of a sine wave at a particular frequency.

and

$$\Phi(\omega) = \tan^{-1}\left(\frac{b}{a}\right).$$

The norm of the amplitude, $|F(\omega)|$ is called the *Fourier spectrum* of f , and the exponent $\Phi(\omega)$ is called the *phase angle*. The square of the amplitude is just $P(\omega) = a^2(\omega) + b^2(\omega)$ and is called the *power spectrum* of f .

In many applications only the amplitude information is needed and the phase information is discarded. However, despite this common practice, phase information should not be ignored. In images, as in sound signals, phase carries considerable information [3]. Openheim and Lim have shown that if we construct synthetic images made from the amplitude information of one image and the phase information of another, it is the image corresponding to the phase data that we perceive, if somewhat degraded.



Figure 5: The amplitude data are taken from the vdu image and the phase data are taken from the face image.



Figure 6: The phase data dominates our perception.

Now, having generated the Fourier transform of a function, we want to be able to reconstruct the original function from its Fourier components. This is simply done by summing up all the Fourier components multiplied by their corresponding basis function, that is,

$$f(x) = \int F(\omega)e^{2\pi i\omega x}d\omega.$$

This is analogous to expressing the vector \mathbf{v} as the sum of its projections onto the basis vectors. Note that the inverse Fourier transform uses the basis functions $e^{2\pi i\omega x}$, whilst the Fourier transform uses the basis functions $e^{-2\pi i\omega x}$. This prevents a sign change occurring in the reconstruction process, since $i \times i = -1$, $-i \times i = 1$.

Example Consider again the square wave form shown in the figure below.

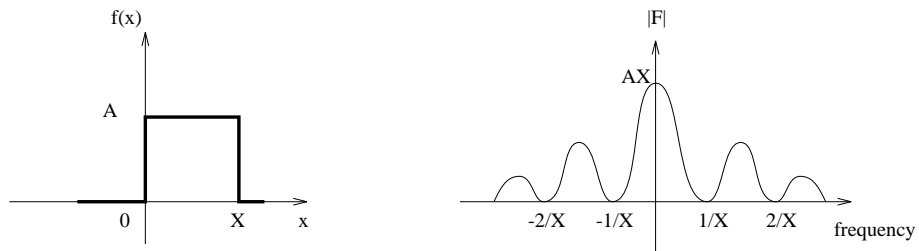


Figure 7: A square wave form and its Fourier spectrum.

Now

$$\begin{aligned} F(\omega) &= \int f(x)e^{-2\pi i\omega x}dx \\ &= \int_0^X Ae^{-2\pi i\omega x}dx \\ &= \frac{-A}{2\pi i\omega}e^{-2\pi i\omega x}\Big|_0^X \\ &= \frac{-A}{2\pi i\omega}[e^{-2\pi i\omega X} - 1] \end{aligned}$$

$$\begin{aligned}
&= \frac{A}{2\pi i\omega} [e^{\pi i\omega X} - e^{-\pi i\omega X}] e^{-\pi i\omega X} \\
&= \frac{A}{\pi\omega} \sin(\pi\omega X) e^{-\pi i\omega X}
\end{aligned}$$

This is a complex-valued quantity, and the Fourier spectrum is given by its modulus, $AX \frac{\sin(\pi\omega X)}{\pi\omega X}$. The last step in this evaluation uses the identity $(e^{i\theta} - e^{-i\theta})/2i = \sin\theta$, which can be seen very easily by looking at the vector geometry defined by the complex exponential terms.

Important properties of the Fourier transform

Suppose we are given two functions f and g , with Fourier transforms F and G , and suppose that a and b are constants. Then

- The Fourier transform is linear, that is,

$$\mathcal{F}(af(x) + bg(x)) = aF(\omega) + bG(\omega).$$

- Changing spatial scale inversely affects frequency and amplitude, that is,

$$\mathcal{F}(f(ax)) = \frac{1}{a} F\left(\frac{\omega}{a}\right).$$

- Shifting the function only changes the phase of the spectrum, that is,

$$\mathcal{F}(f(x - a)) = F(\omega) e^{2\pi i\omega a}.$$

We can also take advantage of symmetries in the spatial and frequency domains as follows:

- if $f(x)$ is real, then $F(-\omega) = F(\omega)^*$
- if $f(x)$ is imaginary, then $F(-\omega) = -F(\omega)^*$
- if $f(x)$ is even, then $F(-\omega) = F(\omega)$
- if $f(x)$ is odd, then $F(-\omega) = -F(\omega)$.

Here, the notation $*$ indicates the complex conjugate operation. Thus, if $f(x)$ is real and even, then $F(\omega)$ is real and even, and if $f(x)$ is real and odd, then $F(\omega)$ is imaginary and odd.

The *Convolution Theorem* tells us that convolution in the spatial domain corresponds to multiplication in the frequency domain, and vice versa. That is,

$$f(x) \otimes g(x) \Leftrightarrow F(\omega)G(\omega)$$

and equivalently

$$f(x)g(x) \Leftrightarrow F(\omega) \otimes G(\omega).$$

Thus, convolution with large masks in the spatial domain can often be done more efficiently as multiplication in the frequency domain. Likewise, division in the frequency domain corresponds to deconvolution in the spatial domain. This is the basis by which image restoration for blur due to focus or motion is done.

Two dimensional Fourier transforms

Now an image is thought of as a two dimensional function and so the Fourier transform of an image is a two dimensional object. Thus, if f is an image, then

$$F(\omega, \nu) = \int \int f(x, y) e^{-2\pi i(\omega x + \nu y)} dx dy.$$

Fortunately, it is possible to calculate this integral in two stages, since the 2D Fourier transform is *separable*. Thus, we first form the Fourier transform with respect to x :

$$F(\omega, y) = \int f(x, y) e^{-2\pi i \omega x} dx$$

and then we calculate the Fourier transform of this function of y :

$$F(\omega, \nu) = \int F(\omega, y) e^{-2\pi i \nu y} dy.$$

The discrete Fourier transform

So far, we have been considering functions defined on the continuous line. In digital images we can only process a function defined on a discrete set of points. This leads us to the *discrete Fourier transform* (DFT), whose equations are very similar to those for the continuous Fourier transform.

The DFT provides information over a discrete number of frequencies, so we need to determine precisely which frequencies these are. To do this, we sample a continuous function f at intervals of Δ , generating a sequence of sampled values

$$f_k = f(k\Delta),$$

where $k = \dots, -2, -1, 0, 1, 2, \dots$. The reciprocal of Δ is the sampling rate, or frequency. For any sampling interval Δ , there is a special frequency ω_c called the Nyquist critical frequency:

$$\omega_c = \frac{1}{2\Delta}.$$

The Nyquist frequency represents the highest frequency that can be represented by something sampled at intervals of Δ , that is, a frequency having wavelength of 2Δ . Alternatively, if one is seeking to describe a function by a set of discrete values, we must sample the function at 2 times the highest frequency in the function.

For example, critical sampling of a sine wave is at 2 sample points per cycle; sampling at any slower rate will “miss” the sine wave.

If we wish to take the Fourier transform of N sampled values of a function $\{f_k, k = 0, 1, \dots\}$ with N items of input, then we will not be able to produce more than N independent items of output.

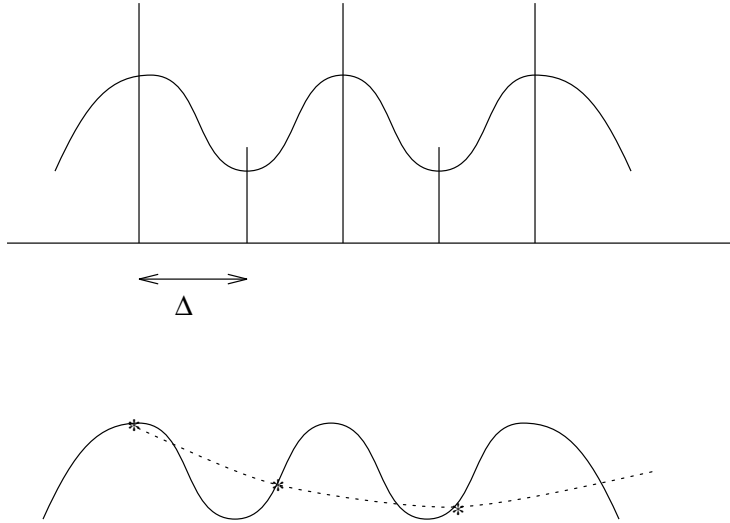


Figure 8: Critical sampling of a sine wave. Sampling at a slower rate will miss the structure of the particular sine wave.

We generate the Fourier transform at discrete frequencies

$$\omega_n = \frac{n}{N\Delta},$$

where $n = \frac{-N}{2}, \frac{-N}{2} + 1, \dots, \frac{N}{2}$. These limits correspond to the upper and lower Nyquist frequencies.

We now approximate the integral equation for the Fourier transform by a discrete summation:

$$\begin{aligned} F(\omega_n) &= \int f(x)e^{-2\pi i\omega_n x} dx \\ &\approx \sum_{k=0}^{N-1} f_k e^{-2\pi i\omega_n k\Delta} \\ &= \Delta \sum_{k=0}^{N-1} f_k e^{-2\pi i k \frac{n}{N}}. \end{aligned}$$

Typically, the sampling distance Δ is set to 1, so the discrete Fourier transform is given by

$$F_n = \sum_{k=0}^{N-1} f_k e^{-2\pi i k n / N}.$$

The *inverse* discrete Fourier transform is given by

$$f_k = \frac{1}{N} \sum_{n=0}^{N-1} F_n e^{2\pi i k n / N}.$$

Note that the only differences between the forward and inverse transforms are (i) changing the sign in the exponential, and (ii) dividing the answer by N .

So far, we are considering n to vary from $-N/2$ to $N/2$ as k varies from 0 to $N - 1$. However, our equation for the DFT is periodic in n with period N , that is,

$$e^{-2\pi i k \frac{1}{N}} = e^{-2\pi i k \frac{N+1}{N}}.$$

Thus, if we let n vary from 0 to $N - 1$ (which is a complete period) we will cover all the frequencies and n will vary in exact correspondence to k . Thus

- $n = 0$ corresponds to the zero frequency component,
- $1 \leq n \leq \frac{N}{2} - 1$ corresponds to the frequencies $0 < \omega < \omega_c$,
- $\frac{N}{2} + 1 \leq n \leq N - 1$ corresponds to the frequencies $-\omega_c < \omega < 0$, and
- $n = N/2$ corresponds to both $-\omega_c$ and ω_c .

All standard implementations of the FFT produce data in this form.

The DFT needs N complex multiplications of f_k by $e^{-2\pi i k n / N}$ and $N - 1$ additions of the resulting values to transform N values; its complexity is thus proportional to N^2 , thus justifying its nickname of the Slow Fourier Transform.

The Fast Fourier Transform (FFT) is an ingenious algorithm which exploits various properties of the Fourier transform to enable the transformation to be done in $O(N \log_2 N)$ operations. However, the FFT requires the size of the input data to be a power of 2; if this is not the case, the data are either truncated or padded out with zeros.

References

- [1] R. Bracewell. *The Fourier Transform and its Applications*. McGraw Hill, 1965.
- [2] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Addison-Wesley Publishing Company, 1992, chapter 3.
- [3] A. V. Oppenheim and J. S. Lim. The importance of phase in signals. *Proceedings of the IEEE* 69:529–541, 1981.
- [4] William H. Press, Brian P. Flannery, Saul A. Teukolsky, and William T. Vetterling. *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press, 1986, chapter 12.