Ubung 2. Fast Fourier Transform in image processing

April 27, 2005

1 Background

Fourier Transform was a revolutionary concept to which it took mathematicians all over the world over a century to "adjust". Basically, the contribution of Fourier Transformation states that any function can be expressed as the integral of sines and/or cosines multiplied by a weighting function. It does not matter how complicated the function is; as long as it meets some mild mathematical conditions, it can be represented in such way. The function, expressed in a Fourier transform, can be reconstructed (recovered) completely via an inverse process. This important properties of Fourier transform allow us to work in the "frequency domain" and then return to the original domain without losing any information.

2 Fourier Transform and its Inverse

The Fourier transform, $F(u)$, of a single variable, continuous function, $f(x)$, is defined by the equation

$$F(u) = \int_{\infty}^{\infty} f(x)e^{-j2\pi ux}\, dx \tag{1}$$

where $j = \sqrt{-1}$. Conversely, given $F(u)$, we can obtain $f(x)$ by means of the *inverse* Fourier transform

$$f(x) = \int_{\infty}^{\infty} F(u)e^{j2\pi ux}\, du \tag{2}$$

These two equations comprise the *Fourier transform pair*, which indicates the fact mentioned before that the original function can be recovered without loss of information.

These equations can be easily extended to two variables, u and v:

$$F(u,v) = \int_{\infty}^{\infty} \int_{\infty}^{\infty} f(x,y)e^{-j2\pi(ux+vy)}\, dxdy$$

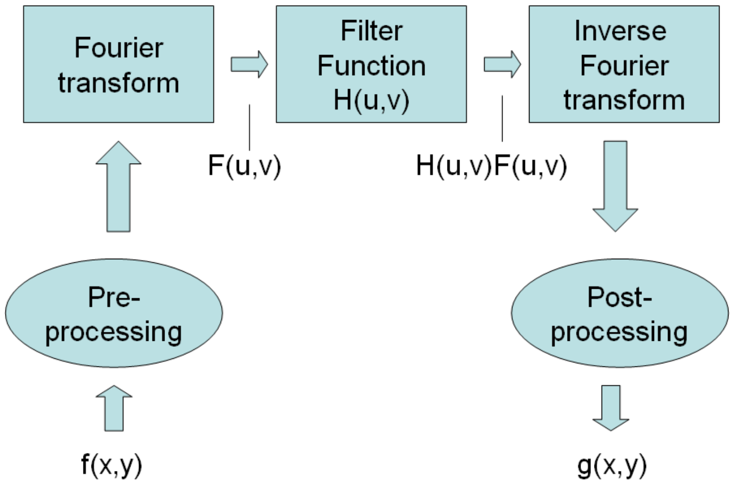

Figure 1: Basic steps for filtering in the frequency domain.

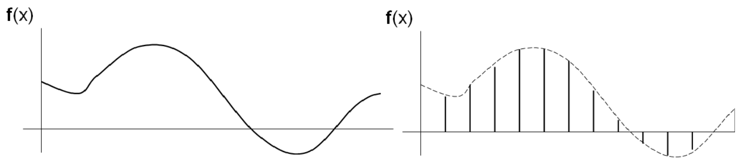

Figure 2: Left: a continuous function $f(x)$. Right: the discrete function $f(x)$.

Similarly, the inverse transform,

$$f(x,y) = \int_{\infty}^{\infty}\int_{\infty}^{\infty} F(u,v)e^{j2\pi(ux+vy)}\, dudv$$

3 Discrete Fourier transform (DFT)

Since the digital images are model by discrete functions, we are more interested on the discrete Fourier transform. The one dimension of discrete fourier transform is given by the equation

$$F(u) = \frac{1}{M}\sum_{x=0}^{M-1} f(x)e^{-j2\pi ux/M} \qquad \text{for} \qquad u = 0,1,2,...,M-1 \tag{3}$$

Note that $f(x)$ in (3) is a discrete function of one variable, while the $f(x)$'s in (1) (2) are continuous functions. See the Figure.2. Similarly, given $F(u)$, we can obtain the original discrete function $f(x)$ by inverse DFT:

$$f(x) = \sum_{u=0}^{M-1} F(u) e^{j2\pi ux/M} \qquad \text{for} \qquad x = 0, 1, 2, ..., M-1 \tag{4}$$

The discrete Fourier transform (3) and its inverse (4) is the foundation for the most frequency based image processing.

The concept of the frequency domain, mentioned numerous times, follows directly from Euler's formula:

$$e^{j\theta} = \cos\theta + j\sin\theta \tag{5}$$

Substituting this expression into Eq. (3), and using the fact that $\cos(-\theta) = \cos(\theta)$, we get

$$F(u) = \frac{1}{M} \sum_{x=0}^{M-1} f(x)[\cos(2\pi ux/M) - j\sin(2\pi ux/M)] \tag{6}$$

for $u = 0, 1, 2, ..., M-1$. Thus, **each** term of the Fourier transform $F(u)$ is composed of the sum of **all** values of the function $f(x)$. The values of $f(x)$, in turn, are multiplied by sines and cosines of various frequencies. The $F(u)$'s are appropriately named *frequency components*, which represent the distribution of $f(x)$ with all x to the frequencies $2\pi ux/M$. Although the x's also affect the frequencies, they are summed out and they all make the same contribution for each value of u. Correspondingly, the values of $f(x)$ is named *time components* (or space components). The DFT can be viewed as "re–distributing" the M time components $f(x)$'s onto M frequency components. Conversely, the inverse DFT is the process that "re–distributes" the M frequency components $F(u)$'s onto M time components.

A useful analogy is to compare the Fourier transform to a glass prism. The prism is a physical devise to split the white light into various color components, each depending on its frequency. The Fourier transform may be viewed as a "mathematical prism" that separates a function $f(x)$ into various components, also based on frequency content.

Extension of the one–dimensional DFT and its inverse to two dimensions is straightforward. The discrete Fourier transform of a image function $f(x, y)$ of size $M \times N$ is given by the equation

$$F(u, v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi(ux/M + vy/N)} \tag{7}$$

As in the 1-D case, this expression must be computed for values of $u = 0, 1, 2, ..., M-1$ and also for $v = 0, 1, 2, ...N-1$. Similarly, given $F(u, v)$, we obtain $f(x, y)$ via the 2-D inverse Fourier transform, given by

$$f(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) e^{j2\pi(ux/M + vy/N)} \tag{8}$$

for $x = 0, 1, 2, ..., M-1$ and $y = 0, 1, 2, ..., N-1$.

4 Fast Fourier Transform

One of the main reasons that the DFT become an essential tool in signal processing was the development of the fast Fourier transform (FFT). Computing the 1-D discrete Fourier transform of M points using Eq.(3) directly requires on the order of M^2 multiplication/addition operations. The FFT accomplishes the same task on the order of $M \log_2 M$ operations. For example, if $M = 1024$, the direct method will require approximately 10^6 operations, while the FFT will require approximately 10^4 operations. This is a computational advantage of 100 to 1. Furthermore, the bigger the problem, the greater the computational advantage. Here we focus on 1-D FFT. The 2-D fast Fourier can be obtained by successive passes of a 1-D transform algorithm.

The FFT algorithm developed in this section is based on the so–called *successive doubling method*. Now we express Eq.(3) in the form

$$F(u) = \frac{1}{M} \sum_{x=0}^{M-1} f(x) W_M^{ux} \tag{9}$$

where

$$W_M = e^{-j2\pi/M} \qquad \text{so} \qquad W_M^{ux} = e^{-j2\pi ux/M} \tag{10}$$

and the number of points M is assumed to be the power of 2, like

$$M = 2^n \tag{11}$$

with n being a positive integer. Hence, M can be expressed as

$$M = 2K \tag{12}$$

with K also being a positive integer. Substituting Eq.(12) into Eq.(9) yields

$$\begin{aligned} F(u) &= \frac{1}{2K} \sum_{x=0}^{2K-1} f(x) W_{2K}^{ux} && (13) \\ &= \frac{1}{2}\Big[\frac{1}{K} \sum_{x=0}^{K-1} f(2x) W_{2K}^{u(2x)} + \frac{1}{K} \sum_{x=0}^{K-1} f(2x+1) W_{2K}^{u(2x+1)}\Big] && (14) \end{aligned}$$

However, it can be shown using Eq.(11) that $W_{2K}^{2uk} = W_K^{ux}$, so Eq.(14) can be expressed as

$$F(u) = \frac{1}{2}\Big[\frac{1}{K} \sum_{x=0}^{K-1} f(2x) W_K^{ux} + \frac{1}{K} \sum_{x=0}^{K-1} f(2x+1) W_K^{ux} W_{2K}^{u}\Big] \tag{15}$$

Defining

$$\begin{aligned} F_{even}(u) &= \frac{1}{K} \sum_{x=0}^{K-1} f(2x) W_K^{ux} && (16) \\ F_{odd}(u) &= \frac{1}{K} \sum_{x=0}^{K-1} f(2x+1) W_K^{ux} && (17) \end{aligned}$$

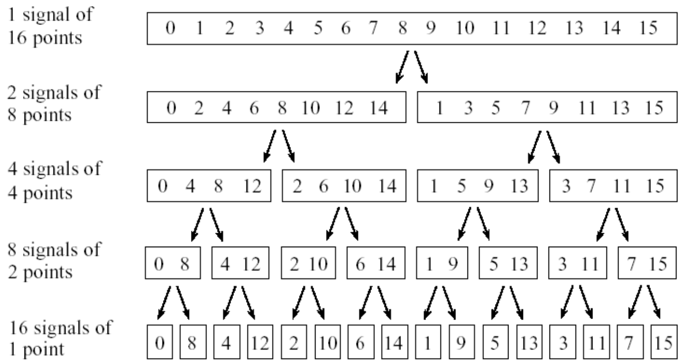

Figure 3: The FFT decomposition. An M point signal is decomposed into M signals each containing a single point. Each stage uses an interlace decomposition, separating the even and odd numbered samples.

for $u = 0, 1, 2, ..., K - 1$. The Eq.(15) can be reduced to

$$F(u) \quad = \quad \frac{1}{2}\Big[F_{even}(u) + F_{odd}(u)W_{2K}^{u}\Big] \tag{18}$$

Also, because $W_M^{u+M} = W_M^u$ and $W_{2M}^{u+M} = -W_{2M}^u$

$$F(u+K) \quad = \quad \frac{1}{2}\Big[F_{even}(u) - F_{odd}(u)W_{2K}^{u}\Big] \tag{19}$$

Careful analysis of Eqs. (16) through (19) reveal some interesting properties of these expressions. A M-point transform can be computed by dividing the original expression into two parts, as indicated in Eq. (18) and (19). Computing the first half of $F(u)$ requires evaluation of the two $(M/2)$-point transforms given in Eqs.(16) and (17). The resulting values of $F_{even}(u)$ and $F_{odd}(u)$ are then substituted into Eq.(18) to obtain $F(u)$ for $u = 0, 1, 2, ..., (M/2 - 1)$. The other half then follows directly from Eq.(19) without additional transform evaluations.

The FFT operates by decomposing an M point time domain signal into M time domain signal each composed of a single point. The second step is to calculate the M frequency spectra corresponding to these M time domain signals. Lastly, the M spectra are synthesized into single frequency spectrum.

- Fig.3 shows an example of the time domain decomposition. In this example, a 16 point signal is decomposed through four separate stages. The decomposition continues until there are M signals composed of a single point. There are $\log_2 M$ stages required in this decomposition.

- The next step in the FFT alogrithm is to find the frequency spectra of 1 point in the time domain. The frequency spectrum of 1 point signal is equal to *itself.* This means that nothing is required to do this step.

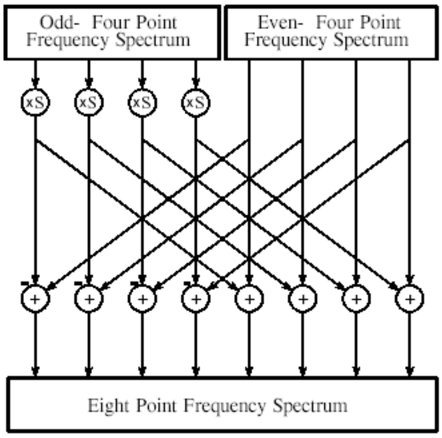

Figure 4: FFT synthesis flow diagram. This shows the method of combining two 4 point frequency spectra into a single 8 point frequency spectrum. The $\times S$ operation means that the signal is multiplied by a sinusoid with an appropriately select frequency

- The last step is to combine the M frequency spectra in the reverse order that the time domain decomposition took place. In the first stage, 16 frequency spectra (1 point each) are synthesized into 8 frequency spectra (2 point each). In the second stage, the 8 frequency spectra (2 points each) are synthesized into 4 frequency spectra (4 points each),and so on. Until a 16 point frequency spectrum is achieved. See the Fig.4.

References

[1] Gonzalez and Woods *Digital Image Processing 2nd Edition*, Prentice Hall 2002.

[2] Steven W. Smith *The Scientist and Engineer's Guide to Digital Signal Processing*, California Technical Publishing