Topic 8: Images in the Frequency Domain

- Fourier Series/Transform
- Images in Frequency Domain
- The Convolution Theorem
- High-Pass, Low-Pass and Band-Pass Filters
A Different type of Basis

• Last week we learned how to represent images using a different basis
• This week we are going to learn how to represent images using a very different and perhaps counter-intuitive basis – the *Fourier basis*
• This basis gives us a representation of our images in *frequency space*
Representing a Function with Sine Functions

\[ f(x) = \begin{cases} 
-1 & \text{if } x < 0 \\
1 & \text{otherwise}
\end{cases} \]

• Let’s try to approximate the above function with only a sine wave
• We are going to use the basic “building block” of a general sine wave:

\[ A \sin(\omega x + \phi) \]
Quick Review: Sinusoid

\[ y(x) = A \sin(\omega x + \phi) \]

\( A \) is the wave’s amplitude
Quick Review: Sinusoid

\[ y(x) = A \sin(\omega x + \phi) \]

\( \omega \) is the **angular** frequency

\[ \omega = 2\pi f \]

where \( f \) is the frequency of the wave
Quick Review: Sinusoid

\[ y(x) = A \sin(\omega x + \phi) \]

\( \phi \) is the phase
Representing a Function with Sine Functions

• First attempt – represent using single sine wave:

\[ f(x) \approx \sin(x) \]
Representing a Function with Sine Functions

• First attempt – represent using single sine wave:

\[ f(x) \approx \frac{4}{\pi} \sin(x) \]
Representing a Function with Sine Functions

• Second attempt – represent using two sine waves of different frequency:

\[ f(x) \approx \frac{4}{\pi} \sin(x) + \frac{4}{3\pi} \sin(3x) \]
Fourier Series

• It turns out we can represent our step function exactly as the infinite sum of the sine waves:

\[ F(x) = \sum_{n=1}^{N=\infty} \frac{4}{(2n - 1)\pi} \sin((2n - 1)x) \]

(Note: Don’t memorize this! We will learn how to calculate these coefficients....)
Fourier Series

- **Discontinuities are difficult** – they require more higher frequency terms to represent
- The representation error for finite terms gives a “ringing” effect – name will make more sense in 2D, but the cause of is more intuitive in 1D
Fourier Transform

• In general we can represent a function \( f(x) \) by the sum of an infinite series of sine waves.

• The way we do this is called the **Fourier transform**, and is usually defined with the **complex** exponential:

\[
F(\omega) = \int_{-\infty}^{\infty} f(x) e^{-i\omega x} \, dx
\]

• \( f(x) \) is the function we want to transform to the frequency domain.

• \( F(\omega) \) is the function in the frequency domain, where \( \omega = 2\pi f \).
Fourier Transform

- **Fourier transform:**

\[
F(\omega) = \int_{-\infty}^{\infty} f(x) e^{-i\omega x} \, dx
\]

- \(f(x)\) is the function we want to transform to the frequency domain
- \(F(\omega)\) is the function in the frequency domain, where \(\omega = 2\pi f\)

- Often this is defined instead with spatial frequency \(f\):

\[
F(f) = \int_{-\infty}^{\infty} f(x) e^{-i2\pi fx} \, dx
\]
Inverse Fourier Transform

• We can also go back to the original spatial signal with the inverse transform!

• *Inverse Fourier transform*:

\[ f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) e^{i\omega x} \, d\omega \]

• \( F(\omega) \) is the function we want to transform to the spatial domain
• \( f(x) \) is the function in the spatial domain
Fourier Transform

• **Fourier transform:**

\[ F(\omega) = \int_{-\infty}^{\infty} f(x) e^{-i\omega x} \, dx \]

• Wait, what happened to the sine?

• Still there, just hidden behind the complex exponential function:

\[ e^{i\omega x} = \cos \omega x + i \sin \omega x \]

(real) \hspace{1cm} (complex)
Fourier Transform

• Behind the complex exponential:

\[ e^{i\omega x} = \cos \omega x + i \sin \omega x \]

(Euler’s formula)

(Note: \( e^{-i\omega x} = \cos -\omega x + i \sin -\omega x \))

• This cos/sin pair can encode the phase of the sinusoid (i.e. direction of vector on unit circle):

\[ u \cos \omega x + v \sin \omega x = A \sin(\omega x + \phi) \]

where,

\[ A = \pm \sqrt{u^2 + v^2}, \quad \phi = \arctan \frac{u}{v} \]

Note: It is not important to understand the complex exponential function, it is just a more compact way of encoding the frequency/phase
Fourier Transform as a Change in Basis

• **Fourier transform:**
  \[ F(\omega) = \int_{-\infty}^{\infty} f(x) i \sin \omega x + f(x) \cos \omega x \, dx \]

• The sine and cosine functions are an **orthogonal basis**
• The Fourier transform decomposes the function \( f(x) \) into a **weighted sum of basis functions** (i.e. sin/cos) in the complex space
• This is similar to the change of basis we saw before, where we defined a vector based on two basis vectors, e.g. \( \mathbf{v} = v_0 \mathbf{i} + v_1 \mathbf{j} \)
Frequency Spectrum

- A sine curve is transformed to a **single point** in the Frequency domain.
- This is because it is a single frequency (i.e. **one term** in the Fourier series).
• More complex functions are transformed into **many points** in the frequency domain
• They are composed of many frequencies (i.e. **many terms** in the Fourier series)
Fourier Series is Just Another Basis!

https://en.wikipedia.org/wiki/Fourier_series
Topic 8: Images in the Frequency Domain

• Fourier Series/Transform
• Images in Frequency Domain
• The Convolution Theorem
• High-Pass, Low-Pass and Band-Pass Filters
Fourier Transform of Images

• The continuous 2D Fourier transform is defined:

\[ F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-i(ux+vy)} \, dx \, dy \]

• The discrete 2D Fourier transform is defined:

\[ F(u, v) = \sum_{x} \sum_{y} f(x, y) e^{-i(ux+vy)} \]

• Images are just a discrete 2D function, so we can also represent them in the frequency domain
Simple Fourier Examples

Fourier is parameterized by $u, v$: frequency components in the x and y directions.
Simple Fourier Examples

Fourier is parameterized by $u, v$ \( \text{frequency components in the x and y directions} \)
Simple Fourier Examples

$f(x, y)$

$F(u, v)$

Fourier is parameterized by $u, v$: frequency components in the x and y directions
Fourier Transform of Images

• As we said of the 1D Fourier transform, it can be thought of as a change of basis, where the basis functions are sinusoids of different frequencies

• Each of the images we just saw is actually a Fourier **basis function**
Fourier Transform of Images

\[ F(u, v) = \sum_{x} \sum_{y} f(x, y) e^{-i(u x+v y)} \]

- With the 2D Fourier transform we can visualize these basis functions as images!

- Above right, we show the 2D basis function, below right, the coordinates of that function in the 2D Frequency domain
Fourier Transform of Images

• Vector form may be more intuitive:
  \[ F(\omega) = f(x, y) e^{-i\omega x} \]
  where \( \omega = (u, v), \quad x = (x, y) \)

• Direction of the basis function (sinusoid) is direction of the vector \( \omega=(u, v) \)

• Frequency is determined by the magnitude of the vector \( \omega=(u, v) \)
Fourier Transform of Images

• Above we show the 2D basis function, below the coordinates of that function in the 2D Frequency domain

• Direction of the basis function (sinusoid) is direction of the vector $\omega = (u, v)$

• Frequency is determined by the magnitude of the vector $\omega = (u, v)$
Fourier Transform of Images

• Above we show the 2D basis function, below the coordinates of that function in the 2D Frequency domain

• Direction of the basis function (sinusoid) is direction of the vector $\omega = (u, v)$

• Frequency is determined by the magnitude of the vector $\omega = (u, v)$
Images and the Fourier Transform

• We have a set of basis 2D sinusoids (let’s say images)
• Images can be decomposed into a **weighted** linear combination of sinusoids of different frequencies
• It is these **weights** that are the values in the Fourier “image” – and they are complex numbers

\[
= w_0 \cdot \begin{array}{cccc}
& & & \\
& +w_1 \cdot & & \\
& & \cdots & \\
& & & +w_2 \cdot \\
& & & \\
\end{array} 
\]

Increasing frequency \( \rightarrow \)
Fourier Transform as a Basis

\[
\exp[j 2\pi (ux + vy)] = \\
\cos[2\pi (ux + vy)] + j \sin[2\pi (ux + vy)]
\]

Real (cos) part

- \((u, v) = (1, 0)\)
- \((u, v) = (1, 1)\)
- \((u, v) = (0, 5)\)

Imaginary (sin) part
Intermission
Fourier Transformed Image

This is a complex valued function! Can’t just display the values as image

\[ f(x, y) \rightarrow F(u, v) \]

Fourier Transform

Error
Fourier Transformed Image

Doesn’t seem much better!

\[ f(x, y) \xrightarrow{\text{Fourier Transform}} F(u, v) \]

\[ |F(u, v)| \]
Fourier Transformed Image

$f(x, y) \rightarrow F(u, v) \rightarrow |F(u, v)|$

Bright spot at $F(0,0)$
Fourier Transformed Image

\[ f(x, y) \rightarrow F(u, v) \rightarrow \log |F(u, v)| \]
DC Component

• F(0,0) is called the DC component
• What is this bright F(0,0) component?
  \[ F(0,0) = \sum_x \sum_y f(x, y) e^{i(0x+0y)} \]

• In the Fourier domain, it’s equal to the sum of all image pixels
• In the spatial domain, it’s the image’s mean brightness/intensity
• This is the information in the image that does not change with spatial location
Frequency/Phase in Fourier

FT(f(x, y)) = F(u, v)

amplitude of sinusoid of frequency \( \omega = |(u, v)| \)

phase of sinusoid of frequency \( \omega = |(u, v)| \)
Frequency/Phase in Fourier

• The Fourier transform of an image gives us an “image” $F(u, v)$ where each pixel is a **complex number** representing the components in the Fourier basis.

• If we express these complex numbers instead in **polar coordinates**:
  - magnitude/radius = frequency component
  - angle = phase component

• Remember we are decomposing our function into sinusoids which have both **frequency** and **phase**, i.e.

$$y(x) = A \sin 2\pi f x + \phi$$
amplitude of sinusoid of frequency $\omega = |(u, v)|$

phase of sinusoid of frequency $\omega = |(u, v)|$

$$y(x) = A \sin(\omega x + \phi)$$
Images and the Fourier Transform

• In image processing we will be focusing on the frequency
• However, without the phase component we can’t reconstruct a spatial image!

\[ f(x, y) \]

\[ \log |F(u, v)| \]
Topic 8: Images in the Frequency Domain

• Fourier Series/Transform
• Images in Frequency Domain
• The Convolution Theorem
Images and the Fourier Transform

- Let’s look at what some of these frequency components look like in the spatial domain

\[ f(x, y) \quad \text{and} \quad \log |F(u, v)| \]
Images and the Fourier Transform

• Let’s zero out the low frequency Fourier components

• We are left with **high frequency components** – i.e. edges!

\[ f(x, y) \quad \text{log} |F(u, v)| \]
Images and the Fourier Transform

- Let’s zero out the high frequency Fourier components
- We are left with **low frequency components** – the image looks **blurred**

\[ f(x, y) \]

\[ \log |F(u, v)| \]
Images and the Fourier Transform

• Let’s zero out the high frequency Fourier components

• We are left with low frequency components – looks blurred

\[ f(x, y) \]
Flashback to 1D

- The representation error for finite terms gives a “ringing” effect – name will make more sense in 2D, but the cause of is more intuitive in 1D
Gaussians and Fourier Transform

• This ringing is because a box filter in the spatial domain is mapped to a sinc function in the frequency domain (and vice versa):

\[
sinc(x) = \frac{\sin(x)}{x}
\]

• However, **Gaussians in spatial domain are also Gaussians in frequency domain**
Gaussian Filters in Frequency Domain

- Instead of using a box, let’s try a Gaussian instead
Gaussian Filters in Frequency Domain

• This is a smoothed image!

• Same result as if we convolved image with Gaussian filter

• But all we did here was multiply our Fourier transformed image by a Gaussian...

• How did we get the same result as convolution (many multiplications per pixel) with only one per pixel?
Gaussian Filters in Frequency Domain

• Taking it even further...
Topic 8: Images in the Frequency Domain

• Fourier Series/Transform
• Images in Frequency Domain
• The Convolution Theorem
• High-Pass, Low-Pass and Band-Pass Filters
The Convolution Theorem

• Let $f(x, y), h(x, y)$ be 2D spatial functions, $F(u, v), H(u, v)$ be their corresponding Fourier transforms, $\mathcal{F}^{-1}$ be the inverse Fourier transform, and $*$ is the convolution operator, then:

$$f(x) * h(x) = \mathcal{F}^{-1}(F(u, v)H(u, v))$$

• **Convolution** in the **spatial domain** is the same as **multiplication** in the **frequency domain**

• Why does this matter?

• Because image filtering operations in the spatial domain can be implemented by a simple convolution in the frequency domain!
Image Filtering in Frequency Domain

- Fourier Transform
- Multiply In Fourier
- Inverse Fourier Transform

Filtered image: $f'(x, y)$
When is Fourier Used for Image Filtering?

• So why don’t we always do image filtering in the frequency domain?
• Because the Fourier transform/inverse Fourier transform steps give us significant overhead, it may not be more efficient than spatial convolution, depending on the filter size
• Usually image filtering is only done in frequency domain for large image filters

• It turns out there is a much more efficient implementation of the Discrete Fourier Transform (DFT) called the Fast Fourier Transform (FFT)
• For a 1D signal with N data points, DFT is $O(N^2)$, FFT is $O(N \log N)$
Topic 8: Images in the Frequency Domain

• Fourier Series/Transform
• Images in Frequency Domain
• The Convolution Theorem
• High-Pass, Low-Pass and Band-Pass Filters
Signal Processing Loan-Words

• There are a number of terms we use in visual computing that are borrowed from signal processing, and the frequency domain
• These are important to understand, and you will hear them used quite a bit
• You’ve already seen one: the *DC component*! This actually stands for direct current, which doesn’t make much sense with images
Low-Pass Filters

- These are examples of “**low pass**” and “**high pass**” filters – a term common in signal processing
- **Low-pass filter**, we let low-frequencies “pass” through and “block” high frequencies

\[ f(x, y) \]

\[ |H(u, v)| \]
High-Pass Filters

- These are examples of “low pass” and “high pass” filters – a term common in signal processing
- **High-pass filter**, we let high-frequencies “pass” through and “block” low frequencies

\[ f(x, y) \quad F(u, v) \ast H(u, v) \quad |H(u, v)| \]
Band-Pass Filters

• A band-pass filter only allows through a range of frequencies, i.e. a frequency “band” – here effected by the difference of two gaussian filters

\[ f(x, y) \]

\[ |H(u, v)| \]
End of Topic 8