# Fourier Matching

CS 510

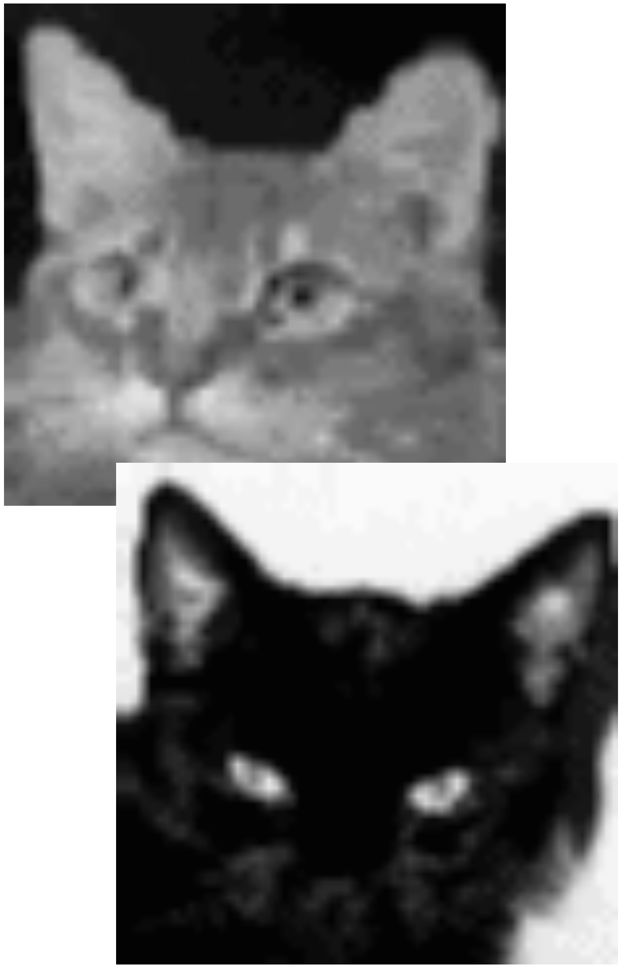
Lecture #7

February 8th, 2013

Colorado State University

*Details are on the assignments page… you have until Monday*

# Programming Assignment #1



Source (Target) Images

Templates

Left Eye

Right Eye

Left Ear

Nose

Colorado State University

# The Discrete Fourier Transform

- For 0 ≤ x < N and 0 ≤ u <N/2:

$$F(u) = \sum_{x=0}^{N-1} f(x)\left[\cos\left(\frac{2\pi ux}{N}\right) - i\sin\left(\frac{2\pi ux}{N}\right)\right]$$

And the discrete inverse transform is:

$$f(x) = \frac{1}{N}\sum_{x=0}^{N-1} F(u)\left[\cos\left(\frac{2\pi ux}{N}\right) + i\sin\left(\frac{2\pi ux}{N}\right)\right]$$

# 2D Fourier Transform

- So far, we have looked only at 1D signals
- For 2D signals, the generalization is:

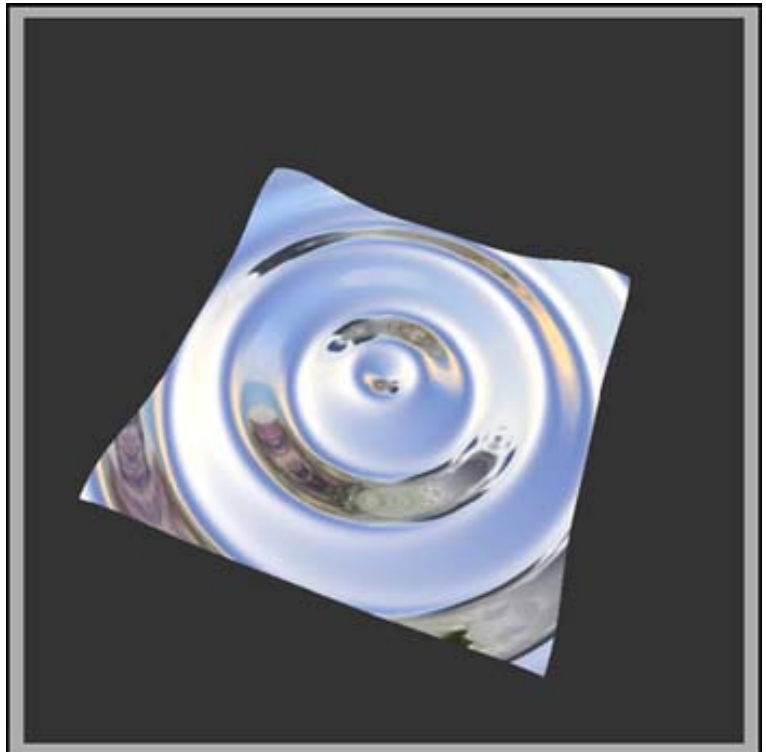
$$F(u,v) \equiv \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\left[\cos\left(2\pi(ux+vy)\right) - i\sin\left(2\pi(ux+vy)\right)\right]$$

- Note that frequencies are now two-dimensional
  - u= freq in x, v = freq in y
- Every frequency (u,v) has a real and an imaginary component.

# 2D sine waves

- This looks like you'd expect in 2D

➢ Note that the frequencies don't have to be equal in the two dimensions.



http://images.google.com/imgres?imgurl=http://developer.nvidia.com/dev_
sine_wave_perturbation_ogl.jpg&imgrefurl=http://developer.nvidia.com/object/
cg_effects_explained.html&usg=__0FimoxuhWMm59cbwhch0TLwGpQM=&h=350&w=350&sz=13&hl=en&start=8&sig2=dBEtH0hp5I1BExgkXAe_kg&tbnid=fcyrIaatfp0P3M:&tbnh=120&tbnw=120&ei=llCYSbLNL4miMoOwoP8L&prev=/images%3Fq%3D2D%2Bsine%2Bwave%26gbv%3D2%26hl%3Den%26sa%3DG

# 2D Discrete Fourier Transform

$$F(u,v) = \sum_{x=-N/2}^{N/2} \sum_{y=-N/2}^{N/2} f(x,y)\left[\cos\left(\frac{2\pi}{N}(ux+vy)\right) - i\sin\left(\frac{2\pi}{N}(ux+vy)\right)\right]$$

- What happened to the bounds on x & y?
- How big is the discrete 2D frequency space representation?

# 1D Fourier bounds

- We define the 1D frequencies as 0 to N/2
- Others use –N/2 to N/2, but remember that
  - cos(x) = cos(-x)
  - sin(x) = -sin(-x)

$$F(u)=\sum_{x=0}^{N-1} f(x)\left[\cos\left(\frac{2\pi ux}{N}\right)-i\sin\left(\frac{2\pi ux}{N}\right)\right]=a+ib$$

$$F(-u)=\sum_{x=0}^{N-1} f(x)\left[\cos\left(\frac{-2\pi ux}{N}\right)-i\sin\left(\frac{-2\pi ux}{N}\right)\right]=a-ib$$

# 2D Frequency Bounds

- What happens in 2D?

$$F(u,v) = \sum_{x=-N/2}^{N/2} \sum_{y=-N/2}^{N/2} f(x,y)\left[\cos\left(\frac{2\pi}{N}(ux+vy)\right) - i\sin\left(\frac{2\pi}{N}(ux+vy)\right)\right] = a+bi$$

$$F(-u,-v) = \sum_{x=-N/2}^{N/2} \sum_{y=-N/2}^{N/2} f(x,y)\left[\cos\left(\frac{2\pi}{N}(-ux+-vy)\right) - i\sin\left(\frac{2\pi}{N}(-ux+-vy)\right)\right] = a-bi$$

$$F(-u,v) = \sum_{x=-N/2}^{N/2} \sum_{y=-N/2}^{N/2} f(x,y)\left[\cos\left(\frac{2\pi}{N}(-ux+vy)\right) - i\sin\left(\frac{2\pi}{N}(-ux+vy)\right)\right] = c+di$$

$$F(u,-v) = \sum_{x=-N/2}^{N/2} \sum_{y=-N/2}^{N/2} f(x,y)\left[\cos\left(\frac{2\pi}{N}(ux+-vy)\right) - i\sin\left(\frac{2\pi}{N}(ux+-vy)\right)\right] = c-di$$

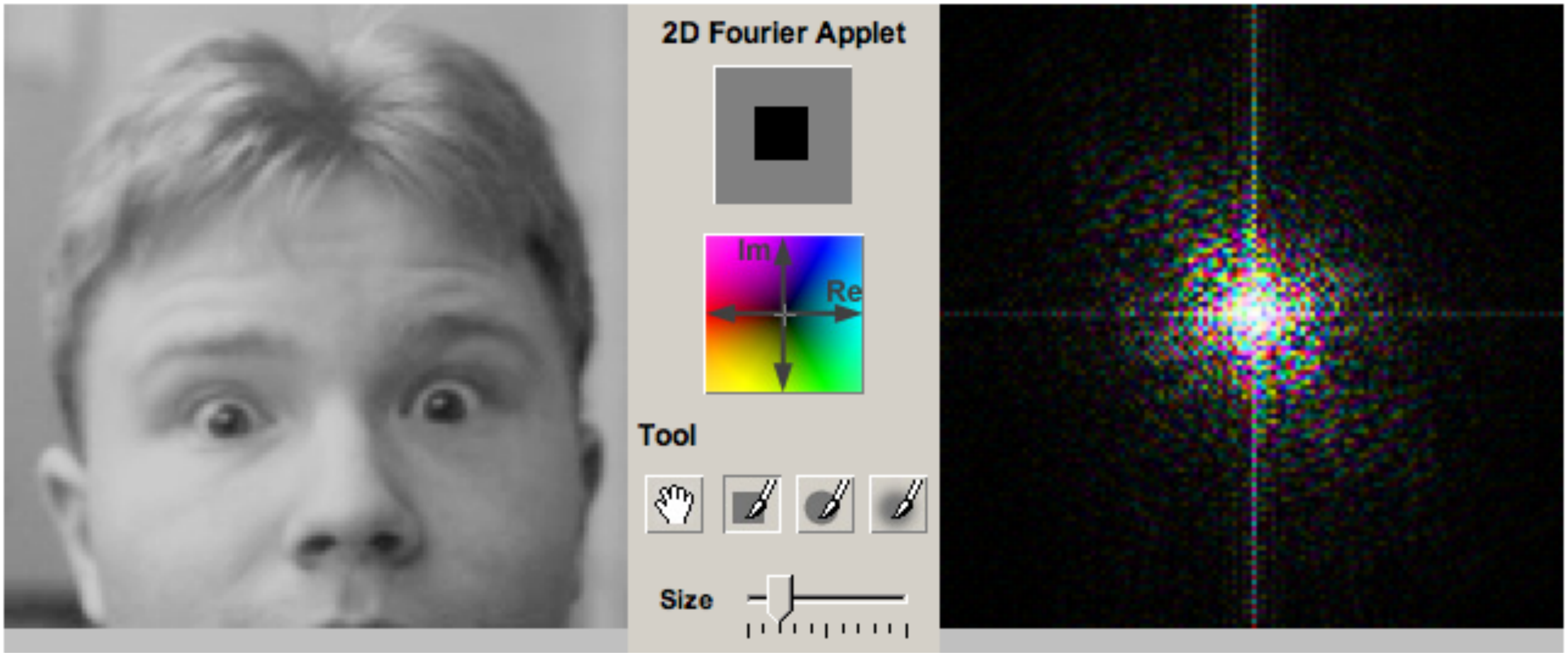
# 2D Frequency Bounds (cont)

- Size of 2D Frequency representation:
  - One dimension must vary from –N/2 to N/2, while the other varies from 0 to N/2
    - Doesn't matter which is which
  - N * (N/2) * 2 values per frequency = $N^2$
  - Same as the source spatial representation

# Showing Frequency Space

- To display a frequency space:
  - We plot it from –N/2 to N/2 in both dimensions
  - The result is symmetric about the origin (and therefore redundant)
  - We can't plot a complex number, so we show the magnitude at every pixel, i.e. sqrt($a^2 + b^2$)
    - Thus discarding the phase information
    - Phase plots are also possible ($\tan^{-1}(b/a)$)

# Showing Frequency Space




http://www.brainflux.org/java/classes/FFT2DApplet.html

# Frequency Space Matching

- The Fourier Transform converts spatial images into *frequencies* and *phases*. Can we use these to match images?
- The Fourier transform is deterministic, so if two images are identical, so are their frequencies and phases.
- Can we tell if two images are similar?

# Scaling Functions

- For simplicity, let us focus on 1D functions (the same principles will apply in 2D)
- If we scale a function f(x), we get f(ax), where a is a constant

$$F(f(ax)) = \int_{-\infty}^{\infty} f(ax)\left[\cos\left(\frac{2\pi ux}{N}\right) - i\sin\left(\frac{2\pi ux}{N}\right)\right]dx = \frac{1}{|a|}F\left(\frac{u}{a}\right)$$

- What does this say?

# Shifting functions

- Similarly, if we shift a function:

$$F\left(f\left(x-x_0\right)\right)=\int_{-\infty}^{\infty} f\left(x-x_0\right)\left[\cos\left(\frac{2\pi ux}{N}\right)-i\sin\left(\frac{2\pi ux}{N}\right)\right]dx$$

$$=F\left(u\right)\left[\cos\left(2\pi x_0 u\right)+i\sin\left(2\pi x_0 u\right)\right]$$

- The last term:
  - Represents a phase shift
    - Not a frequency shift
  - the amount of the phase shift is $2\pi x_0$
- The inverse is also true

# Convolution

- The definition of convolution is:

$$h(x) = f(x) \otimes g(x) = \int_{-\infty}^{\infty} f(u) g(x+u) du$$

  - The mask g() is assumed infinite but zero outside of a finite range.
  - Since the images are ZMUL, cross-correlation is almost an example of convolution

# Correlation Theorem

- The Fourier transform of the cross-correlation is: $$H(x) = F(s)G^*(s)$$
  - F(s) is the Fourier transform of f(x)
  - G(s) is the Fourier transform of g(x)
    - G*(s) is the complex conjugate of G(s), defined as the real part of G(s) *minus* the imaginary part
  - H(s) is the Fourier transform of f(x)⊙g(x)

# So?

- Go back to the shifting relation. The phase difference between two identical images shifted by $(x_0,y_0)$ is:

$$\cos\left(2\pi\left(ux_0,vy_0\right)\right)+i\sin\left(2\pi\left(ux_0,vy_0\right)\right)=e^{i2\pi\left(ux_0,vy_0\right)}$$

- Why is this interesting? Because:

$$\frac{F_1(u,v)F_2^*(u,v)}{\left|F_1(u,v)F_2^*(u,v)\right|}=\cos\left(2\pi\left(ux_0,vy_0\right)\right)+i\sin\left(2\pi\left(ux_0,vy_0\right)\right)$$

# Back to Correlation

- If two images are similar, the maximum phase value corresponds to the best shift $x_0$, $y_0$.
- What is the relationship between the height of the peak and the correlation score of the images under that shift?
- Parseval's theorem says that: $\int_{-\infty}^{\infty} h^2(x)dx = \int_{-\infty}^{\infty} |H(f)|^2 df$
- So normalizing source images normalizes frequency space, and the height of the peak is the correlation score

# This is important…

- To match two (same-sized) images over all possible translations:
  - Cross-correlate in the spatial domain
    - $O(n^2)$
  - Or
    - Apply DFFT to put images in frequency domain
      - O(n log(n))
    - Multiply their frequency representations
      - O(n)
    - Convert back to the spatial domain
      - O(n log(n))

# Frequency space matching

- What about the boundaries?
  - If two same-sized images are correlated, aren't some pixels off the edge?
  - F(u) assumes an infinitely repeating pattern
  - Equivalent to "wrap-around"
- Otherwise, linear correlation gives the same result in the spatial and frequency domains