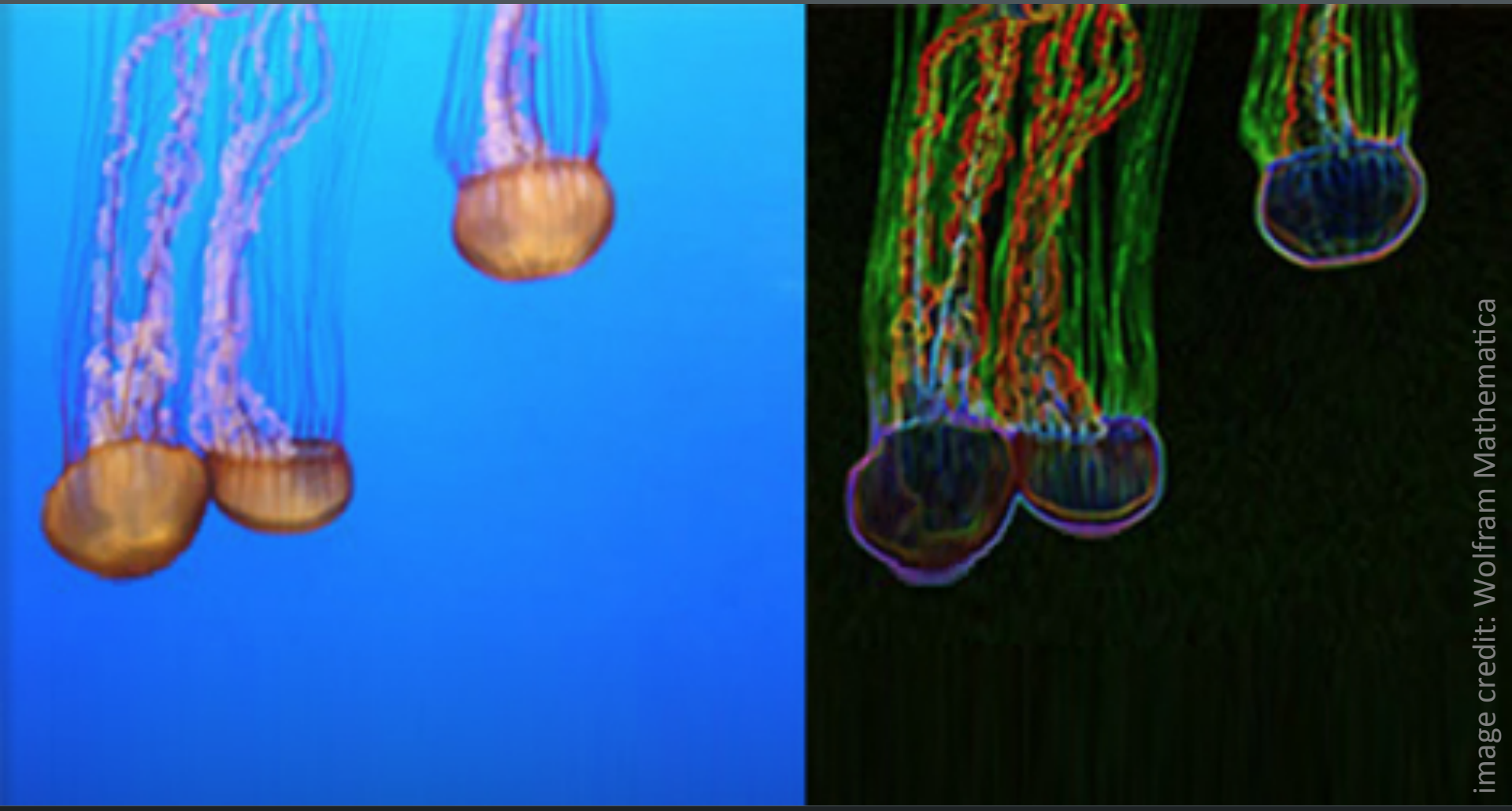


Image Processing

also today: paper discussion

CS 4300/5310

Computer Graphics

ANNOUNCEMENTS

Deadlines

- 2D Project Proposal: today!
  - Submit one per group
- 2D Project main deadline: February 5th

Project Scoping Workshop: Thursday

- Each group brings **hardcopy** of project proposal to class

- Bring laptops if possible

Global Game Jam!

Go to www.northeastern.edu/games/ggj and register now!

IMAGE PROCESSING

Working with Images

- Image Understanding
  - input: image; output: high level understanding
- Image Analysis
  - input: image; output: features
- Image Processing
  - input: image; output: image

Image Understanding

Image Analysis

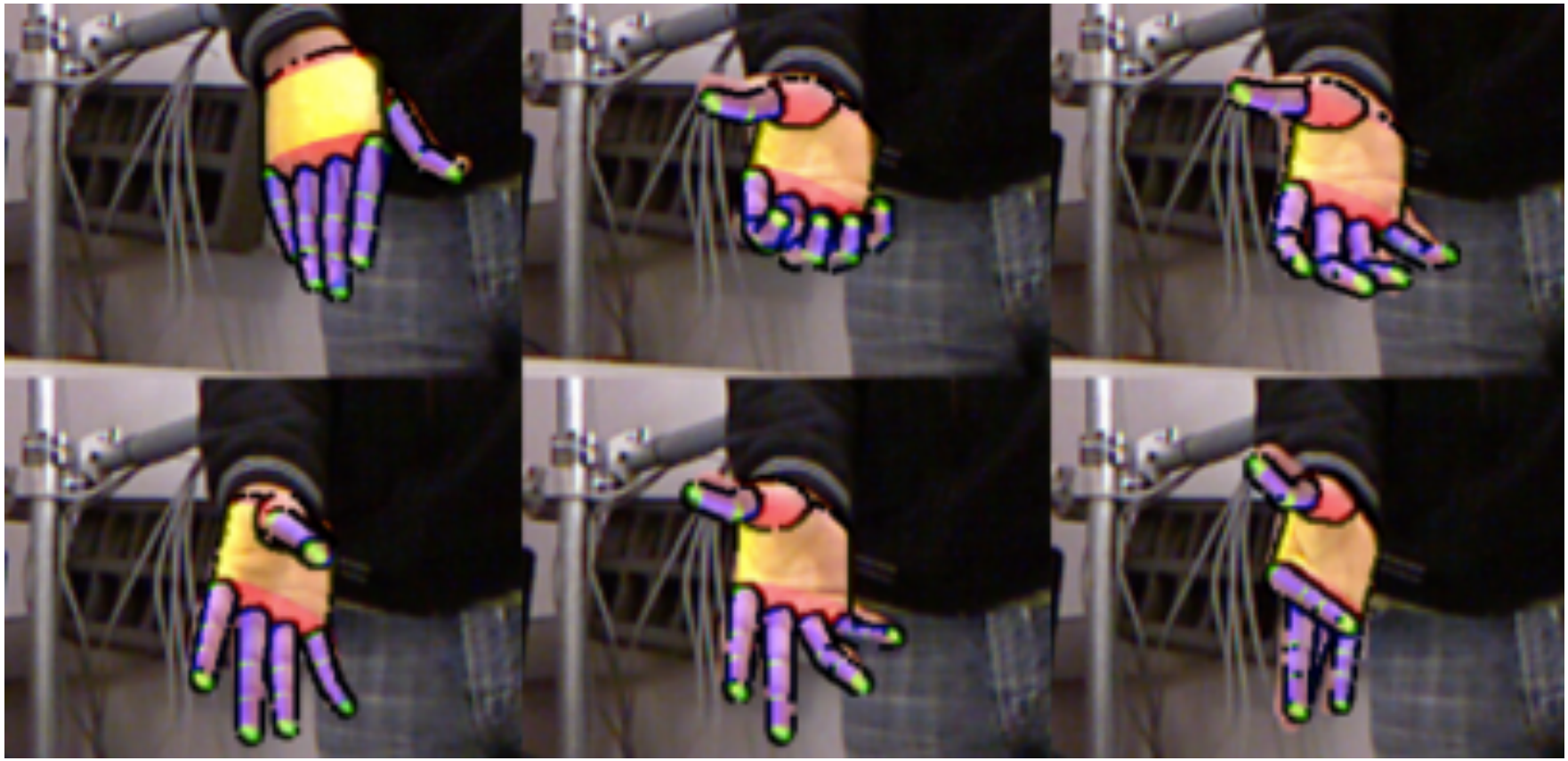
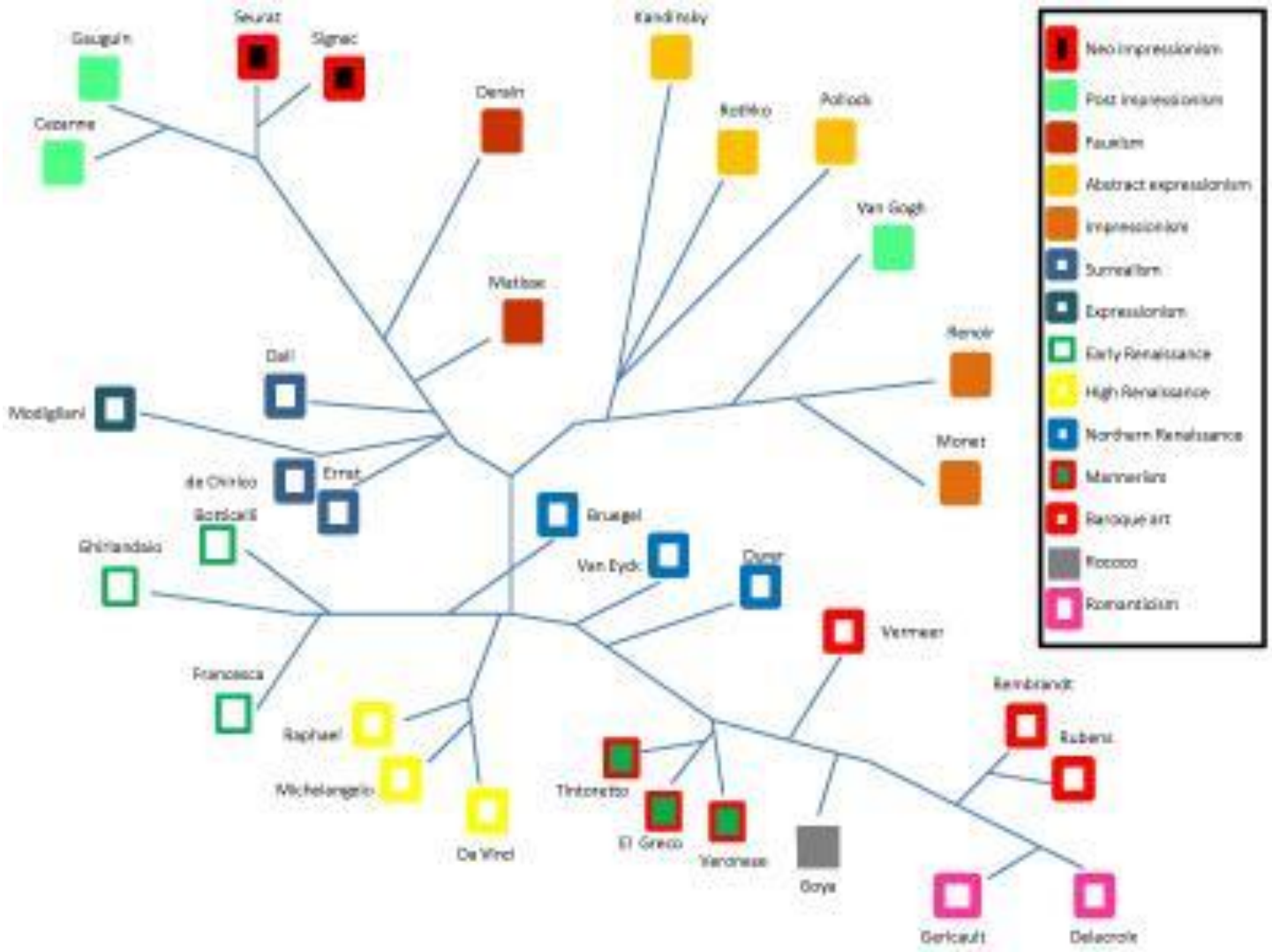


Image Analysis



http://www.sciencedaily.com/releases/2012/09/120926094546.htm

Image Processing

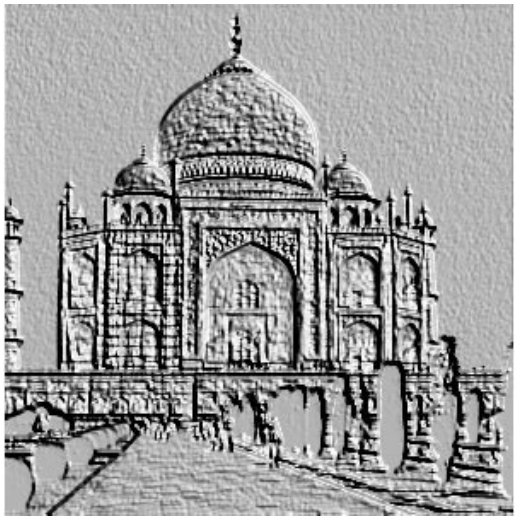


Image Processing

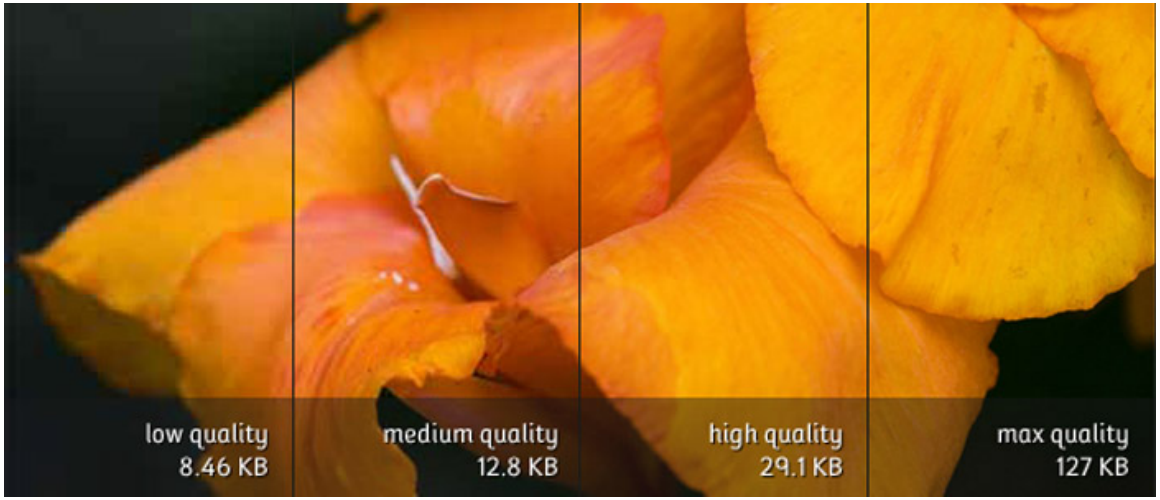
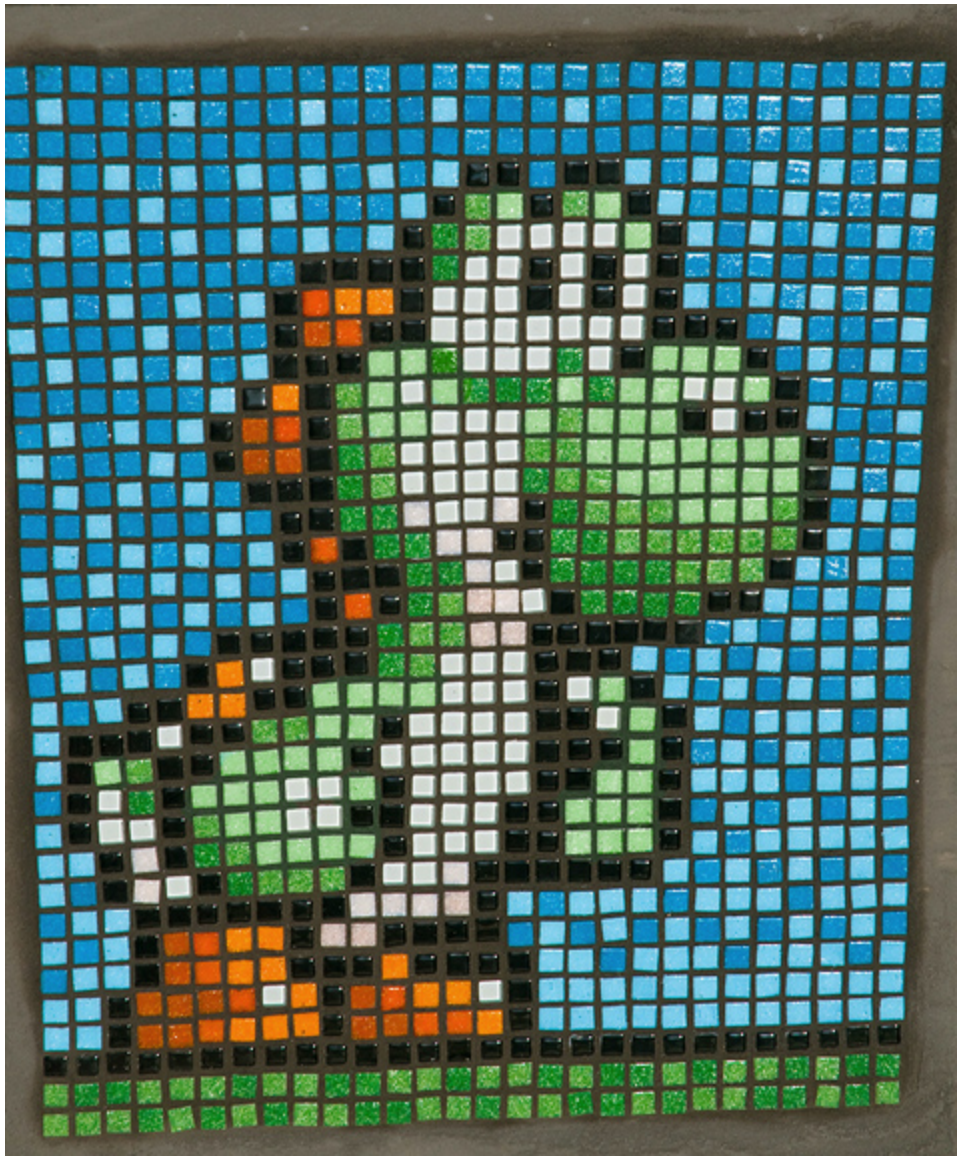


Image Processing

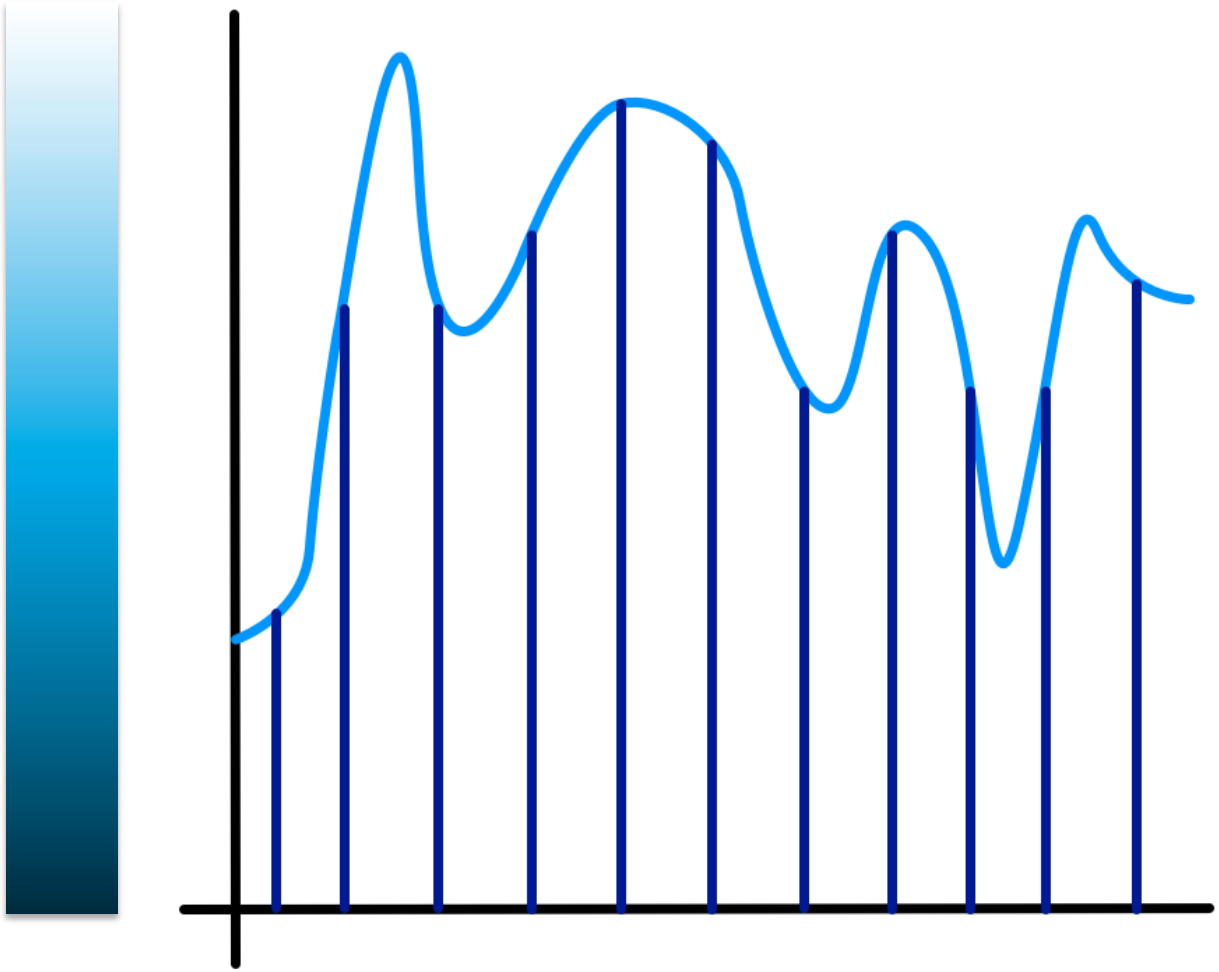
- Image transformations and warping

- Image filtering
  - Spatial domain
  - Frequency domain

Review: What is an Image?



A Sample from a Continuous Function



A Sample from a Continuous Function

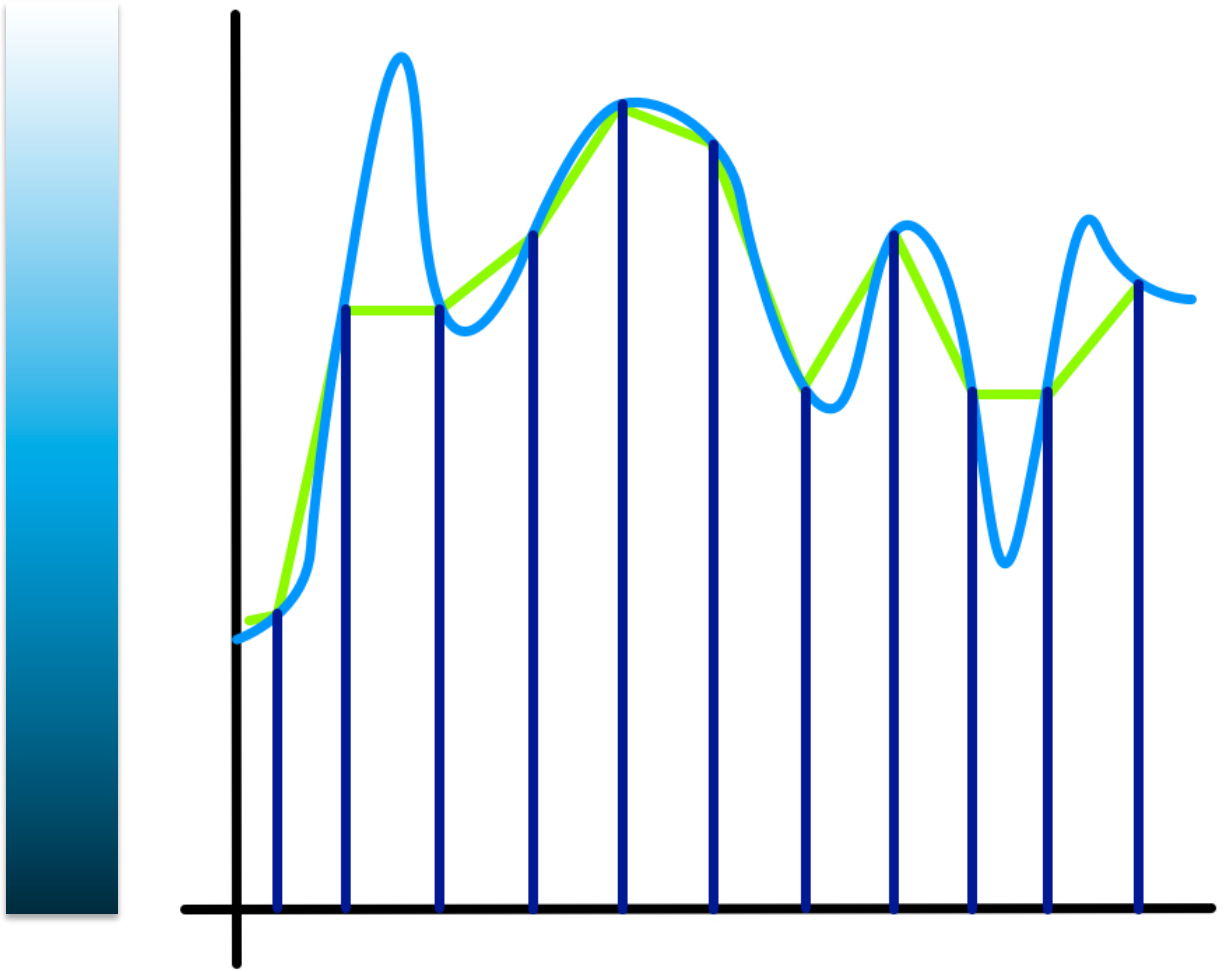
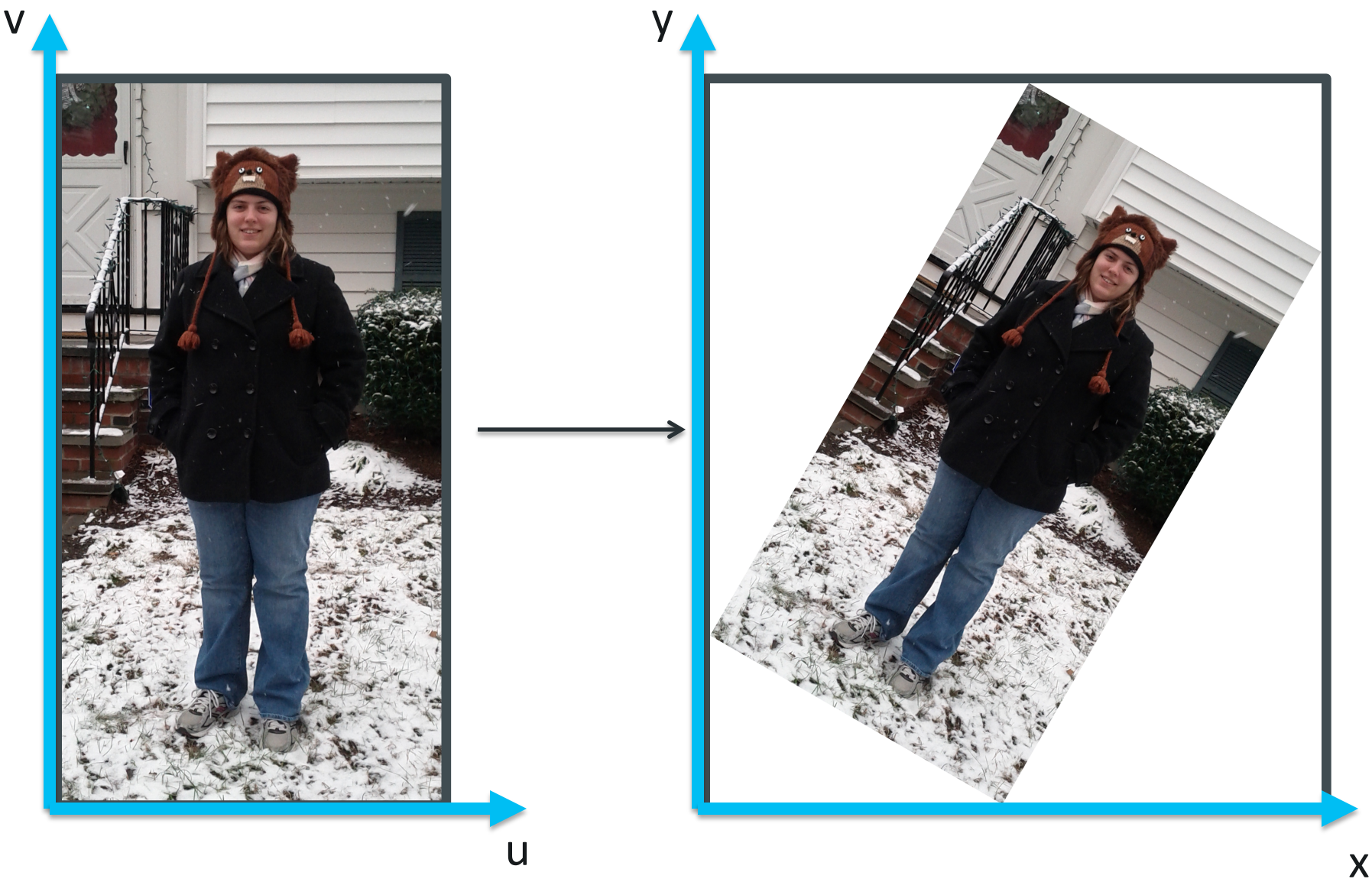
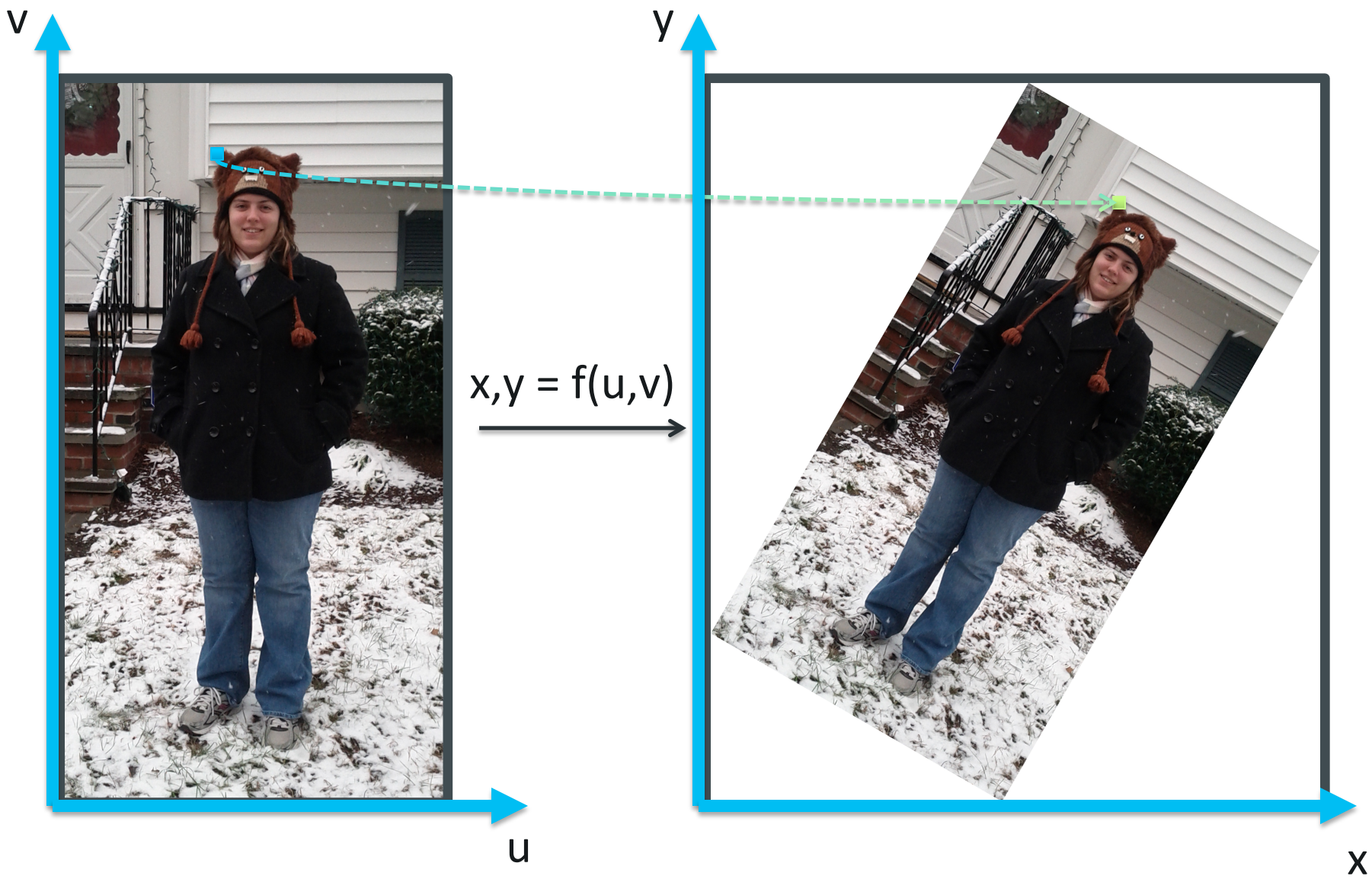


IMAGE WARPING

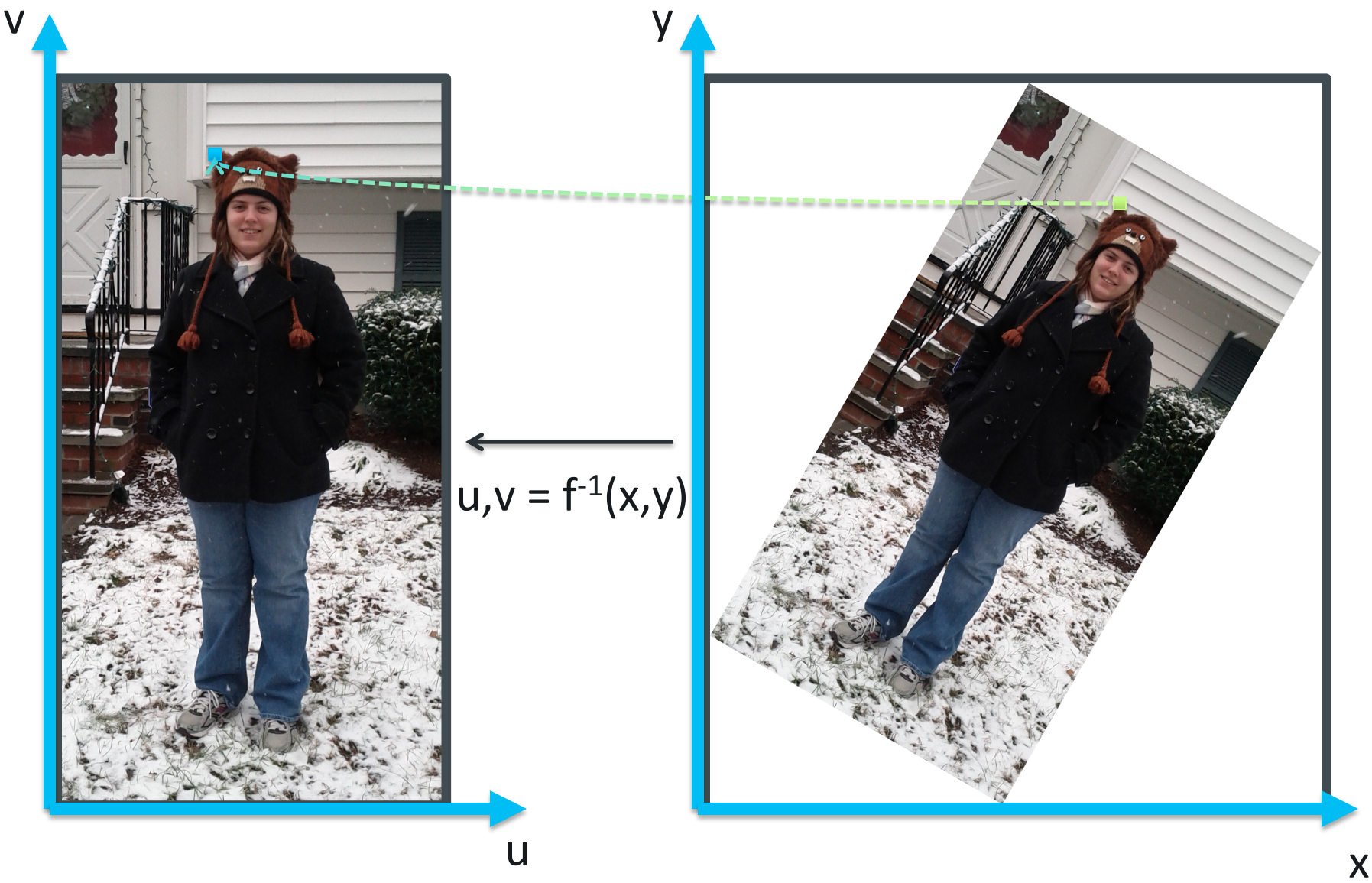
Mapping



Mapping



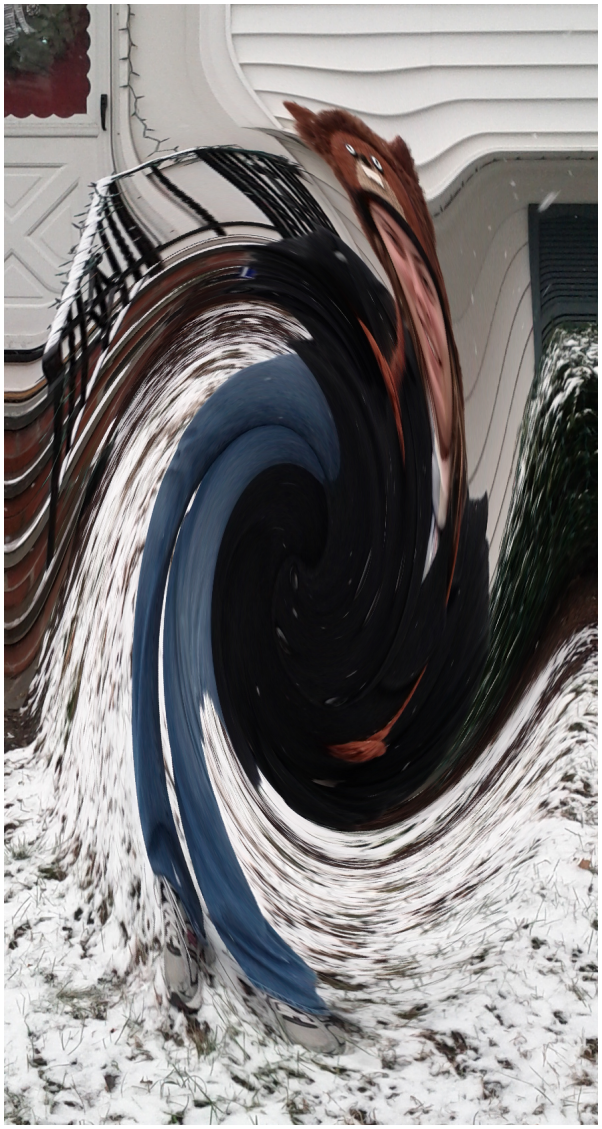
Mapping



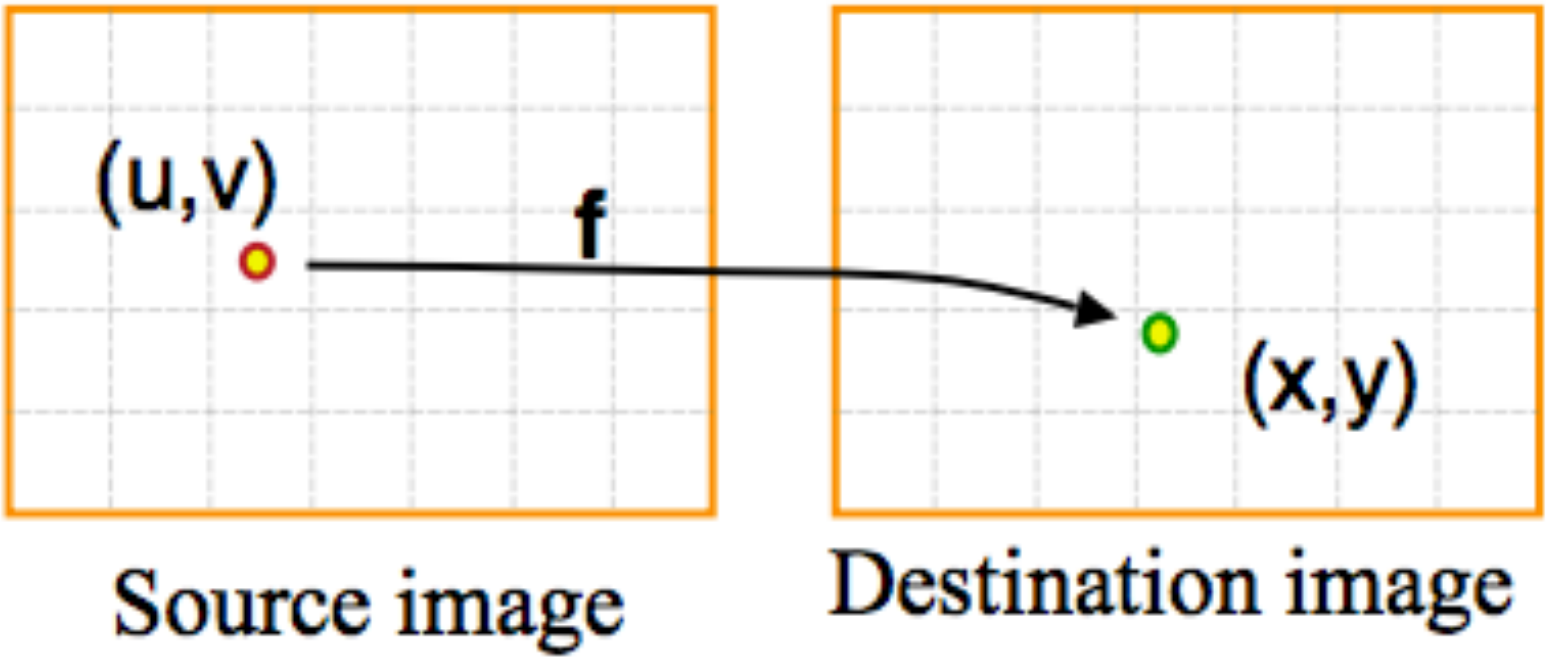
Example Mappings

- Rotate
  - $x = u \cos(\theta) - v \sin(\theta)$
  - $y = u \sin(\theta) + v \cos(\theta)$
- Scale
  - $x = s_x u$
  - $y = s_y v$
- Shear (by X)
  - $x = u + s_x v$
  - $y = v$

Example Mappings

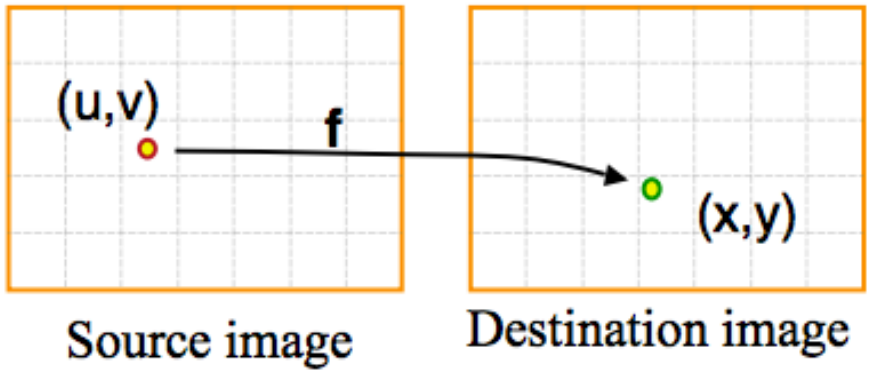


Forward Mapping

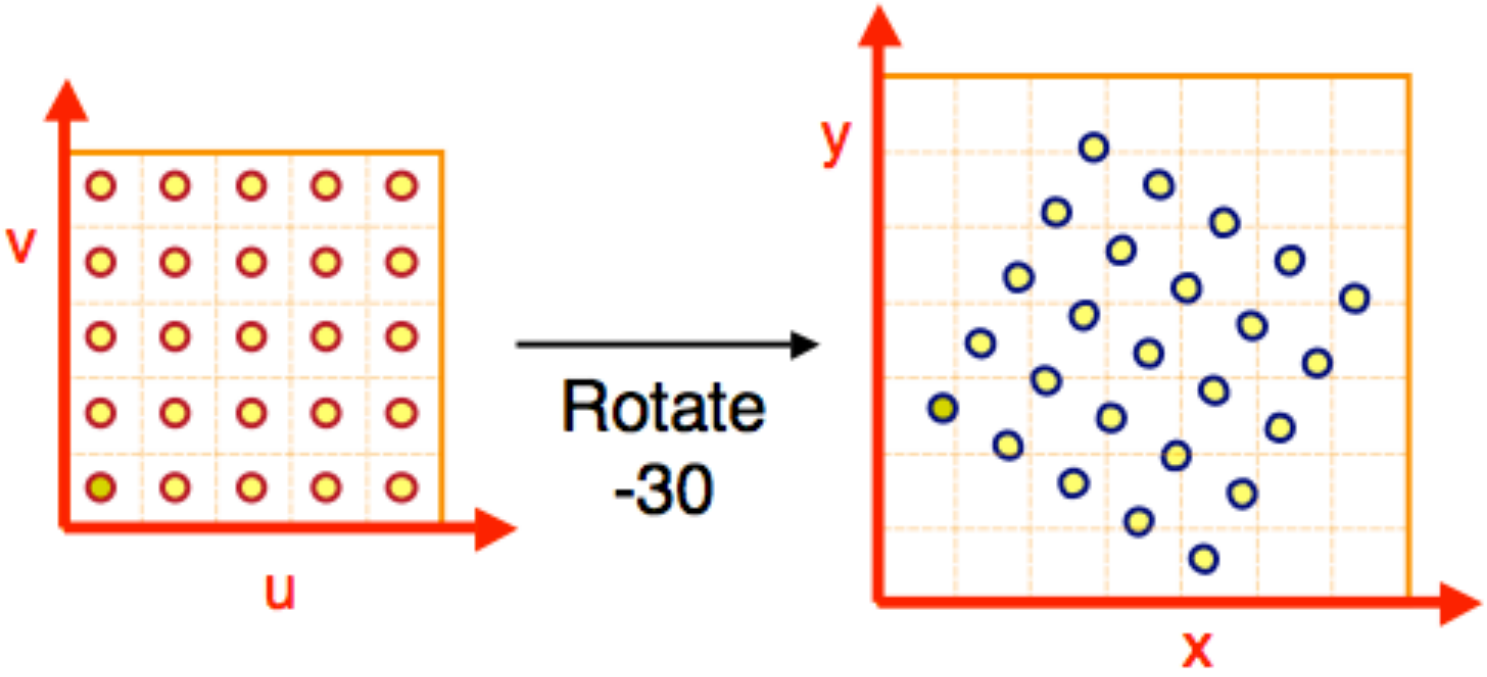


Forward Mapping

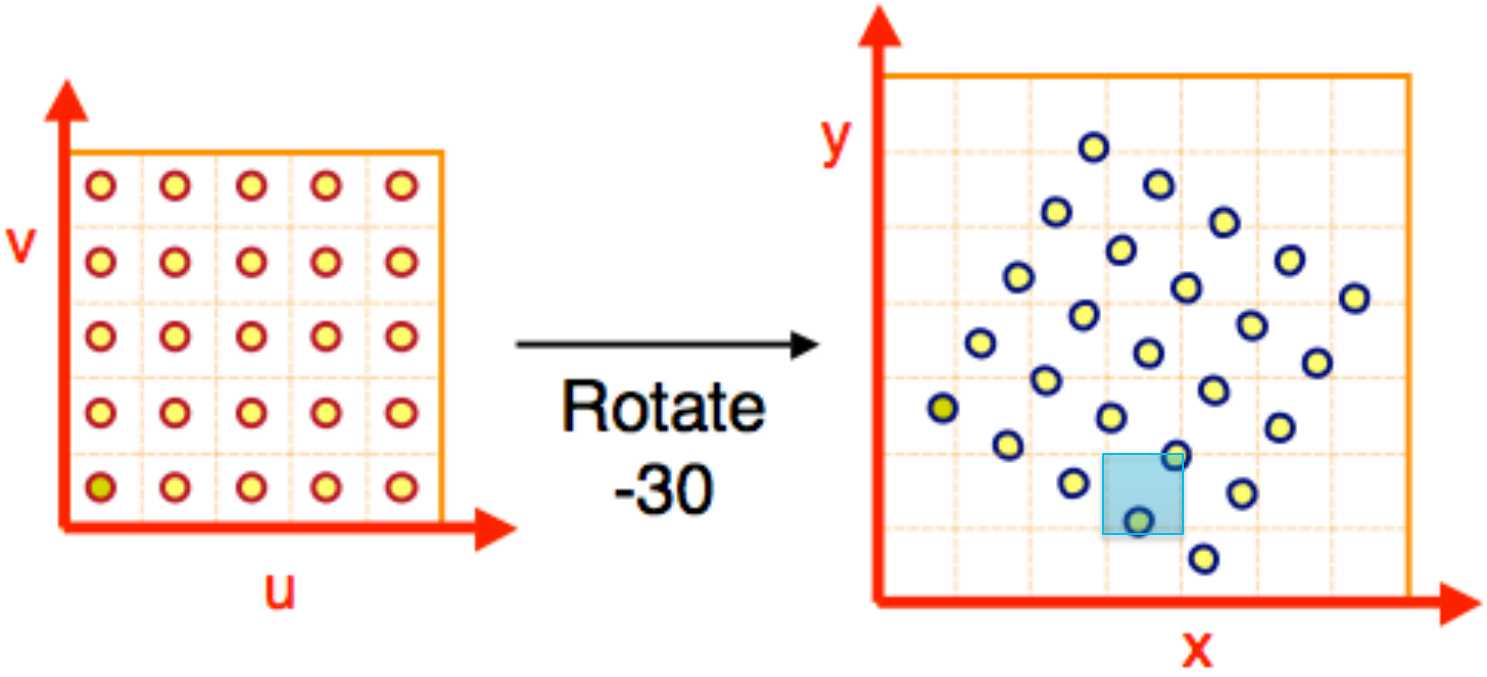
```
for u in range src_width:
        for v in range src_height:
                x, y = f(u, v)
                dest(x, y) = src(u, v)
```



Forward Mapping: Why You Shouldn't Do It

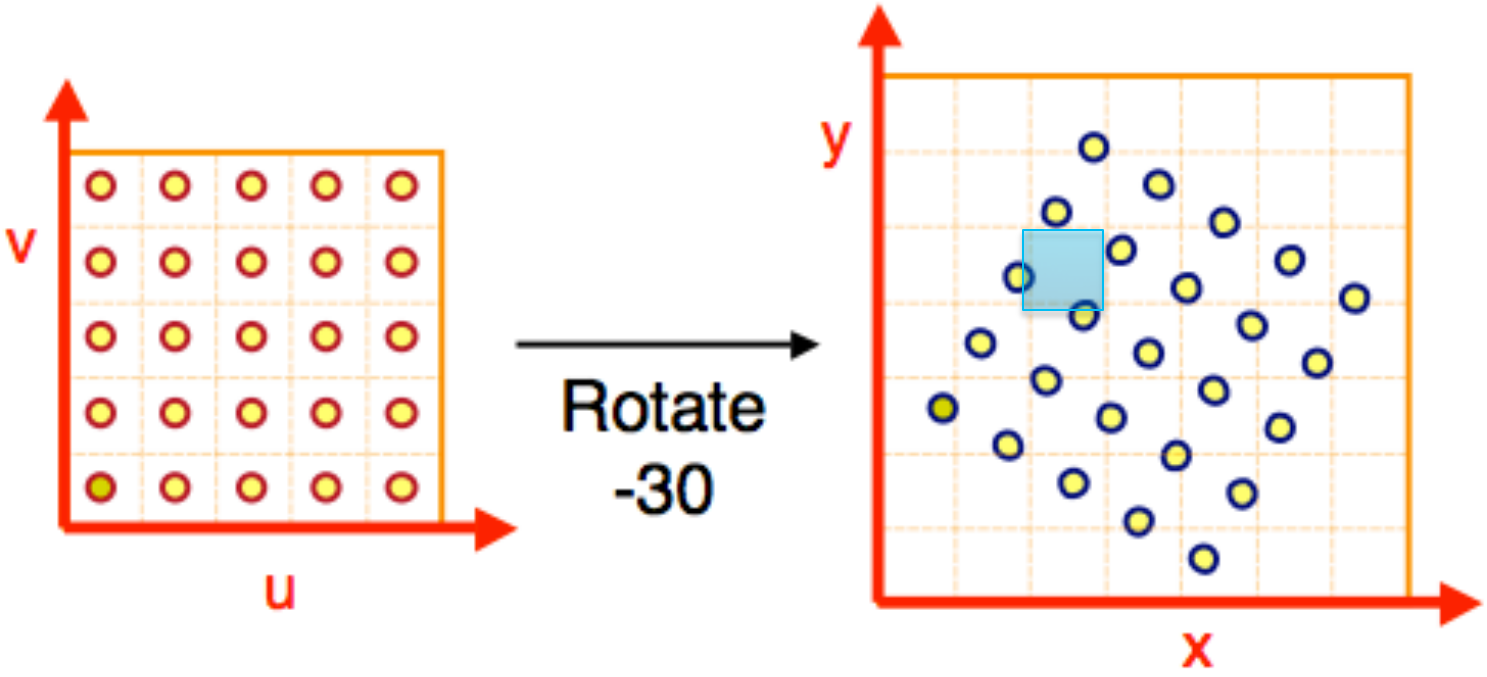


Forward Mapping: Why You Shouldn't Do It



One destination pixel with two source pixels!

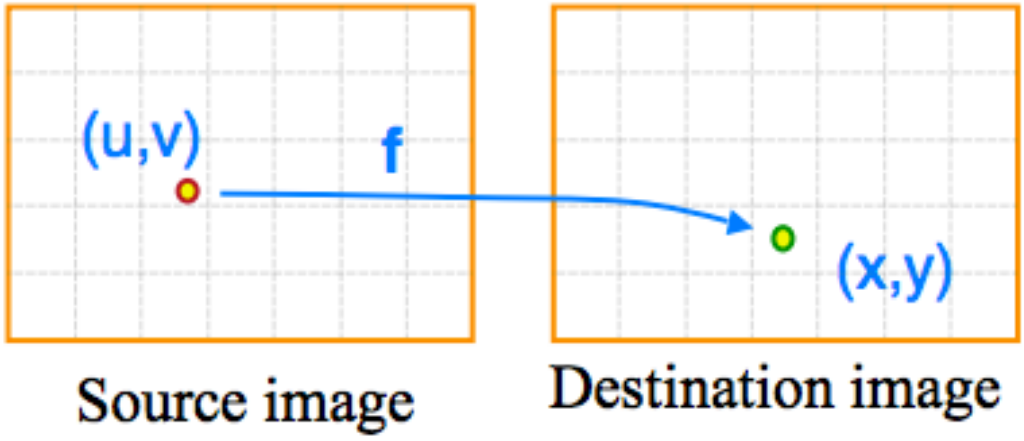
Forward Mapping: Why You Shouldn't Do It



One destination pixel with **zero** source pixels!

Inverse Mapping

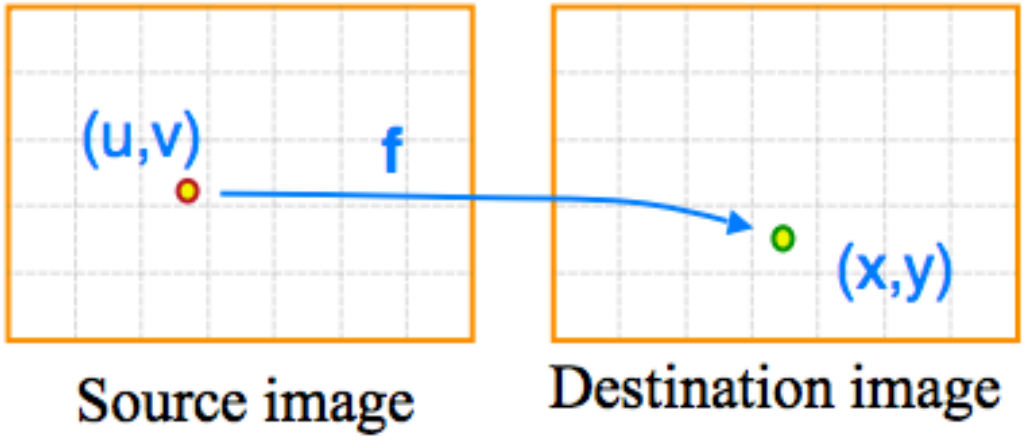
```
for x in range dest_width:
    for y in range dest_height:
        u, v = f^-1(x, y)
        dest(x, y) = src(u, v)
```



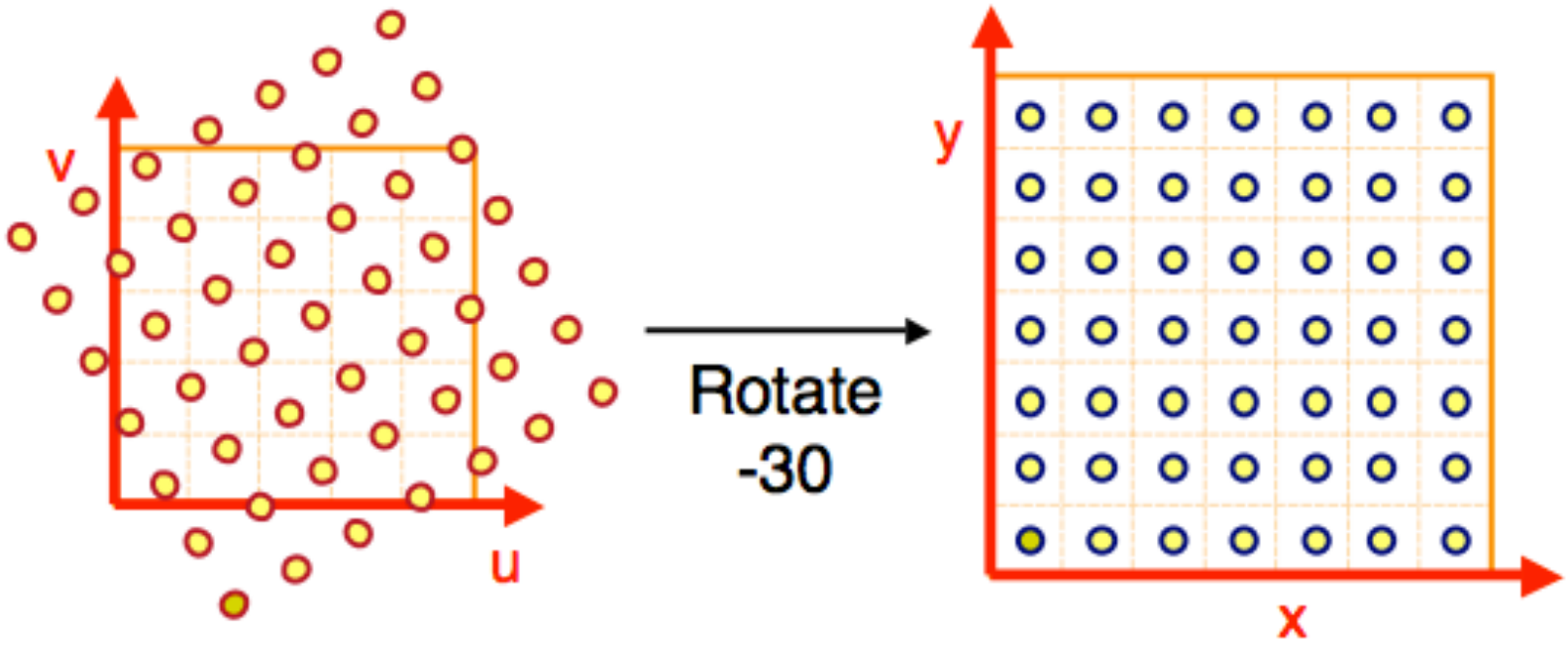
Inverse Mapping

```
for x in range dest_width:
    for y in range dest_height:
        u, v = f^-1(x, y)
        dest(x, y) = src(u, v)
```

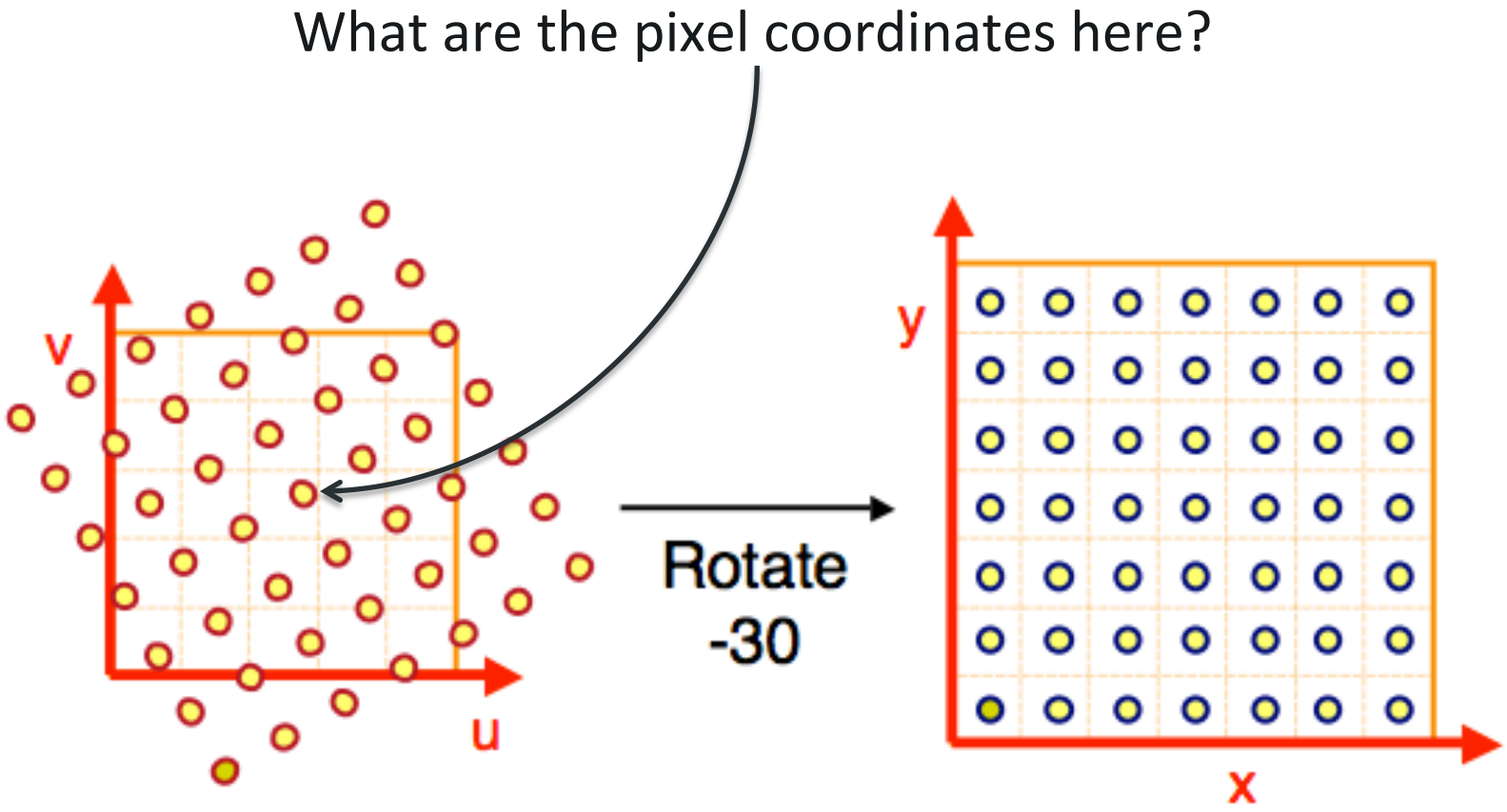
???



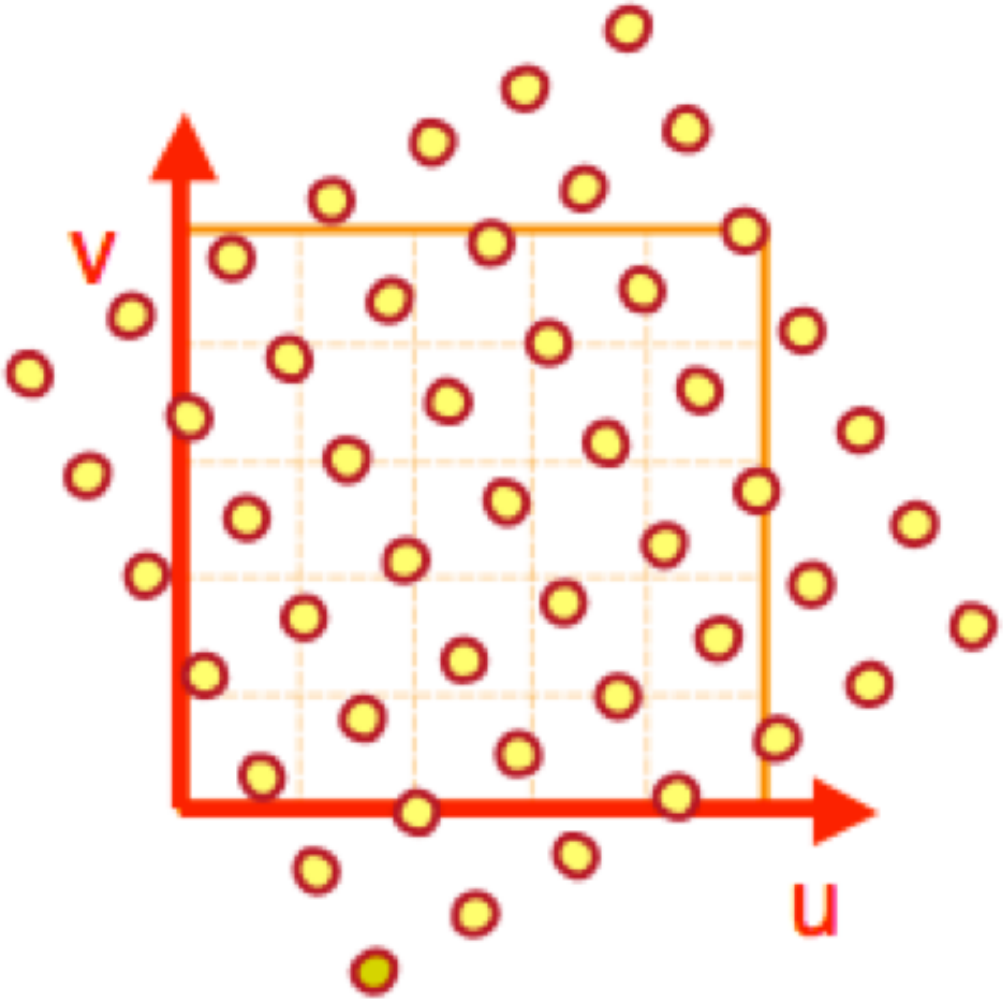
Inverse Mapping



Resampling



Point Sampling



u = int(u’ + 0.5)

v = int(v’ + 0.5)

Triangle Filtering

Find the four closest pixels in src image

a = interpolated color between (u_1,v_2) and (u_2,v_2)

b = interpolated color between (u_1,v_1) and (u_2,v_1)

dest(x,y) = interpolated color between *a* and *b*

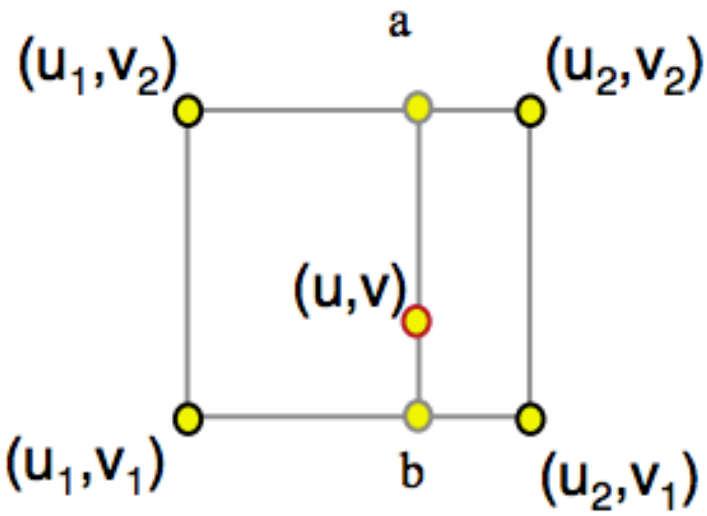


IMAGE FILTERING

Filtering

- Accentuate particular image features
  - Edge detection
  - Sharpening
- Remove sampling artifacts
  - Antialiasing
- Effects
  - Motion blur

Discrete Convolution

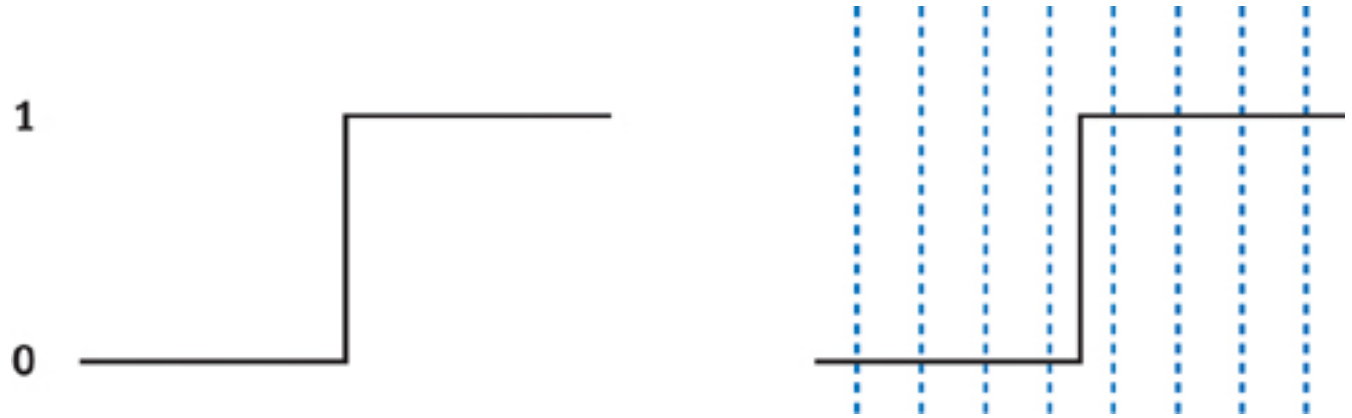
- Filter is an n x n matrix
- Pass filter across input image
  - Align center of filter with each pixel
- New pixel value: weighted average of old pixel values

http://beej.us/blog/data/convolution-image-processing/

Kinds of Aliasing

- Intensity
  - Not enough colors
- Spatial
  - Not enough image resolution
- Temporal
  - Not enough samples of moving image

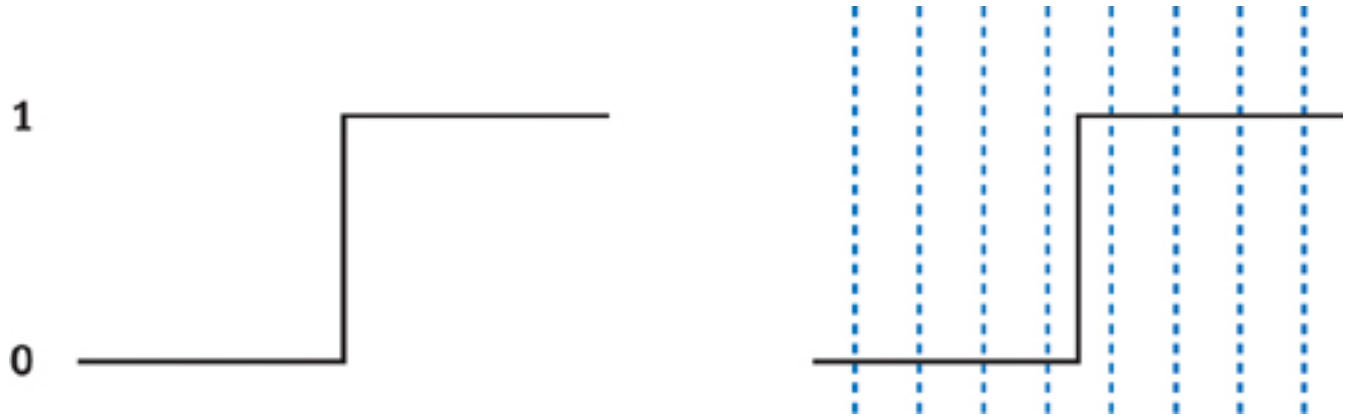
Thinking about Frequencies

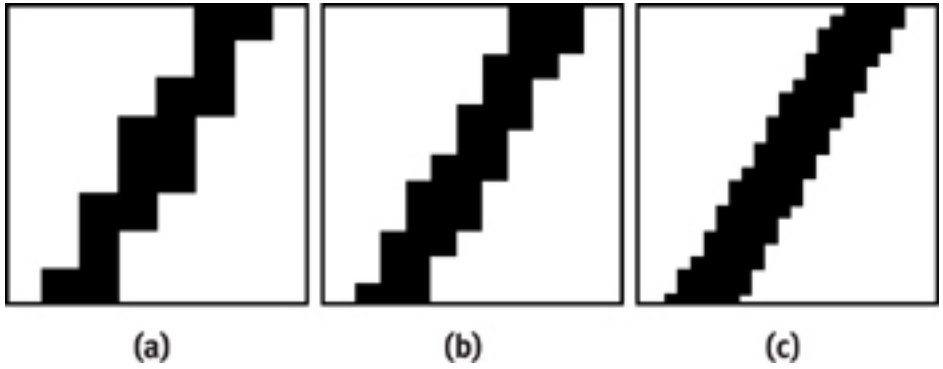


- A line drawn on a piece of paper is like a step function
  - 0: nothing is drawn
  - 1: full intensity
- Frequency of sampling?

source: GPU Gems 2, Ch. 22

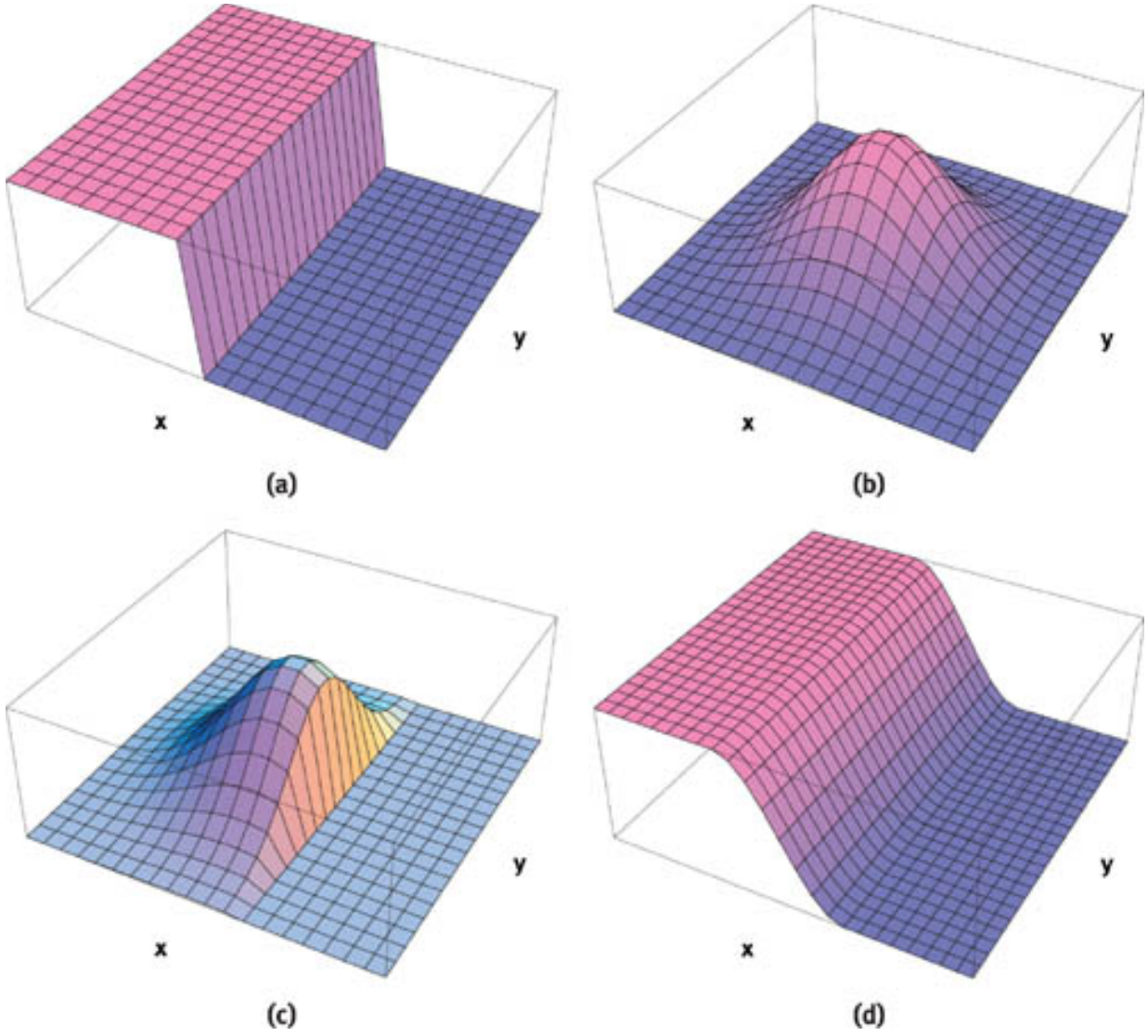
Thinking about Frequencies





source: GPU Gems 2, Ch. 22

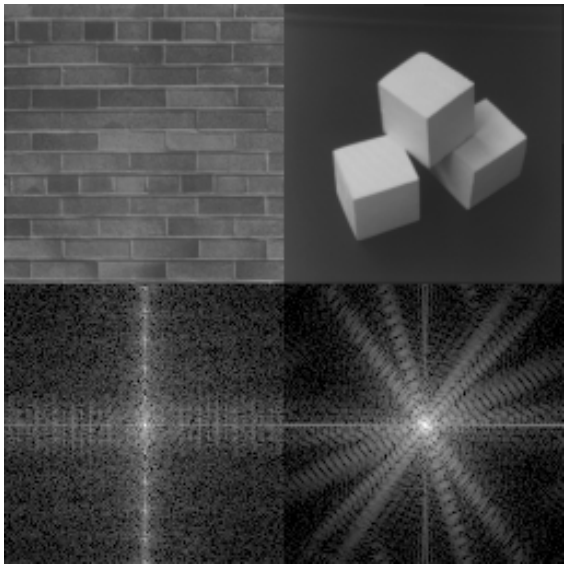
Anti-aliasing

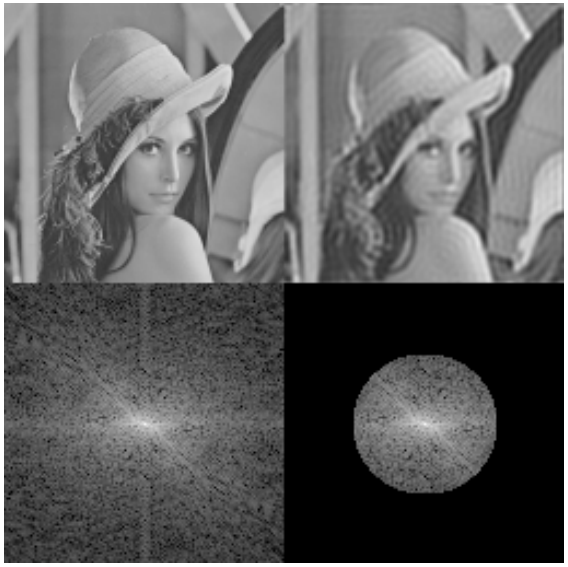


source: GPU Gems 2, Ch. 22

But isn't matrix convolution really slow?

- Not in the frequency domain!
  - Matrix convolution in spatial domain = multiplication in the frequency domain
  - Fast fourier transform (and inverse) takes you between the two representations





source:
http://www.cs.auckland.ac.nz/courses/compsci773s1c/lectures/ImageProcessing-html/fourier.html

PAPER DISCUSSION