CS 322: Image Processing and Computer Vision

Sampling and Aliasing

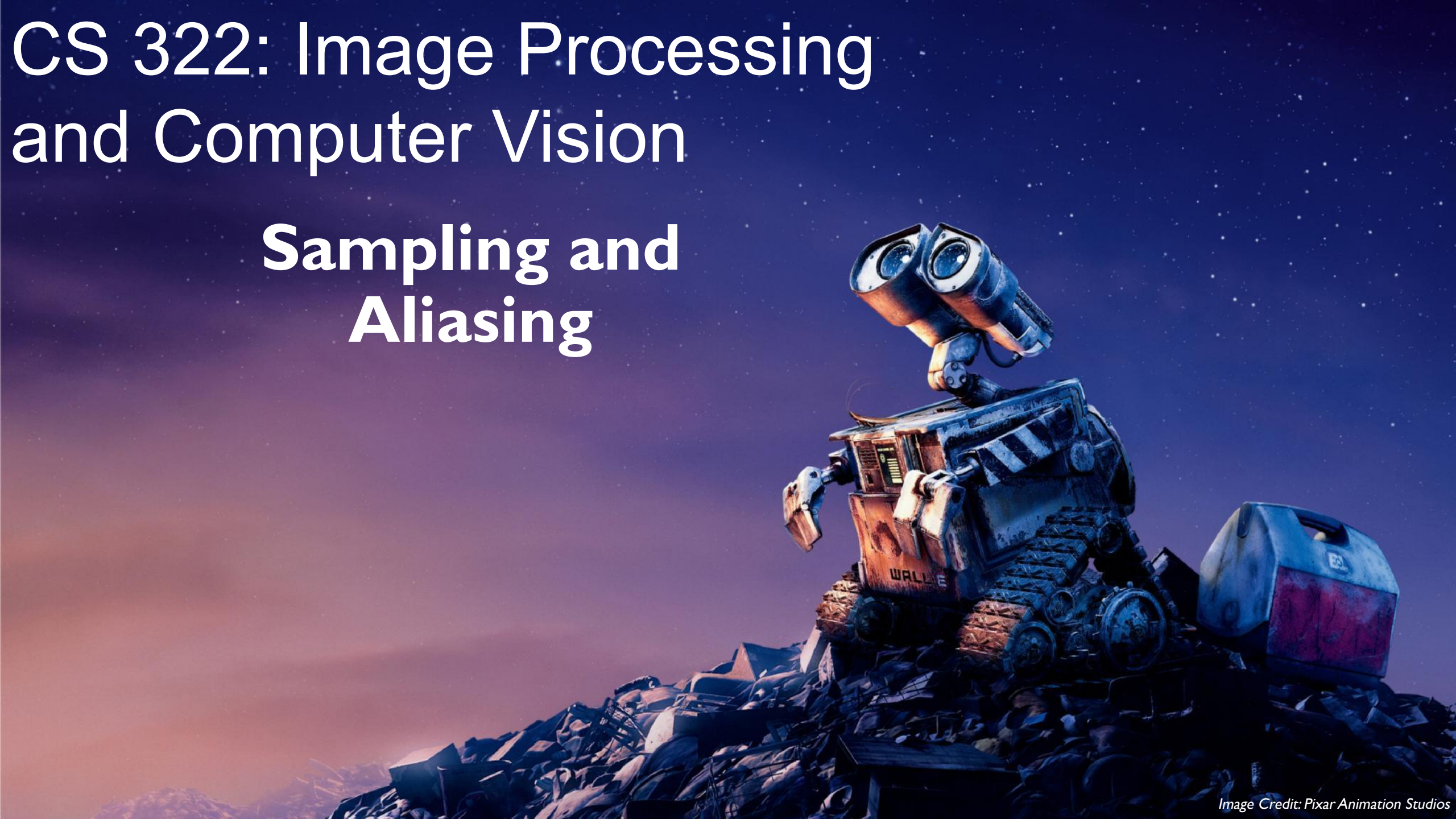


Image Credit: Pixar Animation Studios

CS 322: Image Processing and Computer Vision

Last lecture

- Filtering
- Filtering Properties
- Correlation & Convolution
- Non-linear Filters

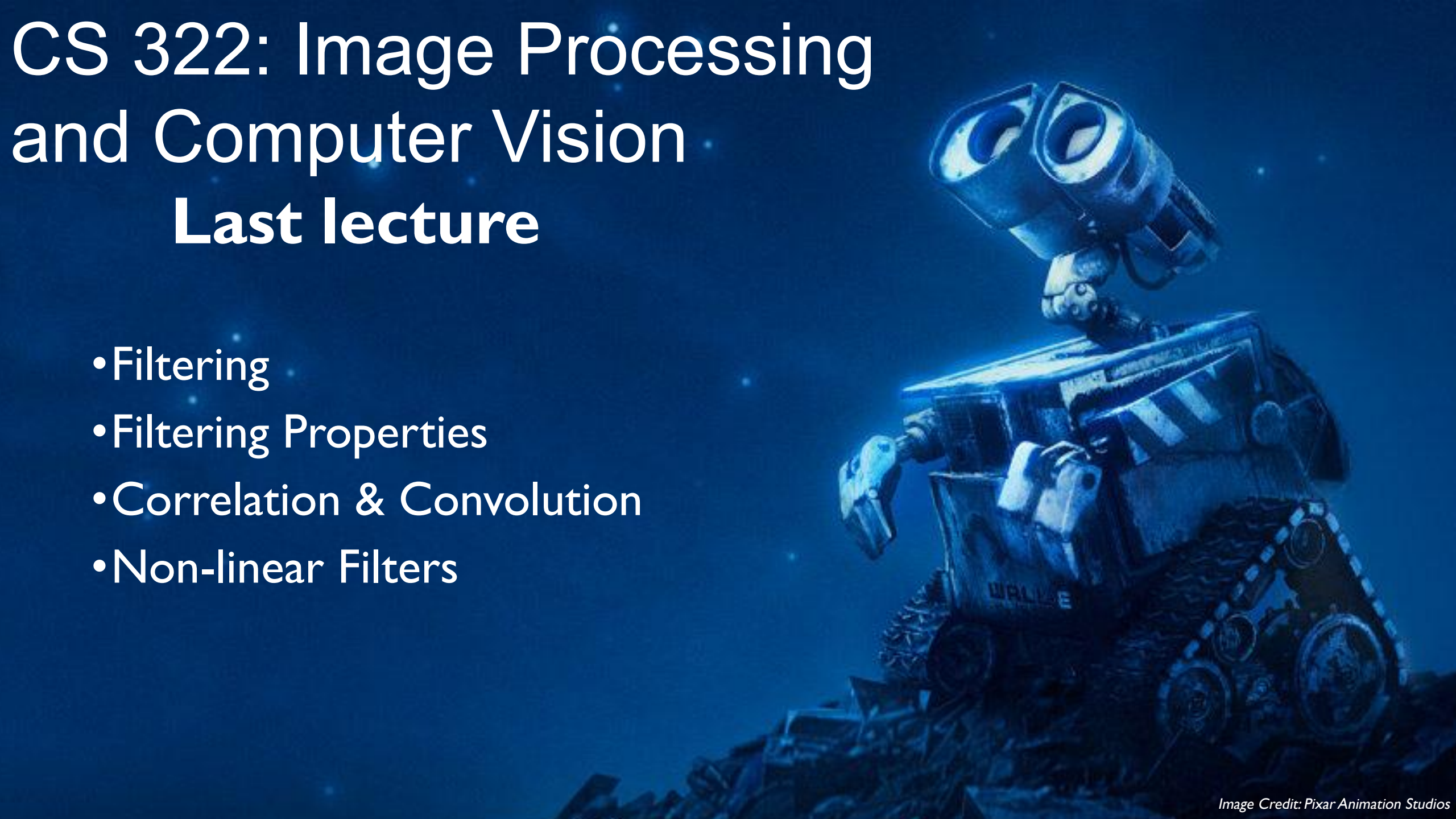
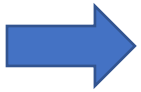


Image Credit: Pixar Animation Studios

Why does a lower resolution image still make sense to us? What information do we lose?

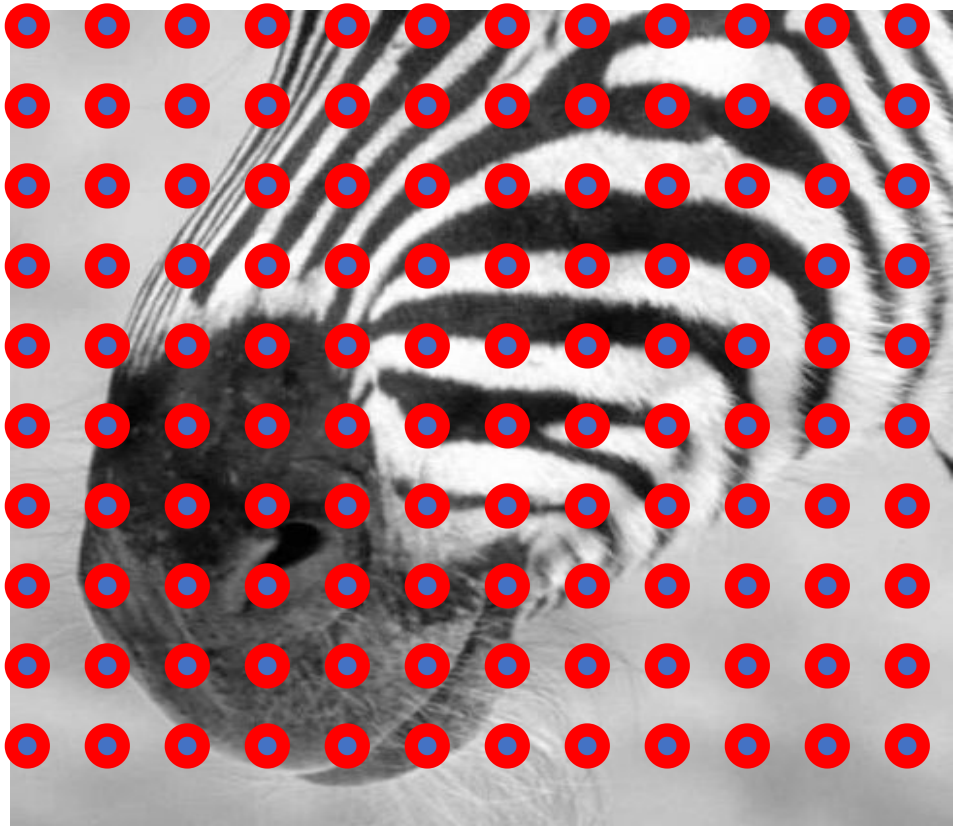


Slide: Derek Hoiem, Image: http://www.flickr.com/photos/igorms/136916757/

Making lower resolution images: Subsampling by a factor of 2

Throw away every other row and column to create a 1/2 size image

Making lower resolution images: Subsampling by a factor of 2



Throw away every other row and column to create a 1/2 size image

512 256 128 64 32 16 8

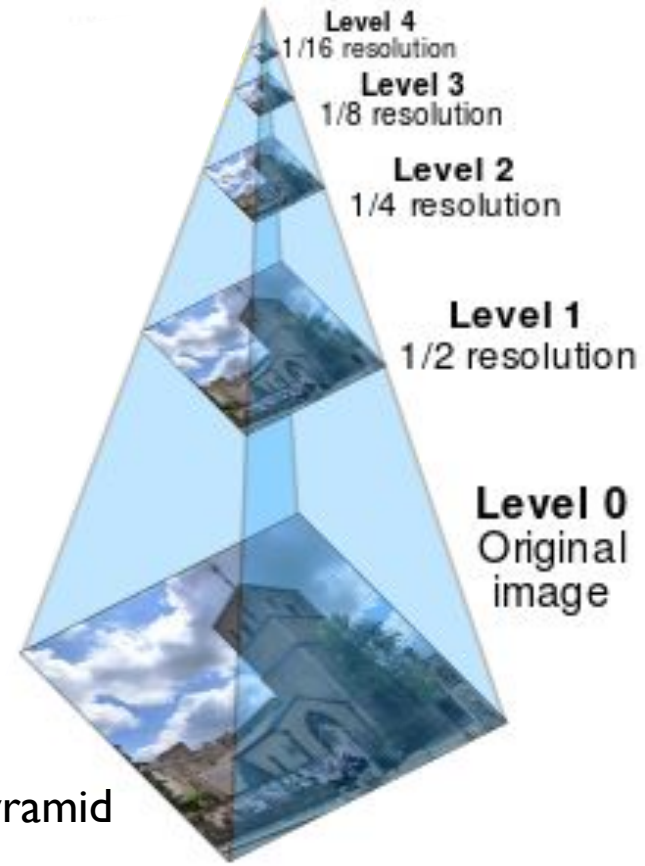


Image Pyramid

Downsampling

Algorithm

1. Start with image of w x h
2. Sample every other pixel
3. `im_small = image[ ::w, ::w ]`

4. To build a pyramid,
   repeat Steps 1 & 2
   until `im_small` is 1 pixel large.

Downsampling

Algorithm

1. Start with image of w x h
2. Sample every other pixel
3. `im_small = image[ ::w, ::w ]`

4. To build a pyramid,
   repeat Steps 1 & 2
   until `im_small` is 1 pixel large.

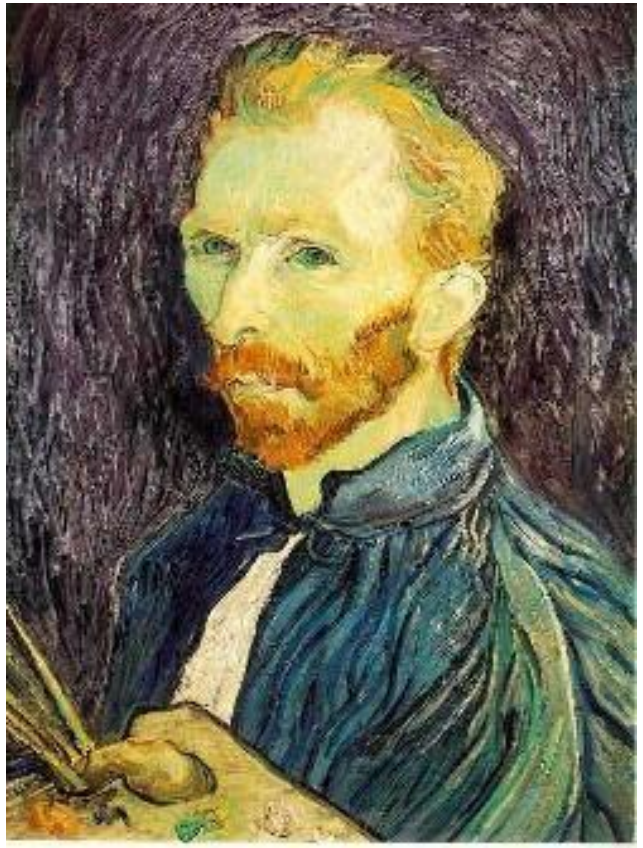
NumPy Syntax

w = 2
::2 -> start at 0,
end at 'end',
increase every 2, until
the end.

e.g.,
0,2,4,6,…,w

(if w is not even, then
this goes to w-1)

Image Subsampling

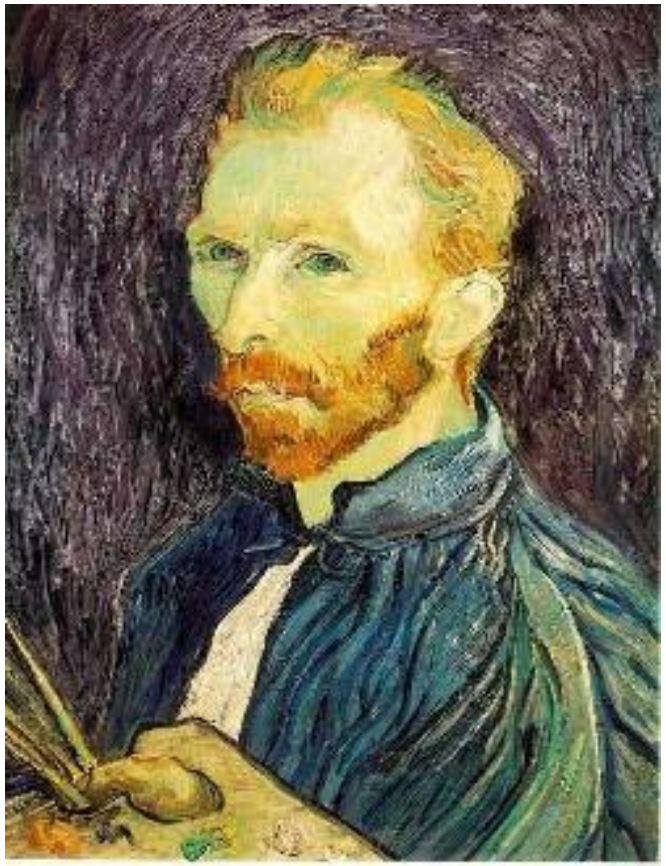


1/4

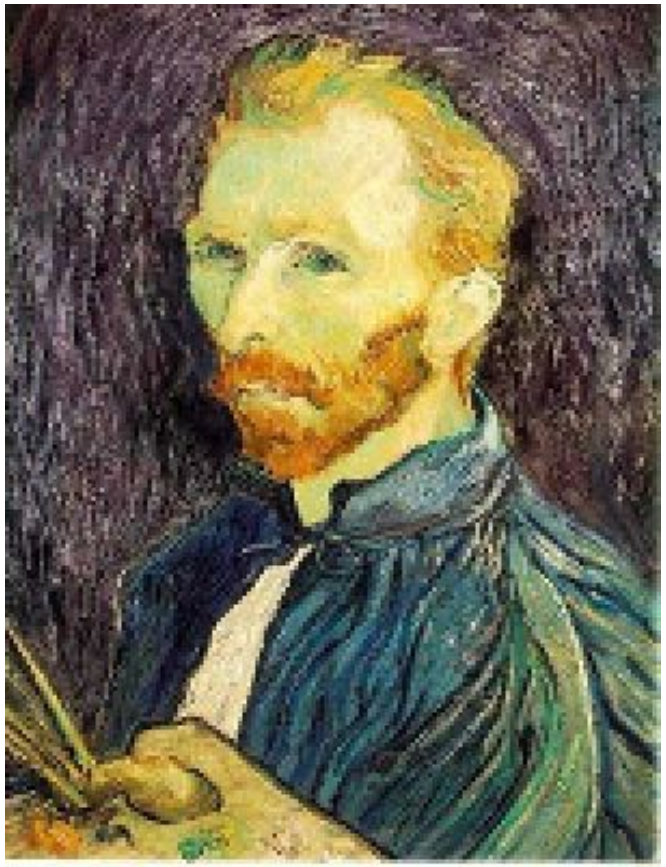
1/8

Throw away every other row and column to create a 1/2 size image.

Subsampling without Filtering



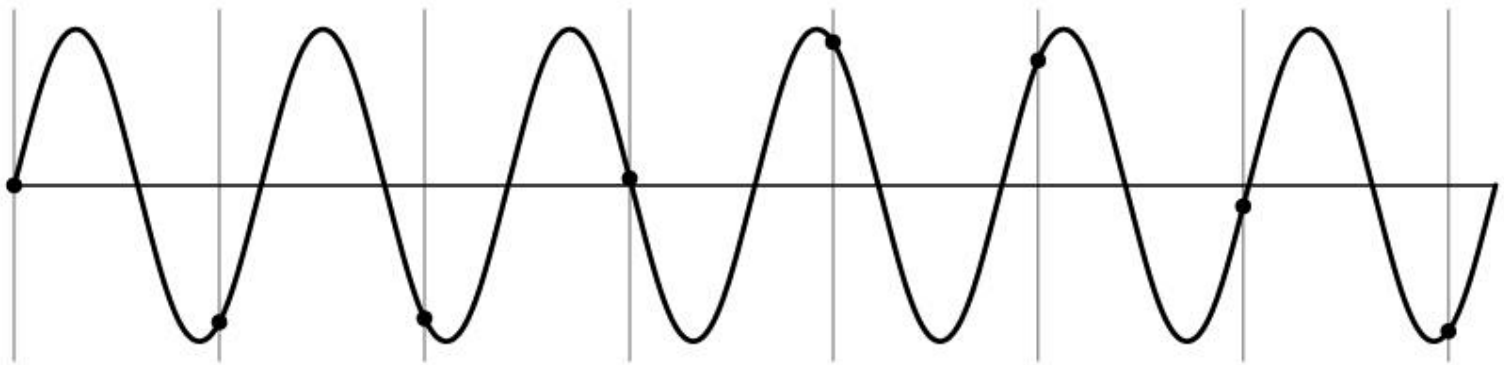
1/2



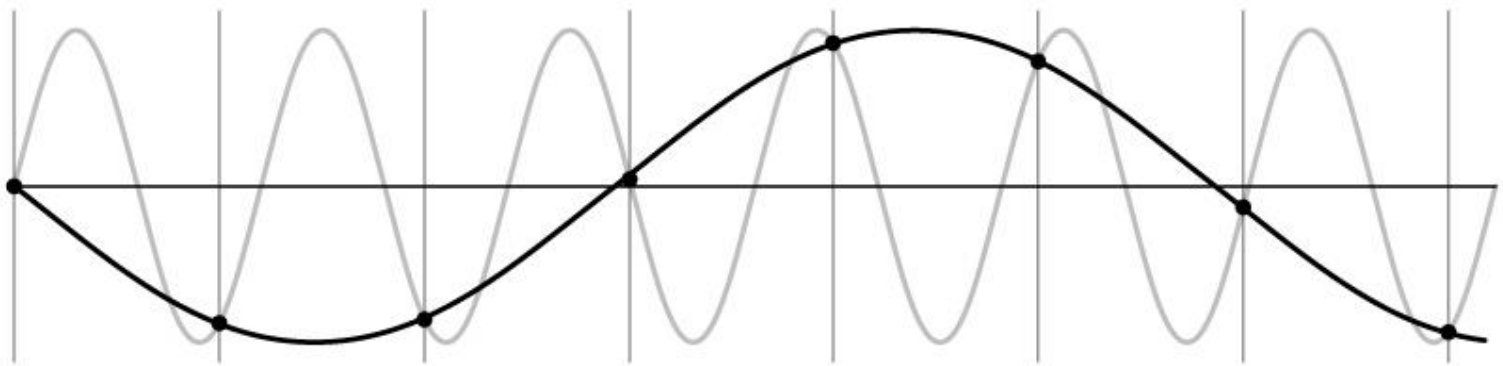
1/4 (2x subsample)

1/8 (4x subsample)

Aliasing

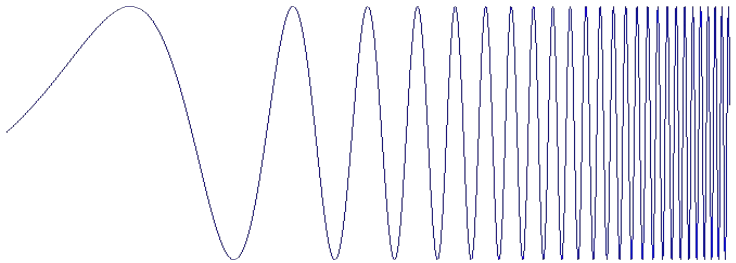


Aliasing



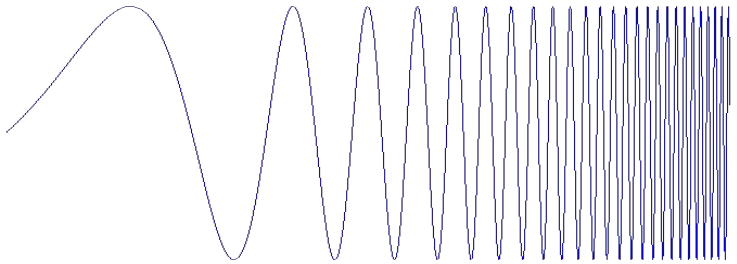
What's happening?

Input signal:

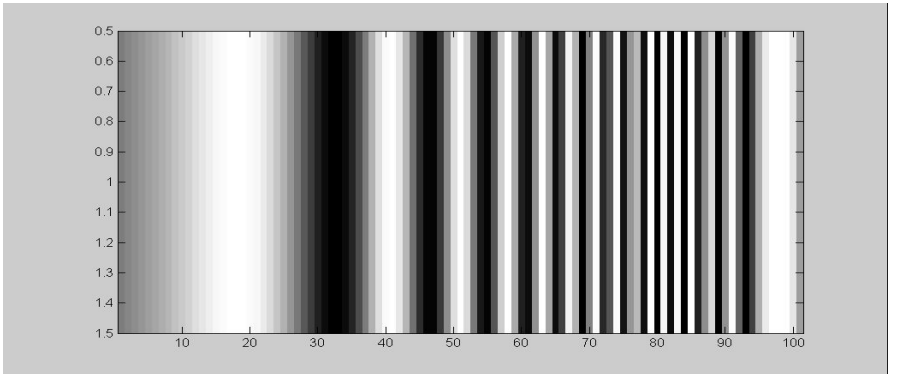


What's happening?

Input signal:



Plot as image:

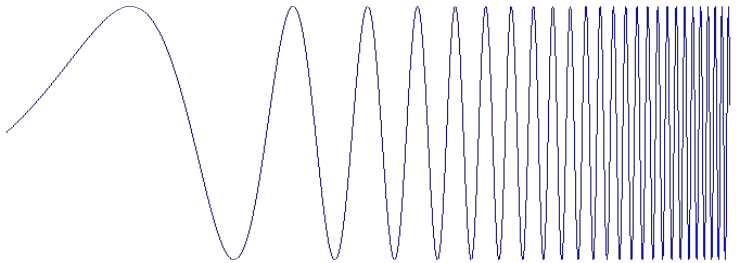



```
x = 0:.05:5;  plt.imshow(sin((2.^x).*x))
```

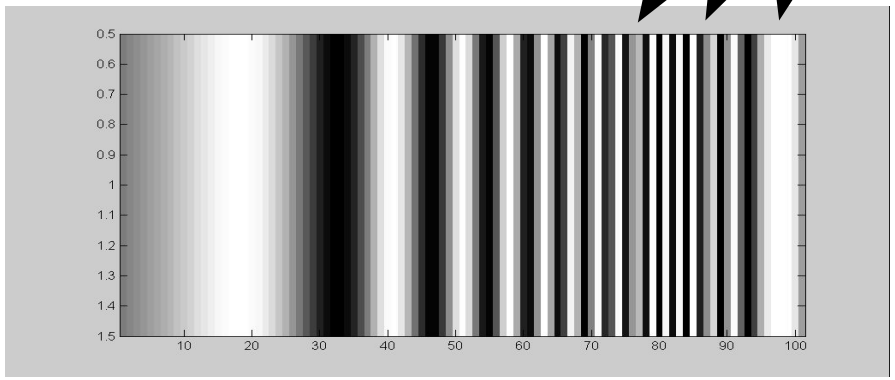
What's happening?

Aliasing!
Not enough
samples

Input signal:



Plot as image:

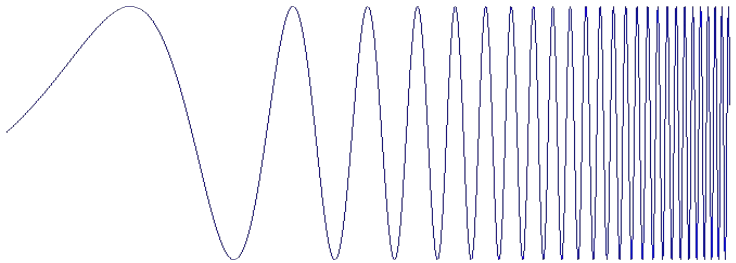



```
x = 0:.05:5;  plt.imshow(sin((2.^x).*x))
```

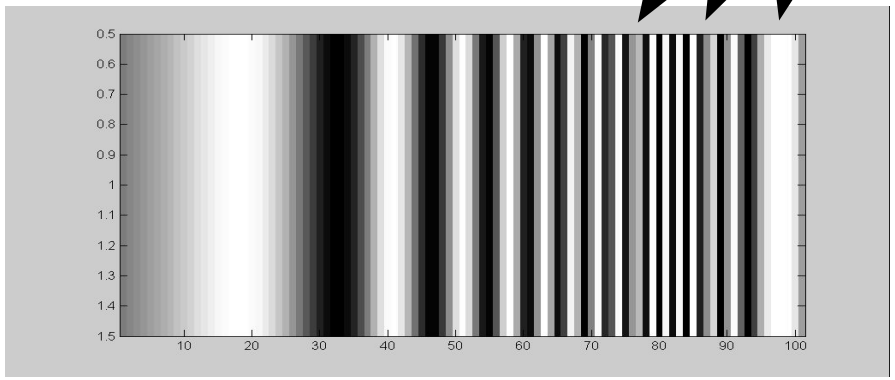
What's happening?

Aliasing!
Not enough
samples

Input signal:



Plot as image:




```
x = 0:.05:5;  plt.imshow(sin((2.^x).*x))
```

Aliasing:

- When two signals become indistinguishable from one another due to sampling.
- They are 'aliases' of one another.

Temporal Aliasing in Videos

[YouTube; JoinBuzzirk; phrancque]

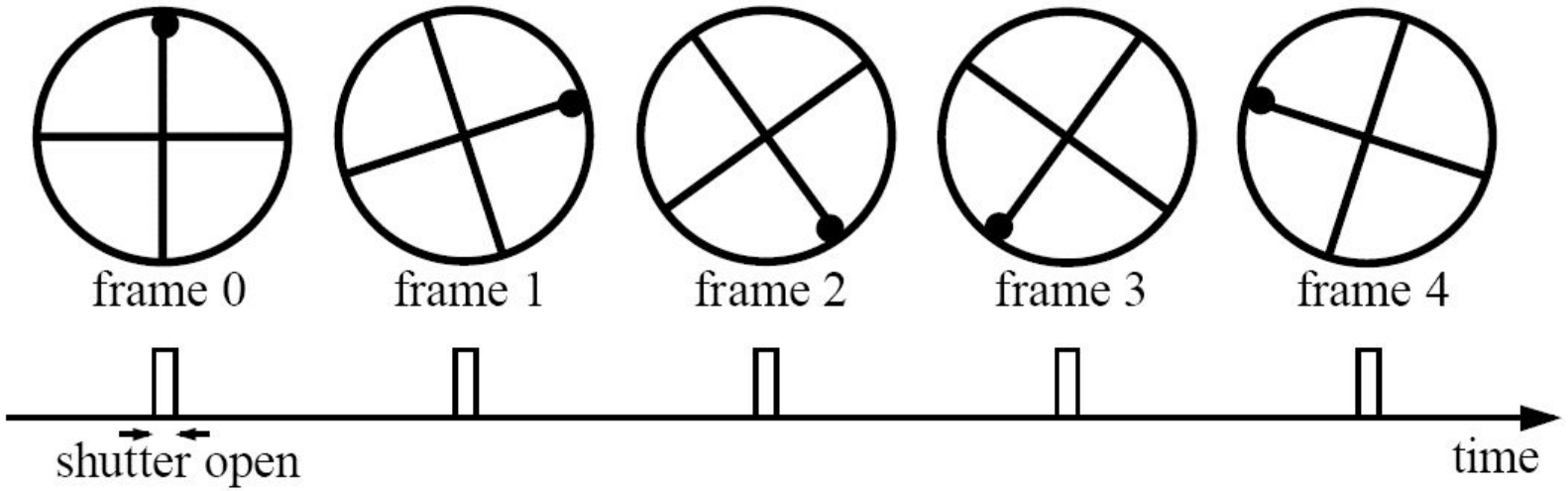
Temporal Aliasing in Videos

[YouTube; JoinBuzzirk; phrancque]

Aliasing in video

Imagine a spoked wheel moving to the right (rotating clockwise). Mark wheel with dot so we can see what's happening.

If camera shutter is only open for a fraction of a frame time (frame time = 1/30 sec. for video, 1/24 sec. for film):



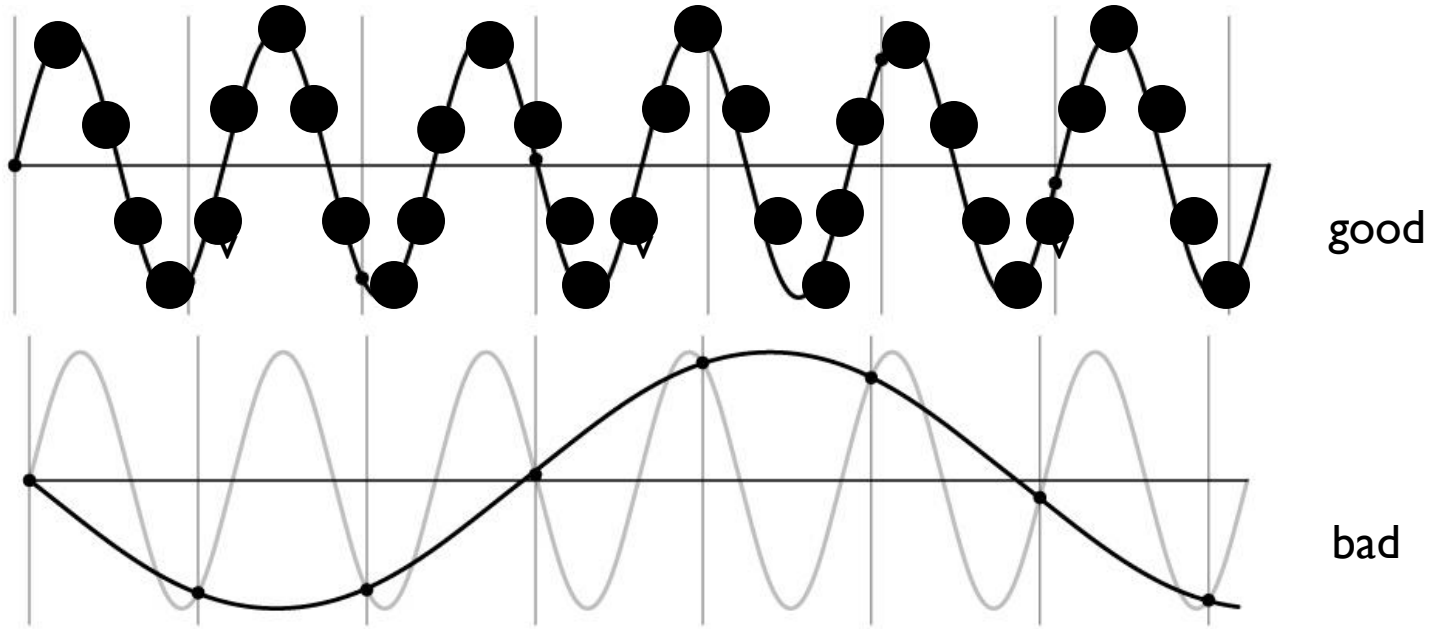
Without dot, wheel appears to be rotating slowly backwards! (counterclockwise)

Nyquist-Shannon Sampling Theorem

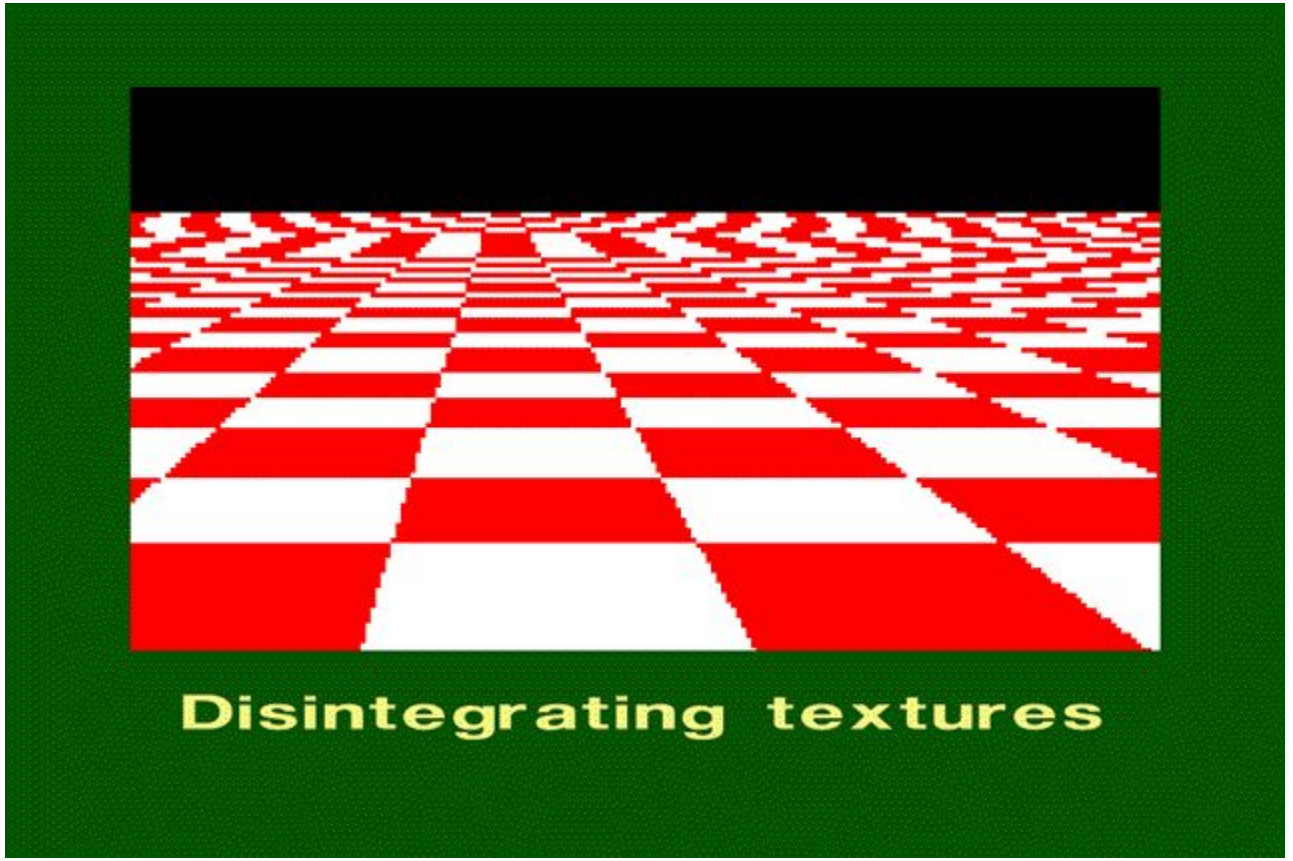
When sampling a signal at discrete intervals,

the sampling frequency must be $\geq 2 \times f_{max}$ max frequency of signal

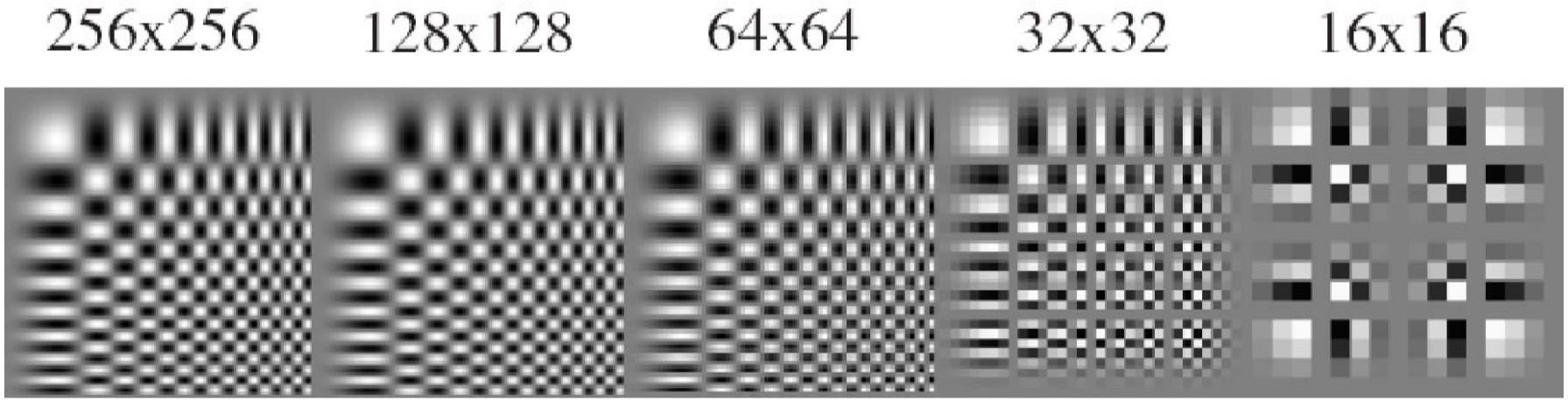
Signal is completely determined if so.



Aliasing in Graphics



Sampling and aliasing



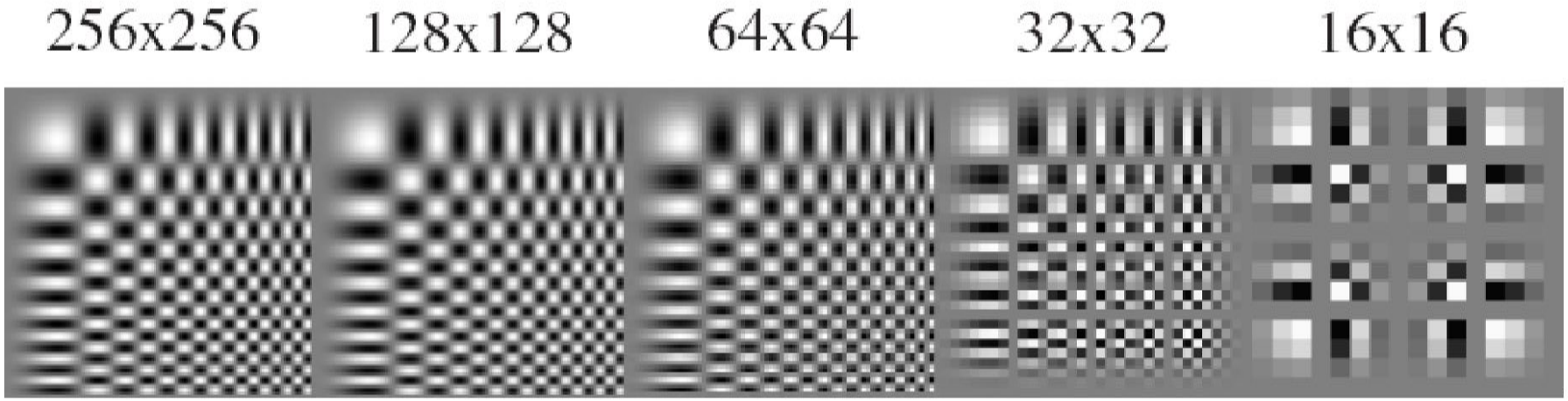
Anti-aliasing

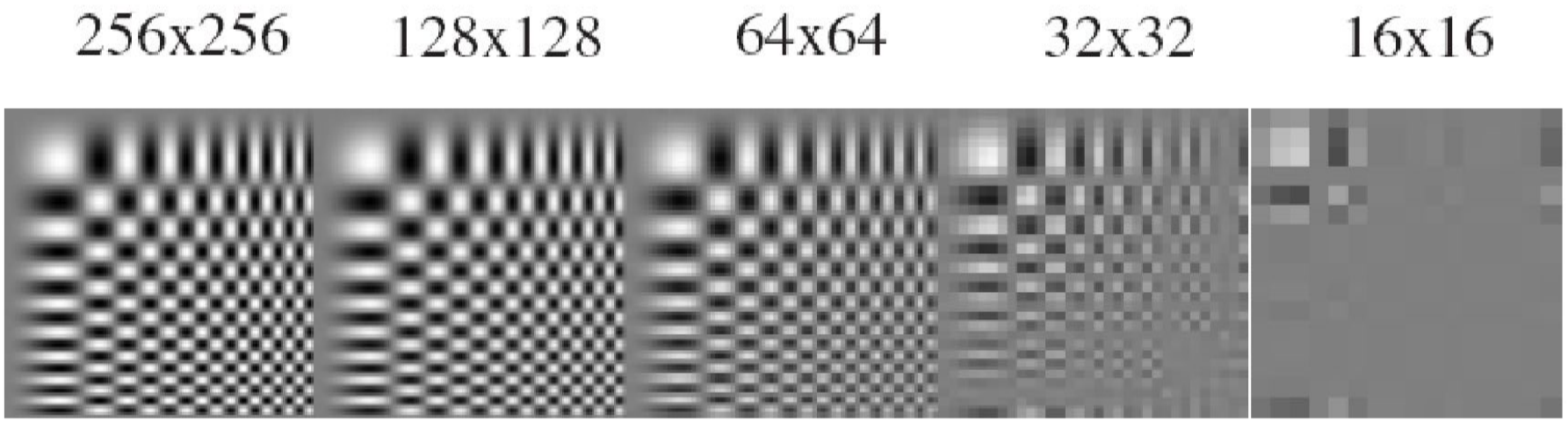
Solutions:

- Sample more often

- Remove all frequencies that are greater than half the new sampling frequency
  - Will lose information
  - But it's better than aliasing
  - How?

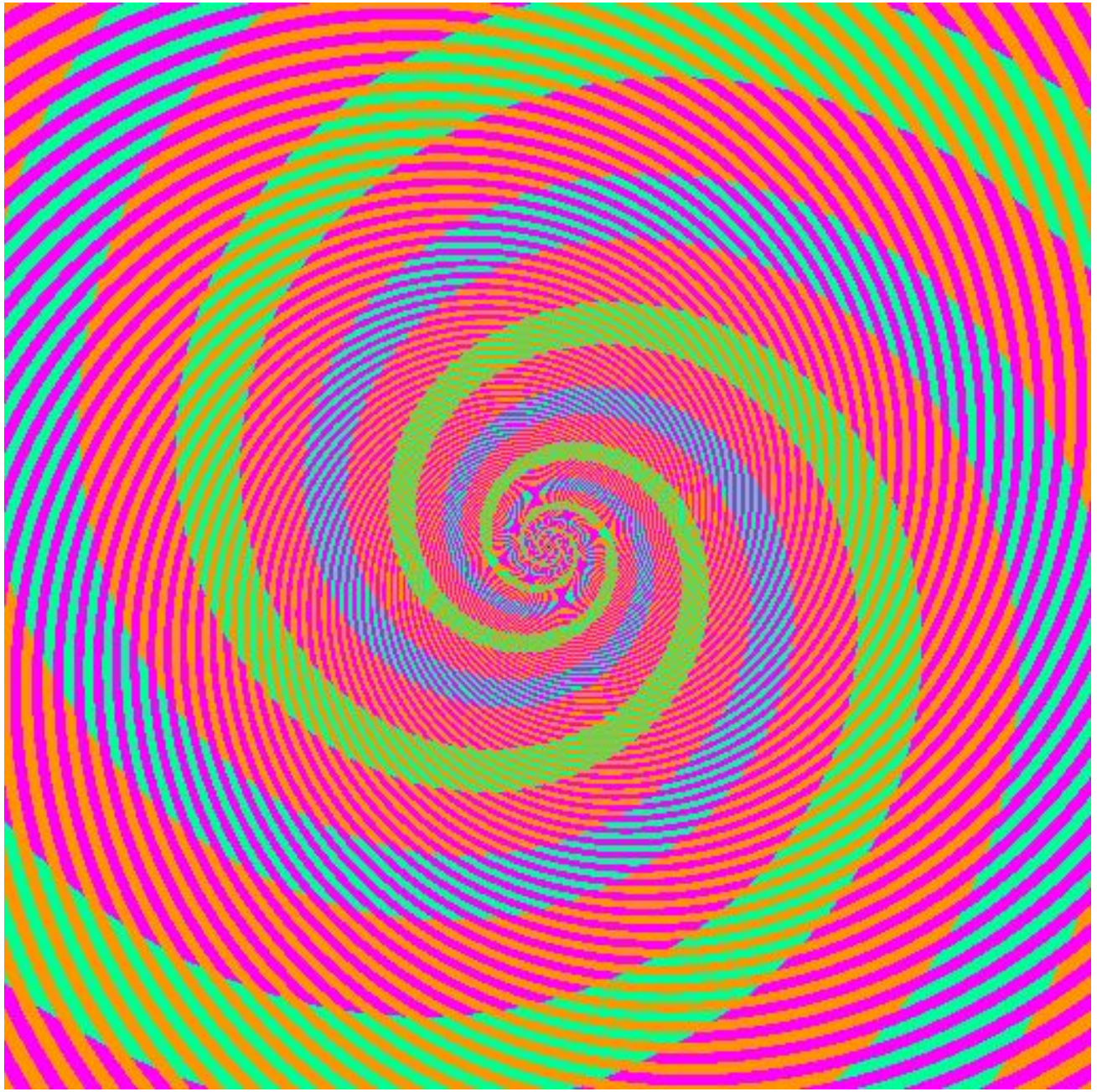
Anti-aliasing



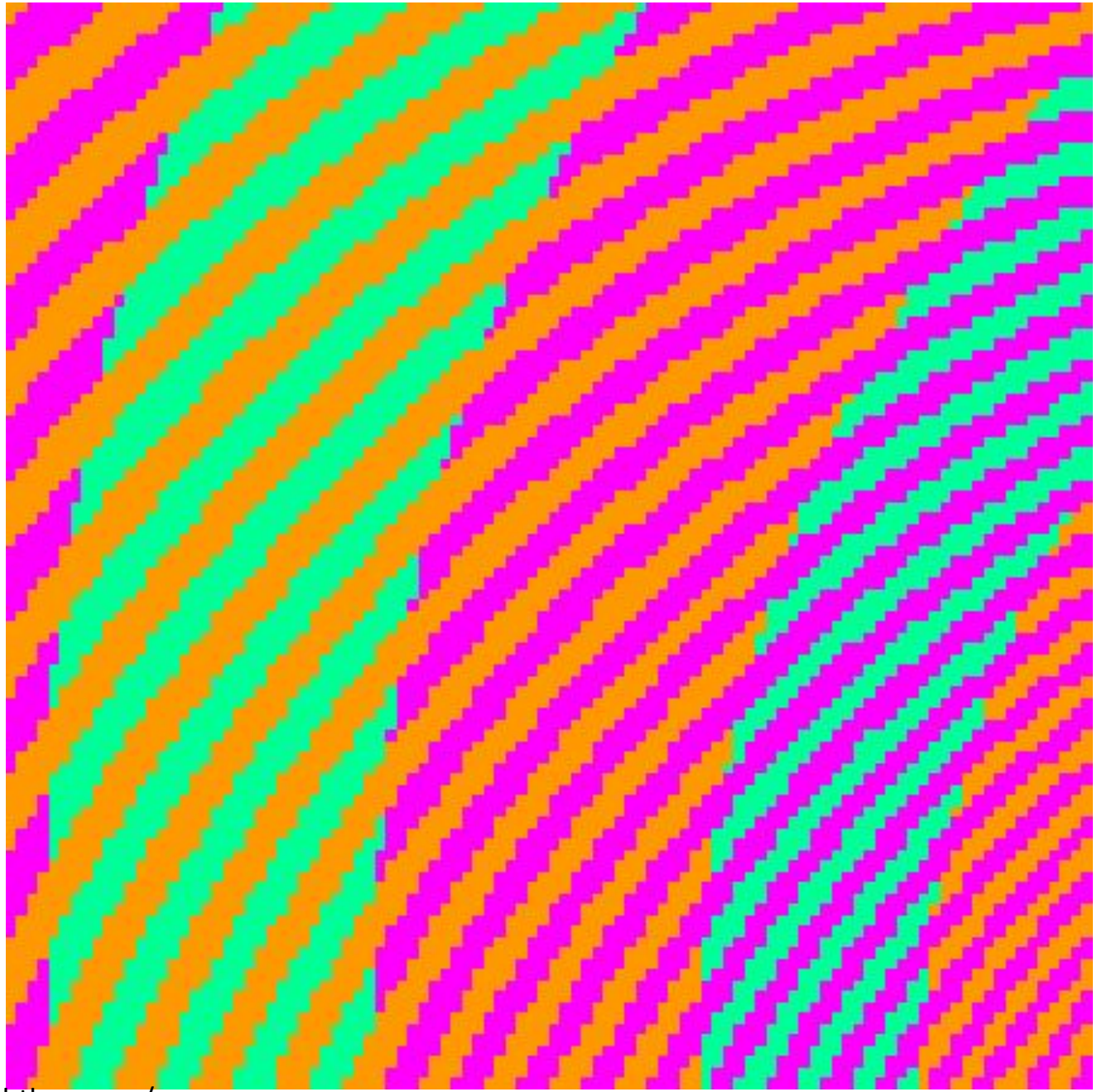


Forsyth and Ponce 2002

Aliasing and Moiré patterns



Aliasing and Moiré patterns



The blue and green colors are the same

http://blogs.discovermagazine.com/badastronomy/2009/06/24/the-blue-and-the-green/

Low-pass and high-pass filtering

Remove high frequencies with a *low pass* filter

- A blur filter!
- *Could also be called a 'high cut' filter (but few do)*

Counterpart: *high pass* filter

- Only keeps the high frequencies
- Edge detection (Sobel) is an example high pass filter

Related: *band pass* / *band stop* (or *notch)* filter

- Only affect *specific* frequencies
- How could we construct one with linear filters?

Algorithm for downsampling by factor of 2

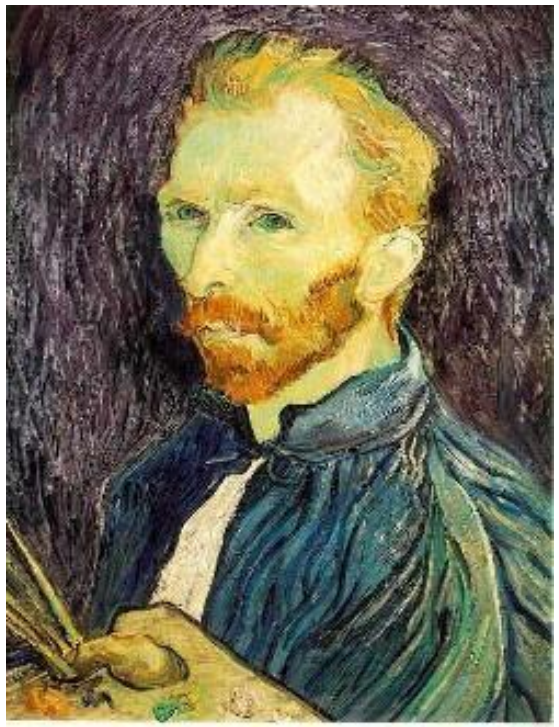
1. Start with image(h, w)
2. Apply low-pass filter

   im_blur = imfilter(image, fspecial('gaussian', 7, 1))

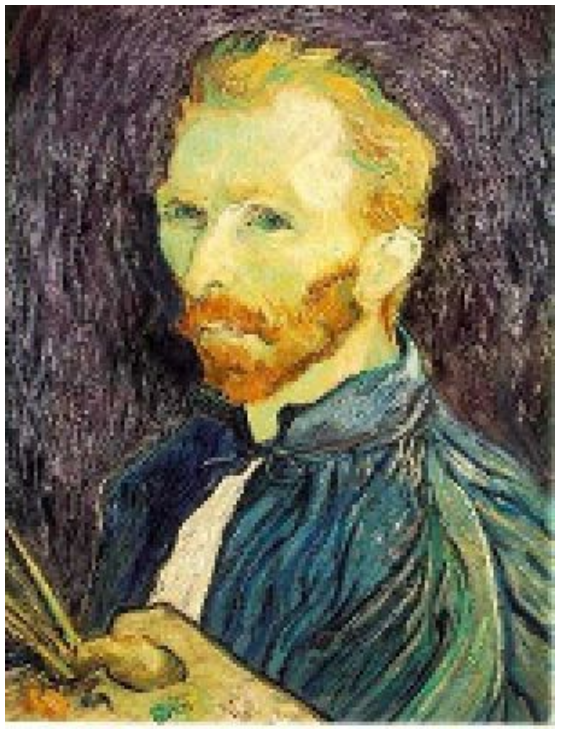
1. Sample every other pixel

   im_small = im_blur(1:2:end, 1:2:end);

Subsampling without Filtering



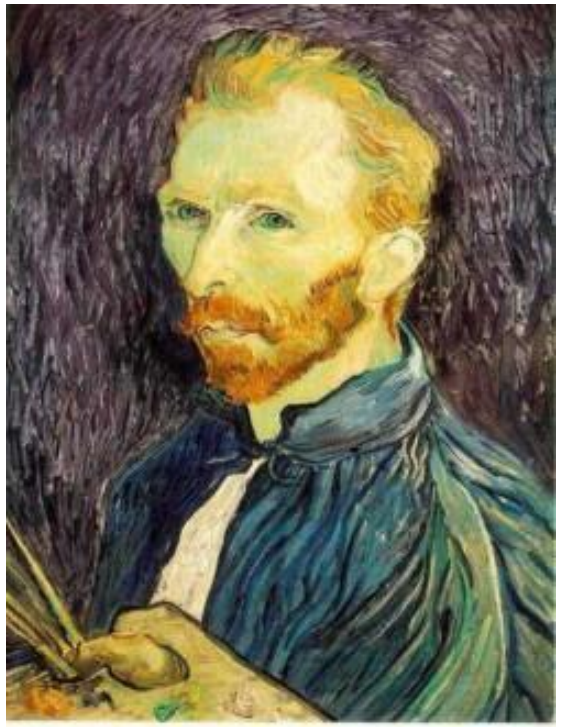
1/2



1/4 (2x subsample)

1/8 (4x subsample)

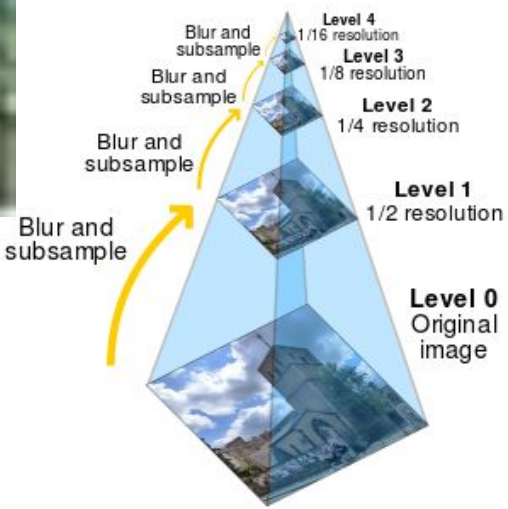
Subsampling with Gaussian Pre-filtering



Gaussian 1/2

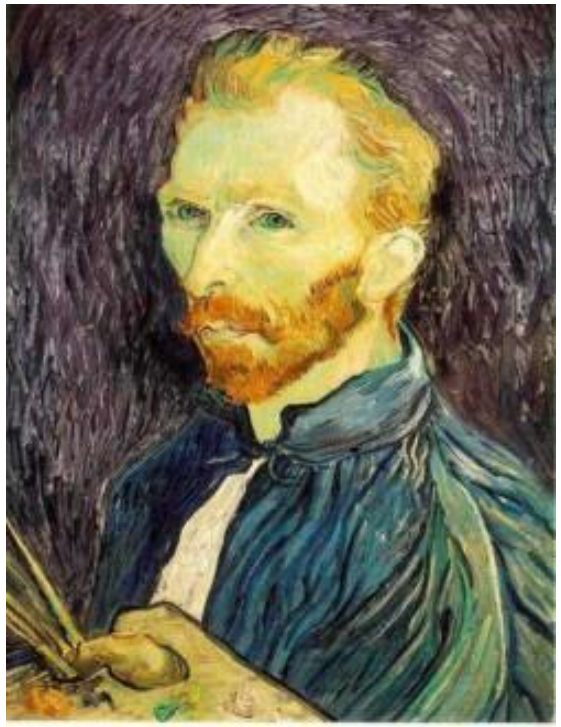
G 1/4

G 1/8



Gaussian Pyramid. Burt and Adelson, 1983

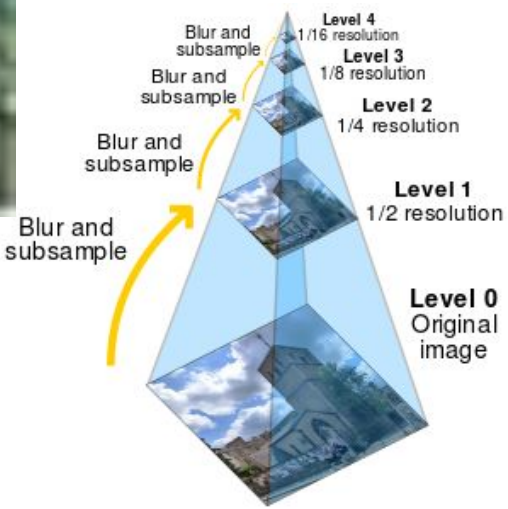
Subsampling with Gaussian Pre-filtering



Gaussian 1/2

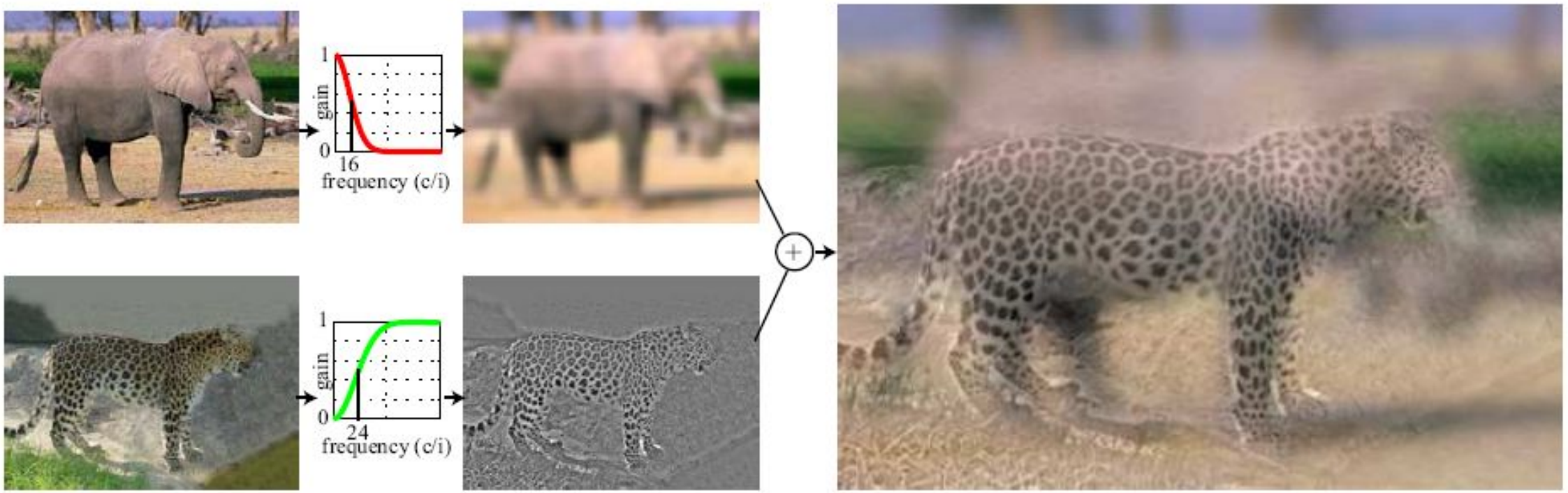
G 1/4

G 1/8



Gaussian Pyramid. Burt and Adelson, 1983

Hybrid Images



Hybrid Images. A. Oliva, A. Torralba, P.G. Schyns

CS 322: Image Processing and Computer Vision

Next episode...

- Frequency Domain
- Hybrid Images
- Fourier Transforms
- FT as filtering

Image Credit: Wallpapercrafter.com