CSCI 497P/597P: Computer Vision

Lecture 7:
Upsampling
Project 1 Overview
More numpy

Announcements

Goals

- Know how to upsample images naively

- Know how to upsample images using reconstruction filters.

  - Understand how to upsample using bilinear interpolation and how it relates to reconstruction filters.

- Get exposed to a bit more numpy

Reconstruction

L_0

L_1

L_2

L_3

L_4

L_5


```
levels = reverse(levels)
img = levels[0]
for i in 1..len(levels):
  img = upscale_2x(img)
  img += L_i
```

Upsampling

- But how do we make images bigger?

- Again: a naive way and a principled way.

```
levels = reverse(levels)
img = levels[0]
for i in 1..len(levels):
  img = upscale_2x(img)
  img += L_i
```

Upsampling

- This image is too small for my screen. How do I make it 10x bigger?

Upsampling

- This image is too small for my screen. How do I make it 10x bigger?
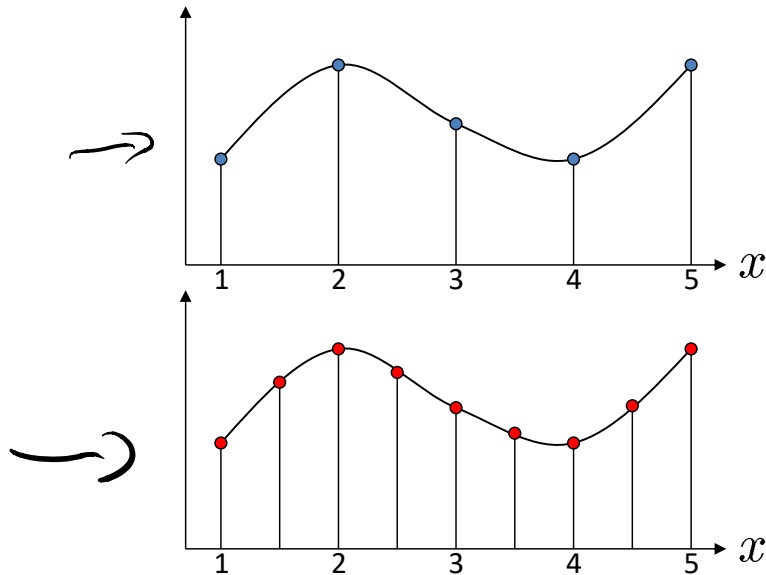

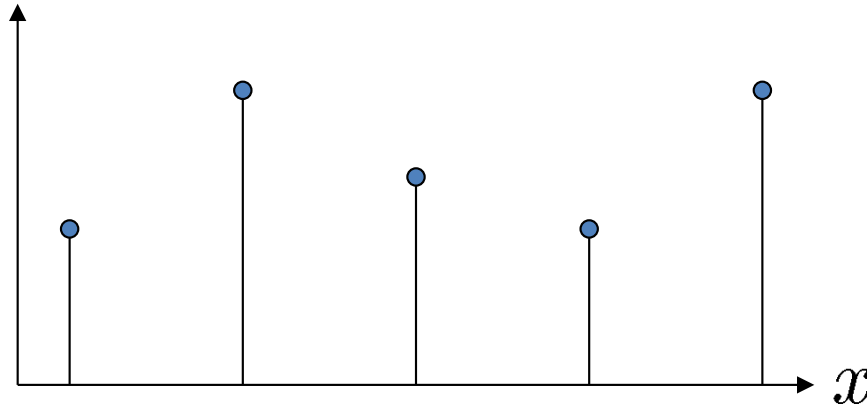- Simple approach: repeat each row and column 10 times

Upsampling: Interpolation

- Another way to look at this: we need to double the *sampling rate*.
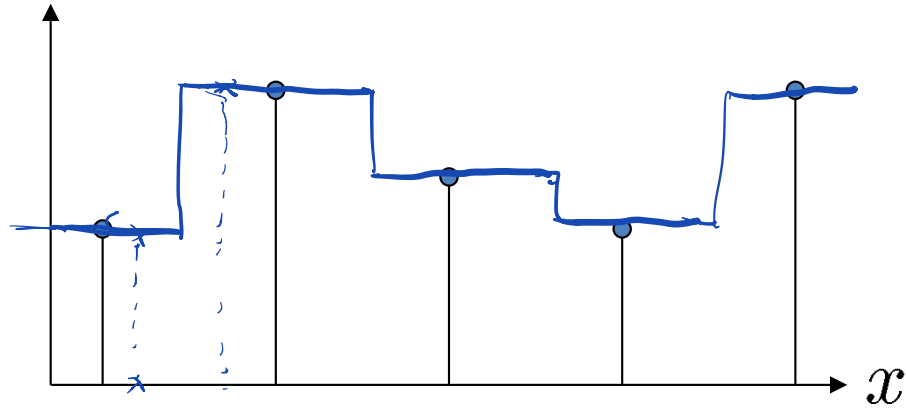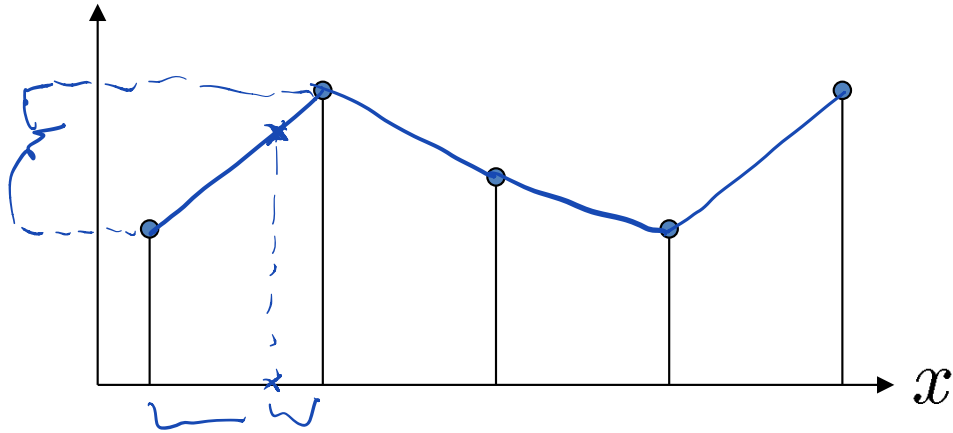
Upsampling: Interpolation

- Another way to look at this: we need to double the *sampling rate*.

- But we don't actually know the continuous function:
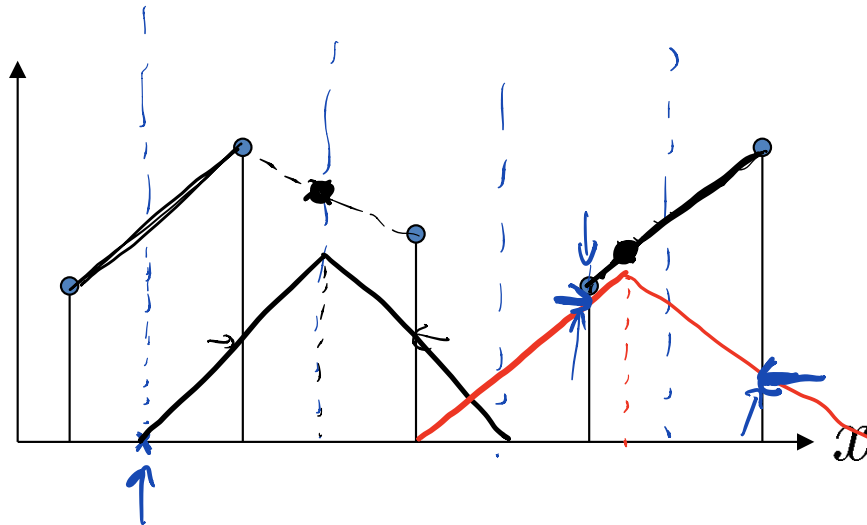
Upsampling: Nearest Neighbor
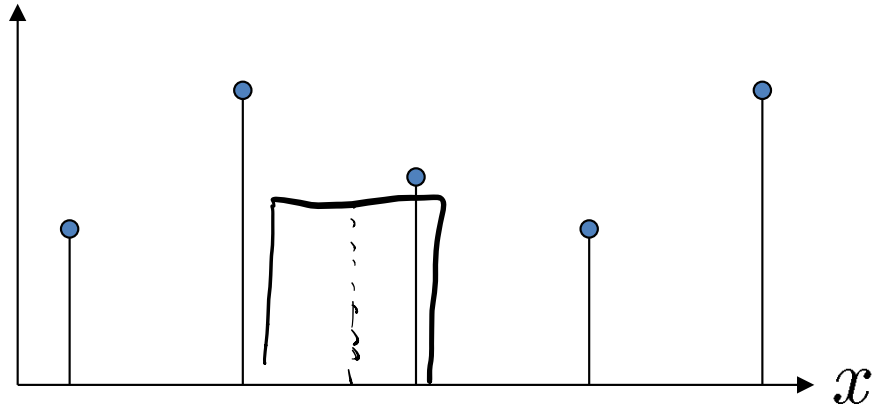
Upsampling: Linear
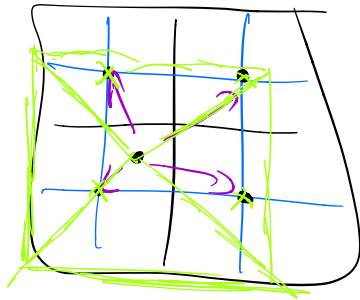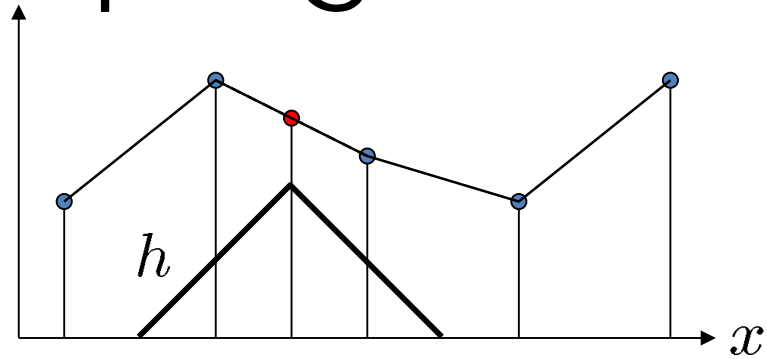
Upsampling: Linear

A filtering perspective

Upsampling: Nearest Neighbor

A filtering perspective

Upsampling Filters in 2D

h

x

1	2	1
2	4	2
1	2	1

1D: h

2D:
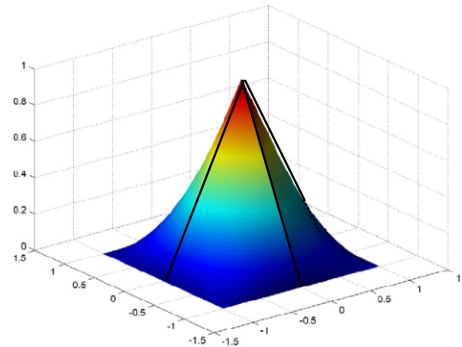
"tent filter"

Upsampling by 4X

1. Make 4Hx4W image of zeros.
2. Fill in every 4th pixel
3. Filter*!
   *and multiply by 16

Project 1: Part 1

High pass

Low pass

+

Project 1: Part 1

=

Project 1: Part 1

Project 1: Part 2

- Demo

A little more numpy...

- Once more to the playground!