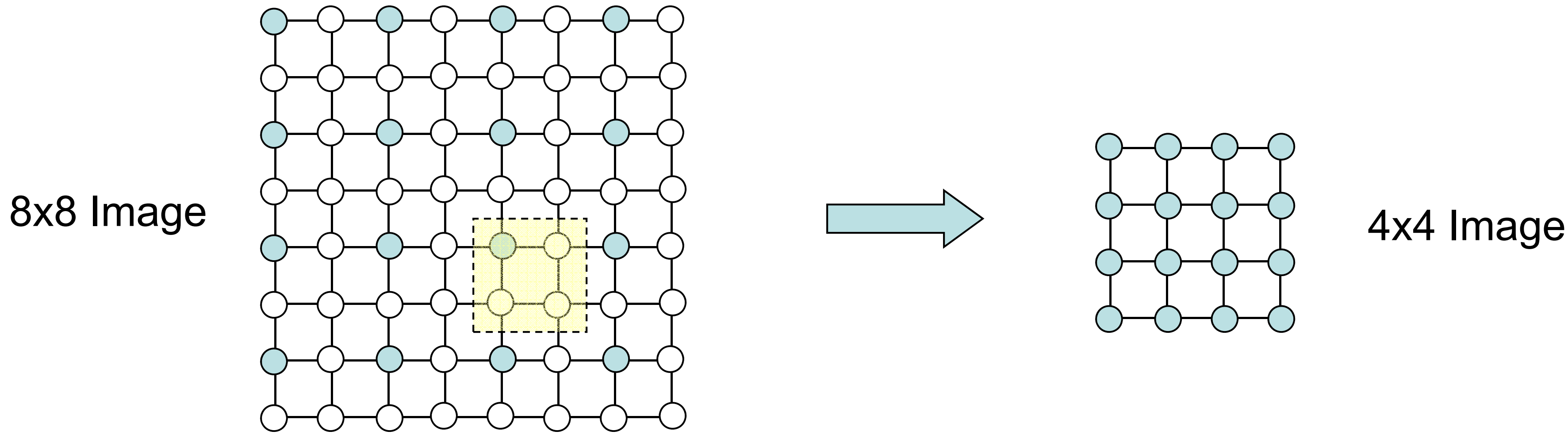
Image Resizing

- Image resizing:
  - Enlarge or reduce the image size (number of pixels)
  - Equivalent to
    - First reconstruct the continuous image from samples
    - Then Resample the image at a different sampling rate
  - Can be done w/o reconstructing the continuous image explicitly
- Image down-sampling (resample at a lower rate)
  - Spatial domain view
  - Frequency domain view: need for prefilter
- Image up-sampling (resample at a higher rate)
  - Spatial domain view
  - Different interpolation filters
    - Nearest neighbor, Bilinear, Bicubic

Image Down-Sampling

- Example:
  - reduce a 512x512 image to 256x256 = factor of 2 downsampling in both horizontal and vertical directions
  - In general, we can down-sample by an arbitrary factor in the horizontal and vertical directions
- How should we obtain the smaller image ?

Down Sampling by a Factor of Two



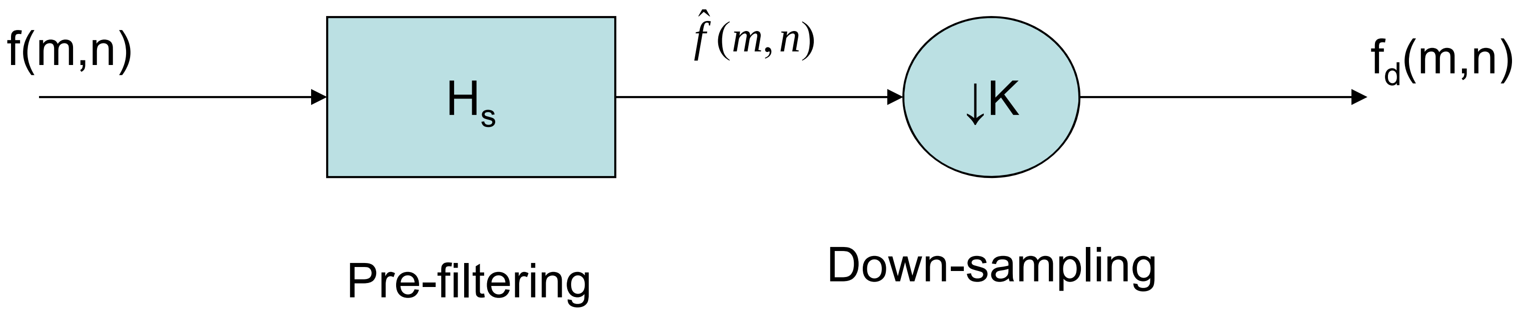
- Without Pre-filtering (simple approach)

$$f_d(m,n) = f(2m,2n)$$

- Averaging Filter

$$f_d(m,n) = [f(2m,2n) + f(2m,2n+1) + f(2m+1,2n) + f(2m+1,2n+1)]/4$$

Down Sampling by a Factor of K



$$f_d(m,n) = \hat{f}(Km, Kn)$$

For factor of K down sampling, the prefilter should be low pass filter with cutoff at fs/(2K), if fs is the original sampling frequency

In terms of digital frequency, the cutoff should be 1/(2K)

Example: Image Down-Sample

Without prefiltering

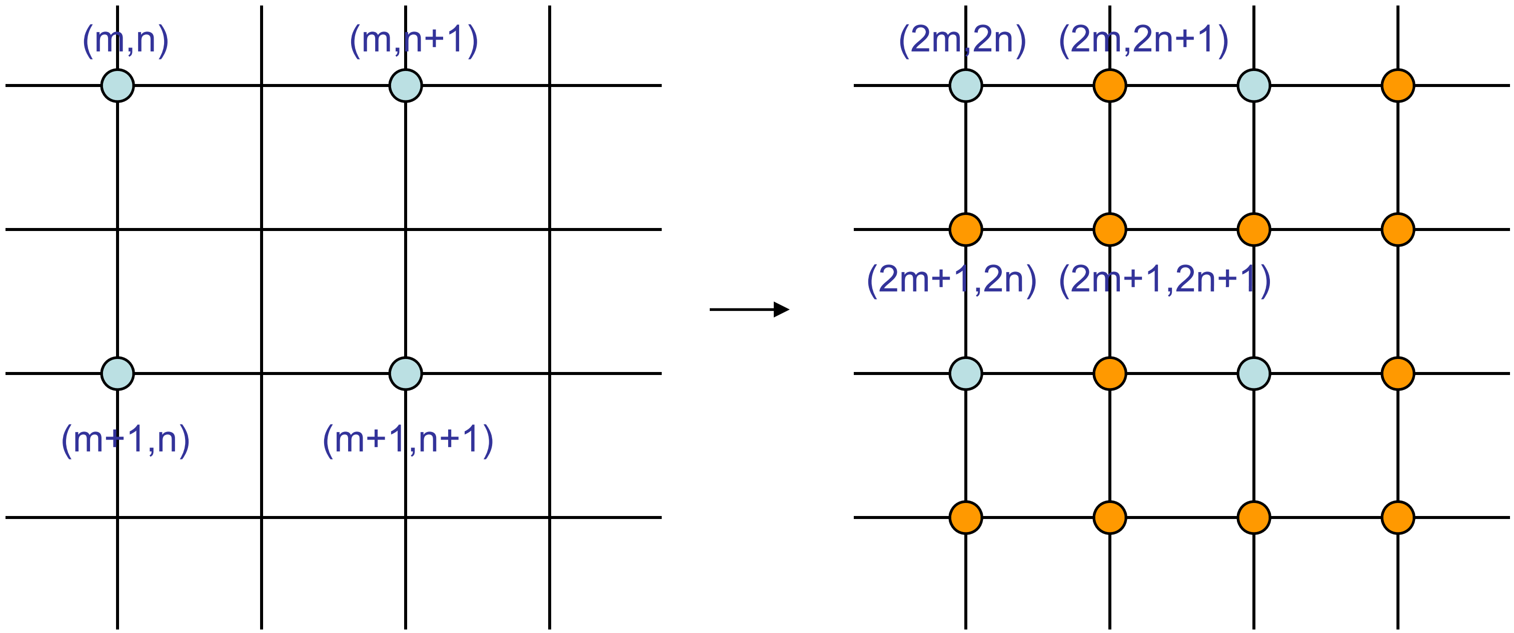
With prefiltering (no aliasing, but blurring!)

Image Up-Sampling

- Produce a larger image from a smaller one
  - Eg. 512x512 -> 1024x1024
  - More generally we may up-sample by an arbitrary factor L
- Questions:
  - How should we generate a larger image?
  - Does the enlarged image carry more information?
- Connection with interpolation of a continuous image from discrete image
  - First interpolate to continuous image, then sampling at a higher sampling rate, L*fs
  - Can be realized with the same interpolation filter, but only evaluate at $x=m\Delta x'$, $y=n\Delta y'$, $\Delta x'=\Delta x/L$, $\Delta y'=\Delta y/L$
  - Ideally using the sinc filter!

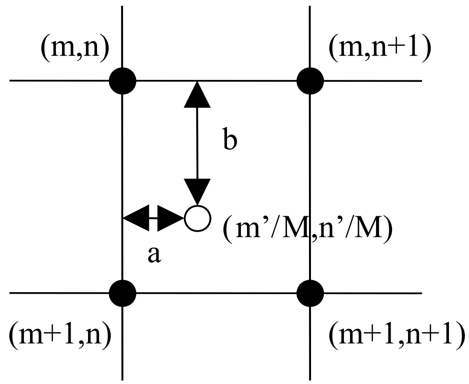
$$\hat{f}(x,y)=\sum_m \sum_n f_s(m,n)\frac{\sin \pi f_{s,x}(x-m\Delta x)}{\pi f_{s,x}(x-m\Delta x)}\frac{\sin \pi f_{s,y}(y-m\Delta y)}{\pi f_{s,y}(y-m\Delta y)}$$

Example: Factor of 2 Up-Sampling



Green samples are retained in the interpolated image;
Orange samples are estimated from surrounding green samples.

Nearest Neighbor Interpolation (pixel replication)



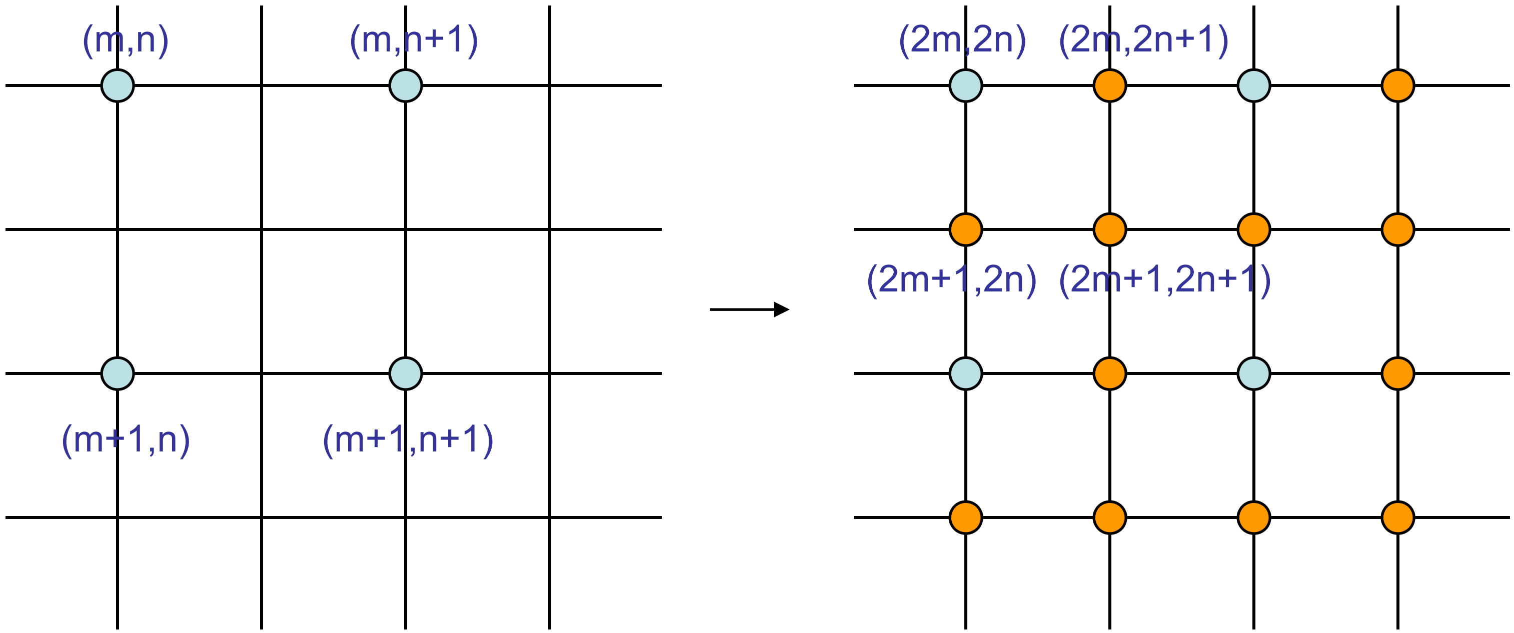
O[m',n'] (the resized image) takes the value of the sample nearest to (m'/M,n'/M) in I[m,n] (the original image):

$$O[m',n'] = I[(\text{int})\,(m+0.5), (\text{int})\,(n+0.5)],\ m = m'/M,\ n = n'/M.$$

Also known as pixel replication: each original pixel is replaced by MxM pixels of the sample value

Equivalent to using the sample-and-hold interpolation filter.

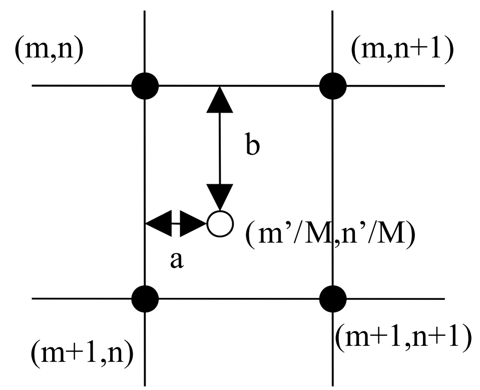
Special Case: M=2



Nearest Neighbor:

O[2m,2n]=I[m,n]
O[2m,2n+1]= I[m,n]
O[2m+1,2n]= I[m,n]
O[2m+1,2n+1]= I[m,n]

Bilinear Interpolation



- O(m',n') takes a weighted average of 4 samples nearest to (m'/M,n'/M) in I(m,n).

- Direct interpolation: each new sample takes 4 multiplications:

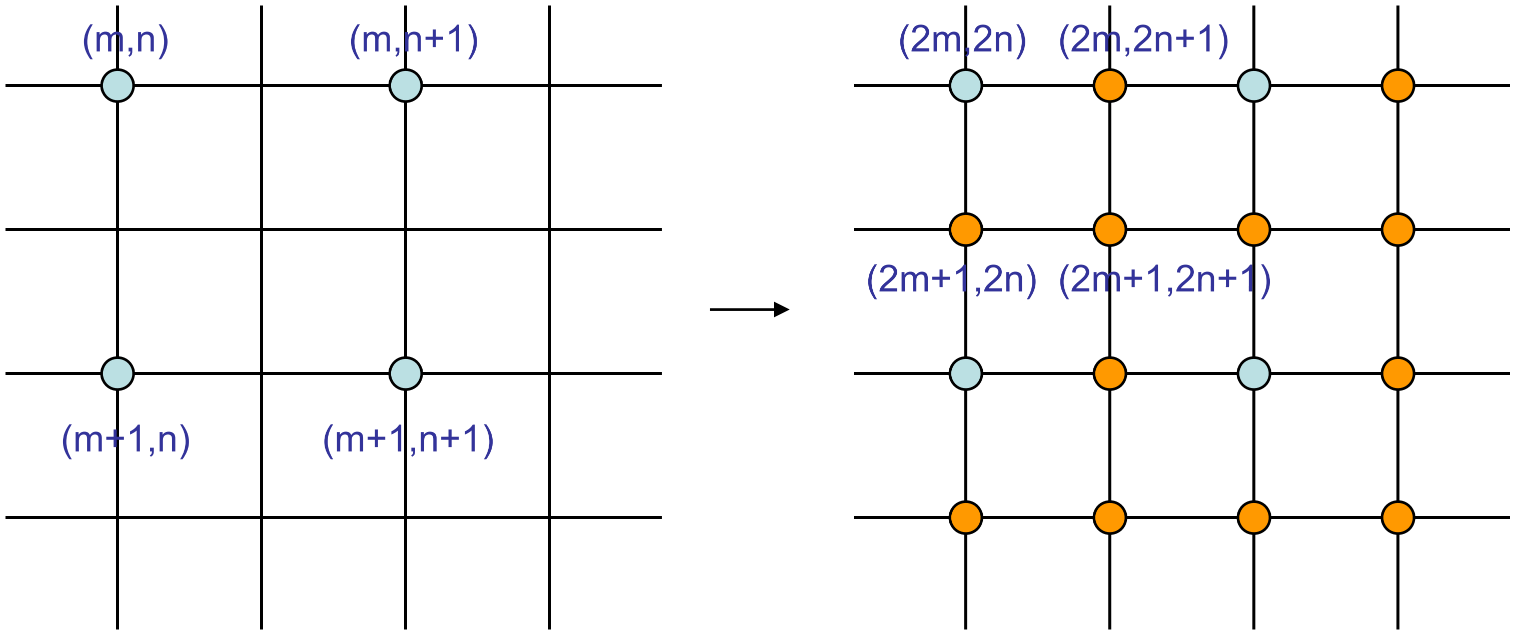
  O[m',n']=(1-a)*(1-b)*I[m,n]+a*(1-b)*I[m,n+1]+(1-a)*b*I[m+1,n]+a*b*I[m+1,n+1]

- Separable interpolation:

  i) interpolate along each row y: F[m,n']=(1-a)*I[m,n]+a*I[m,n+1]

  ii) interpolate along each column x': O[m',n']=(1-b)*F[m',n]+b*F[m'+1,n]

Special Case: M=2



Bilinear Interpolation:

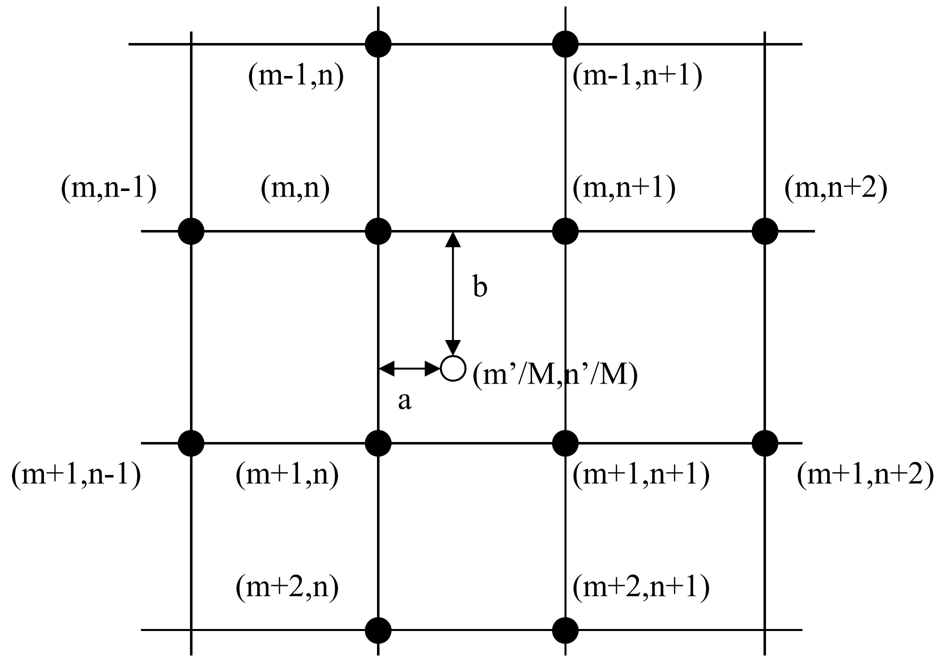
O[2m,2n]=I[m,n]

O[2m,2n+1]=(I[m,n]+I[m,n+1])/2

O[2m+1,2n]=(I[m,n]+I[m+1,n])/2

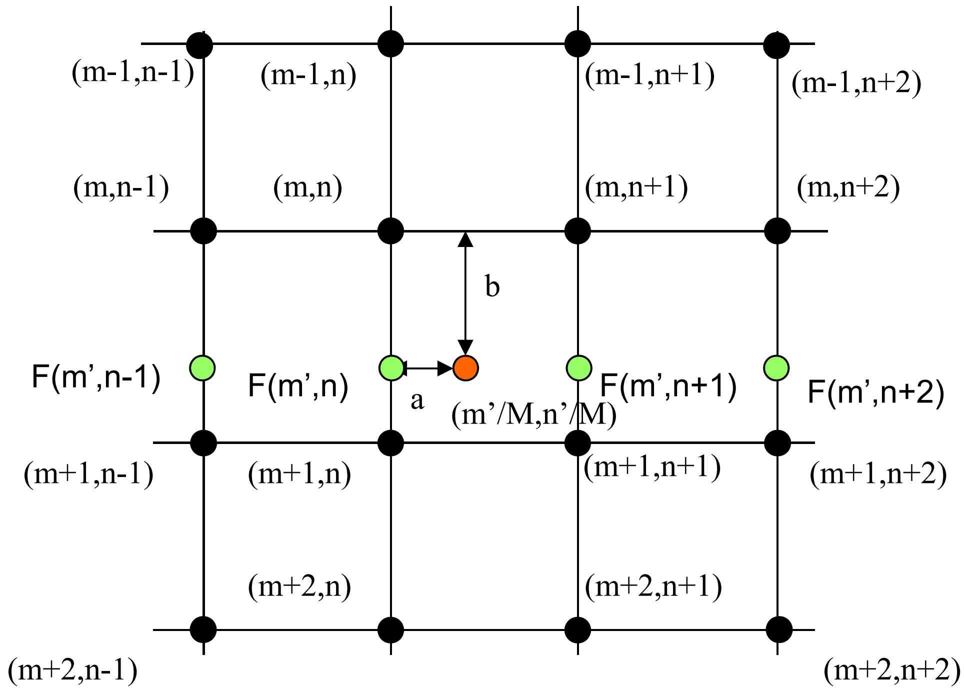
O[2m+1,2n+1]=(I[m,n]+I[m,n+1]+I[m+1,n]+I[m+1,n+1])/4

Bicubic Interpolation



- O(m',n') is interpolated from 16 samples nearest to (m'/M,n'/M) in I(m,n).
- Direct interpolation: each new sample takes 16 multiplications
- Separable interpolation:
  - i) interpolate along each row y: I[m,n]->F[m,n'] (from 4 samples)
  - ii) interpolate along each column x': F[m,n']-> O[m',n'] (from 4 samples)

Interpolation Formula



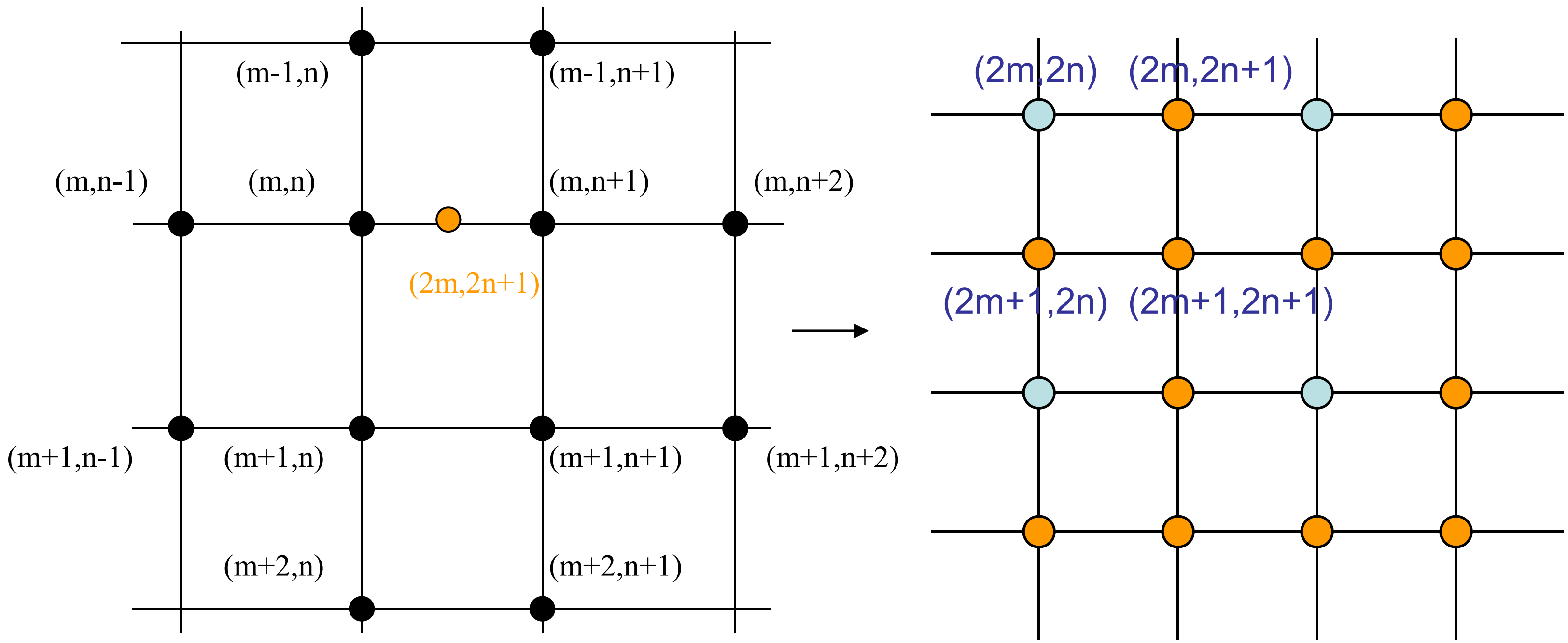
$$F[m',n] = -b(1-b)^2 I[m-1,n] + (1-2b^2+b^3)I[m,n] + b(1+b-b^2)I[m+1,n] - b^2(1-b)I[m+2,n],$$

$$where \quad m = (\text{int})\frac{m'}{M}, b = \frac{m'}{M} - m$$

$$O[m',n'] = -a(1-a)^2 F[m',n-1] + (1-2a^2+a^3)F[m',n] + a(1+a-a^2)F[m',n+1] - a^2(1-a)F[m',n+2],$$

$$where \quad n = (\text{int})\frac{n'}{M}, a = \frac{n'}{M} - n$$

Special Case: M=2



Bicubic interpolation in Horizontal direction

F[2m,2n]=I[m,n]
F[2m,2n+1]= -(1/8)I[m,n-1]+(5/8)I[m,n]+(5/8)I[m,n+1]-(1/8)I(m,n+2)

Same operation then repeats in vertical direction

Up-Sampled from w/o Prefiltering

Original

Nearest neighbor

Bilinear

Bicubic

Up-Sampled from with Prefiltering

Original

Nearest neighbor

Bilinear

Bicubic