Digital Image Processing

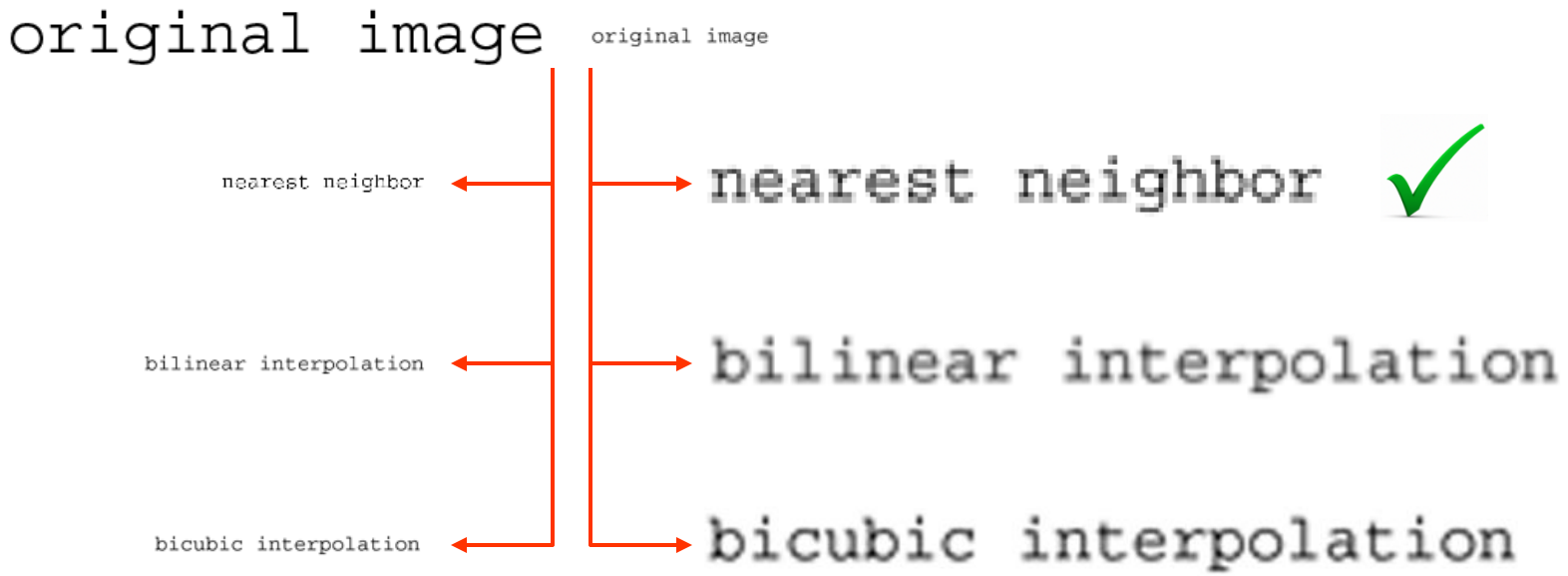
Lecture # 2B: Fundamentals

Contents

- Image Interpolation
  - Pixel Replication
  - Pixel Decimation
  - Nearest Neighbor Interpolation
  - Comparison of Bilinear & Nearest Neighbor Interpolation

Image Interpolation

- Image resizing
- Three methods



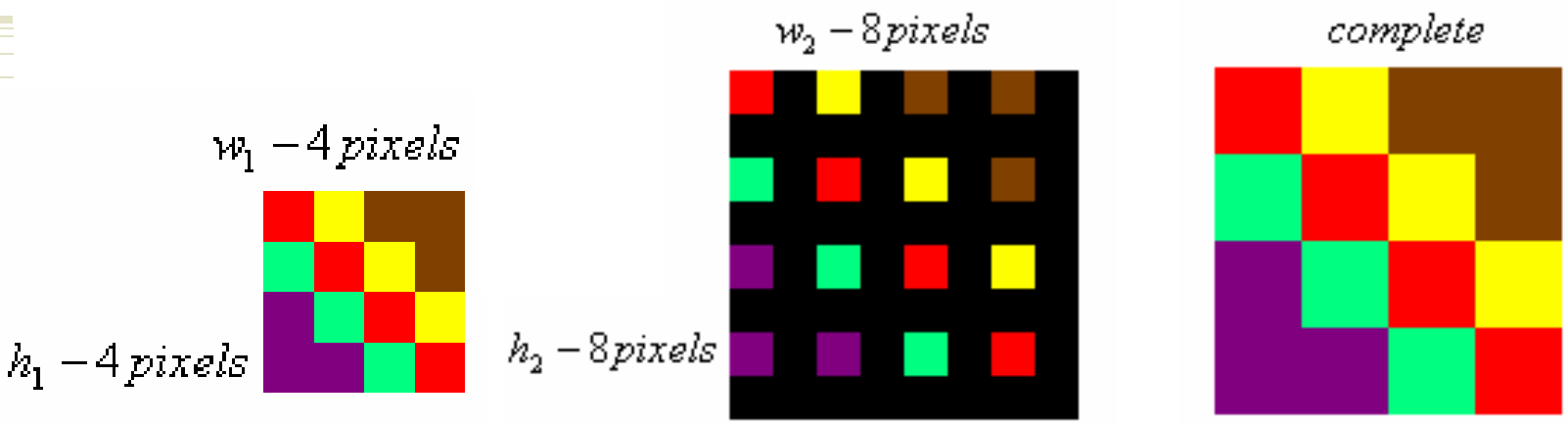
Enlarging an Image

- Pixel replication

[1 2 3 4 5]

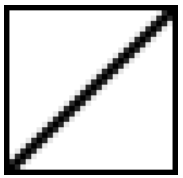
[1 1 2 2 3 3 4 4 5 5] (One step)

[1 1 1 2 2 2 3 3 3 4 4 4 5 5 5] (Two step)

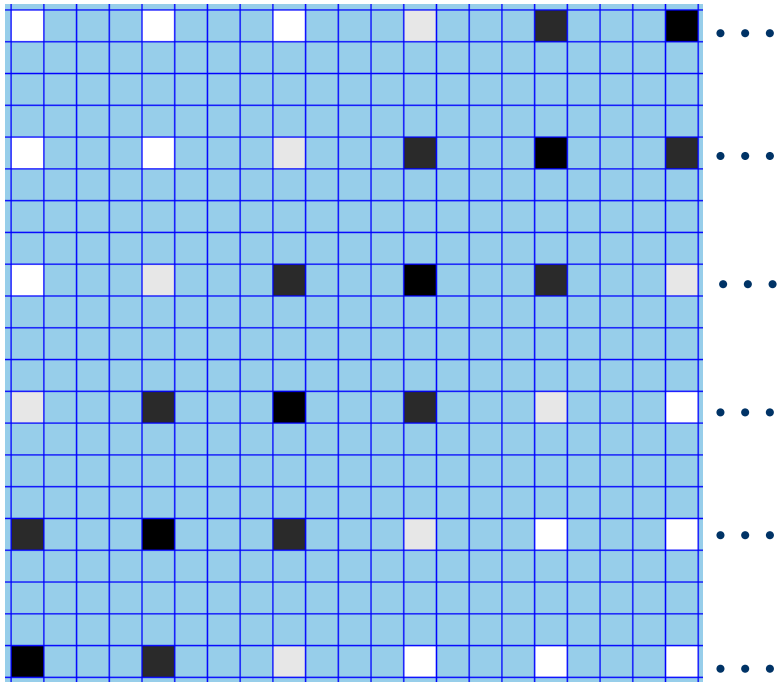


Enlarging an Image

Example: zoom this image 4x to get this image.

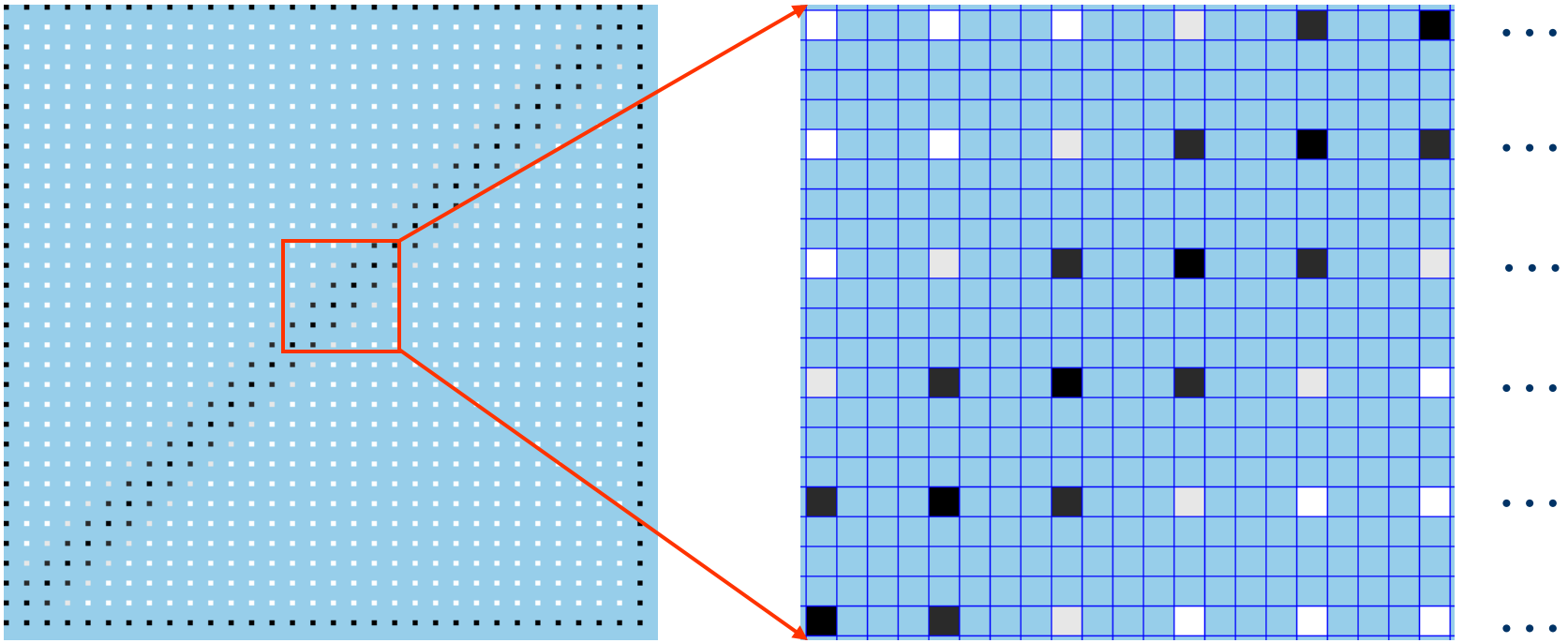


Start with a blank image 4 times the linear dimensions of the original.



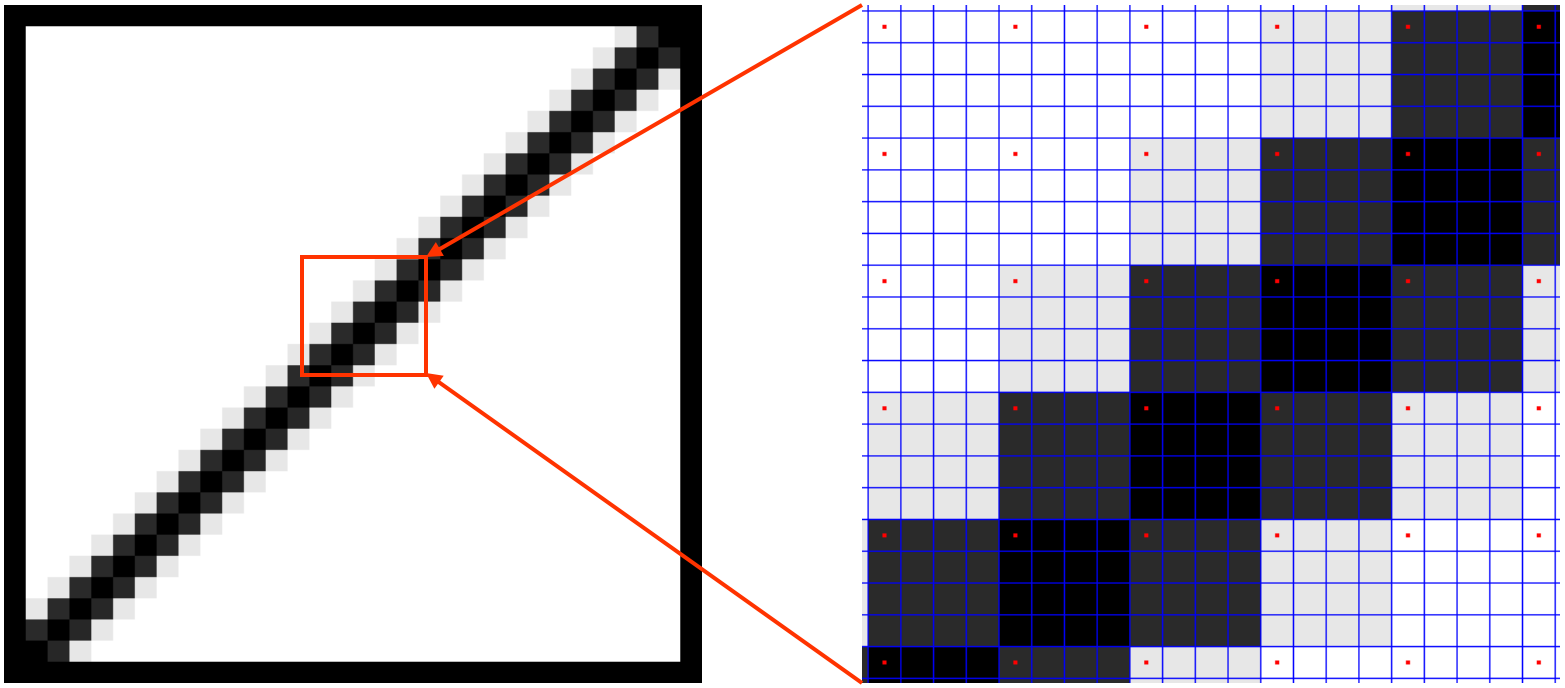
Fill in every 4th pixel in every 4th row with the original pixel values.

Enlarging an Image



Detail showing every 4th pixel in every 4th row with the original pixel values.

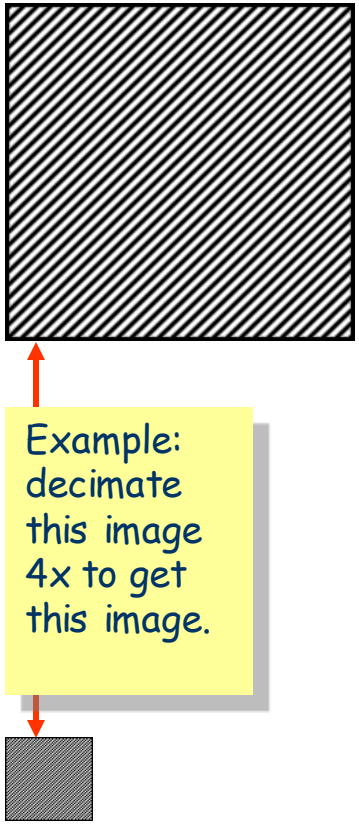
Enlarging an Image



Replicate the values

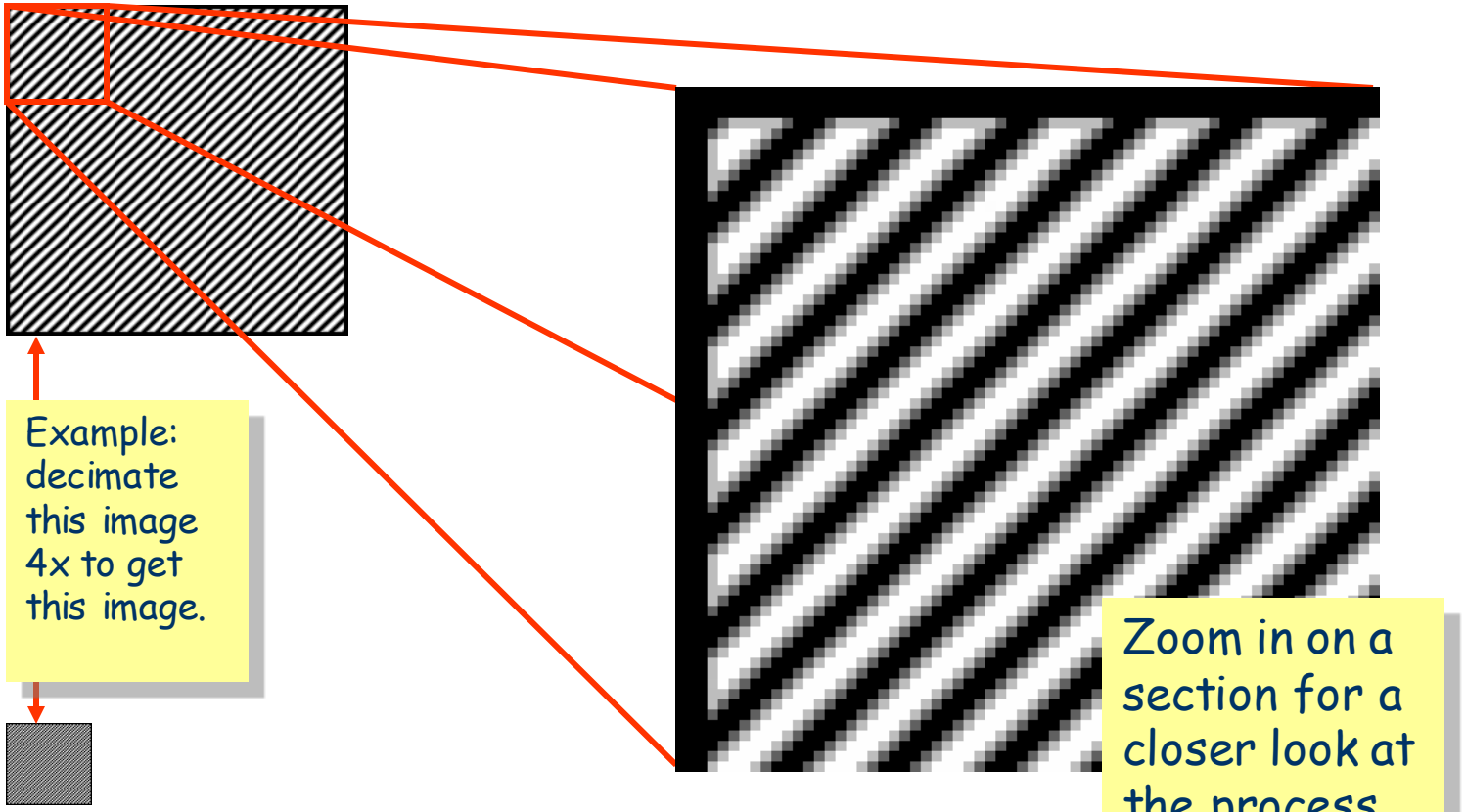
Reducing an Image

- Pixel Decimation

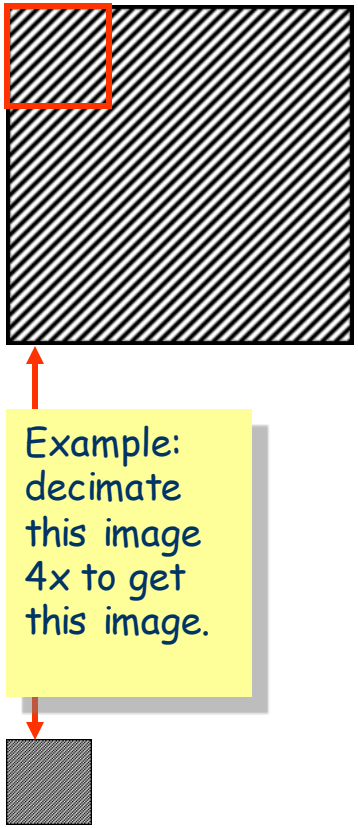


Decimation by a factor of *n*: take every *n*th pixel in every *n*th row

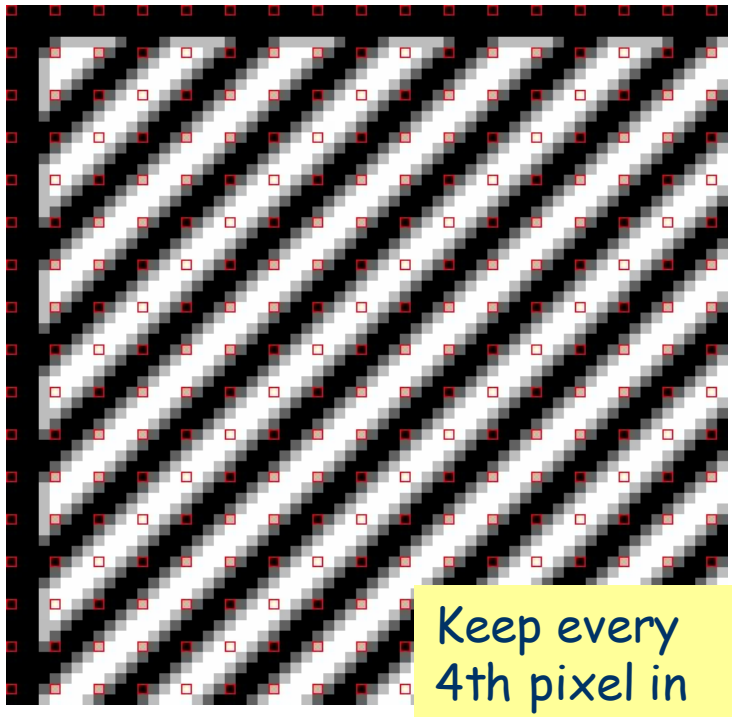
Reducing an Image



Reducing an Image

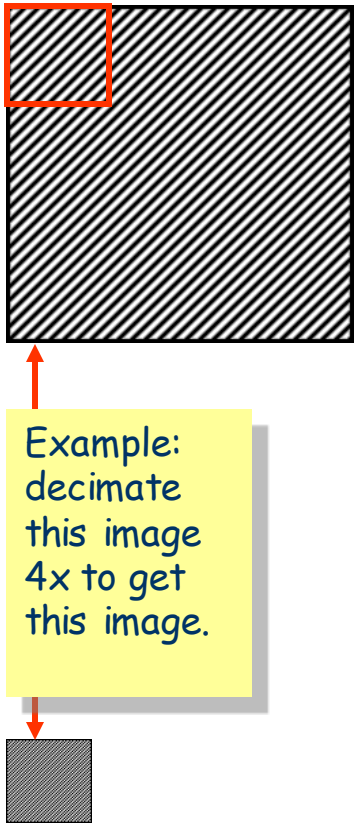


Example: decimate this image 4x to get this image.

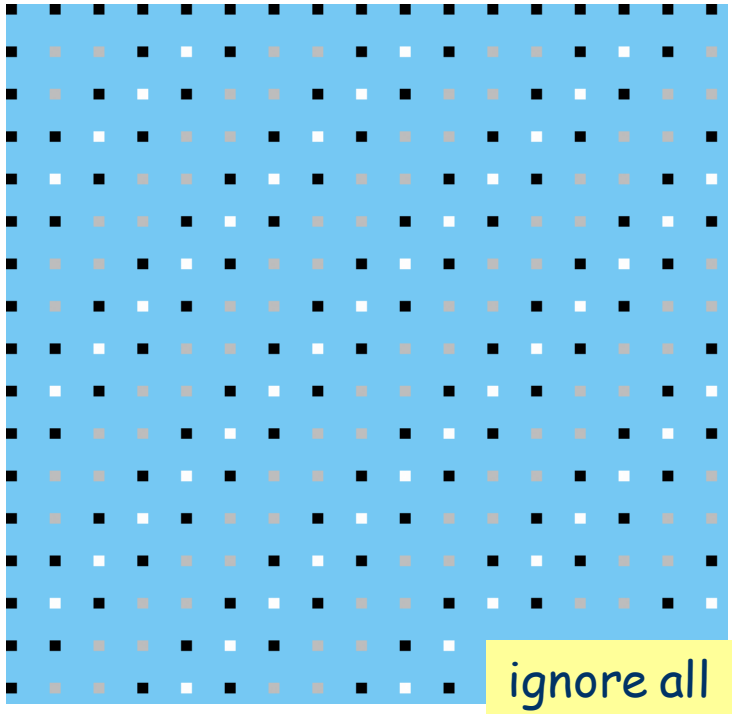


Keep every 4th pixel in every 4th row

Reducing an Image

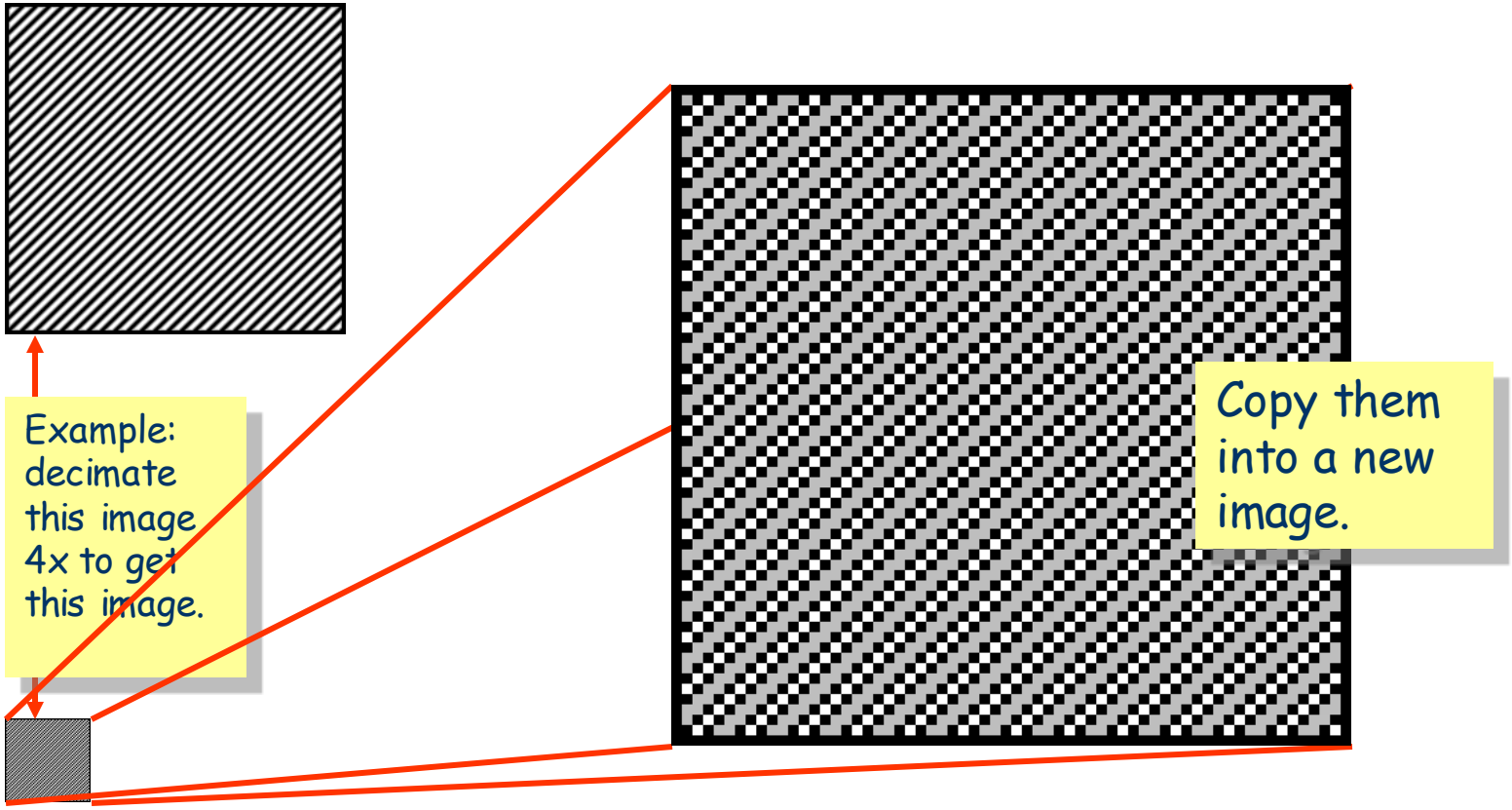


Example: decimate this image 4x to get this image.



ignore all the others

Reducing an Image



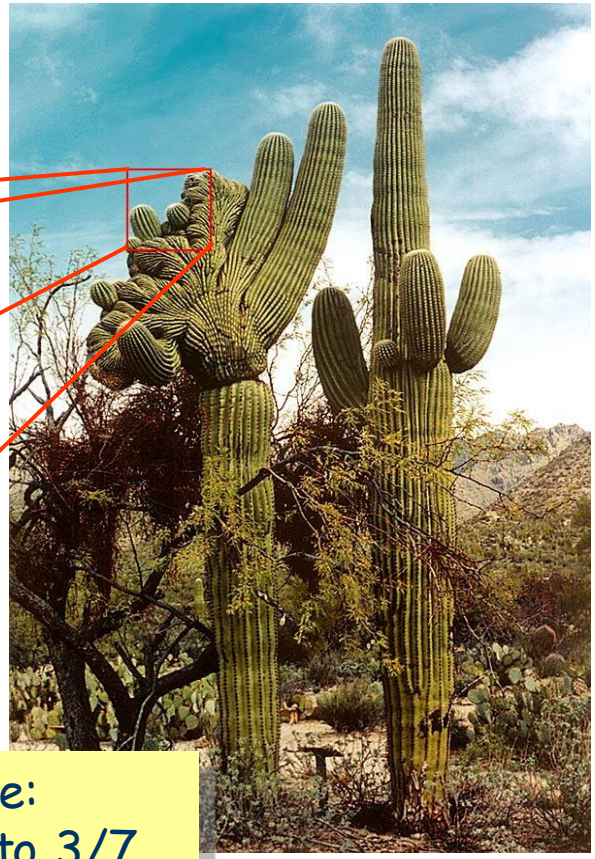
Nearest Neighbor Interpolation

The "Nearest Neighbor" algorithm is a generalization of pixel replication and decimation.

It also includes fractional resizing, *i.e.* resizing an image so that it has p/q of the pixels per row and p/q of the rows in the original. (p and q are both integers.)

Nearest Neighbor Interpolation

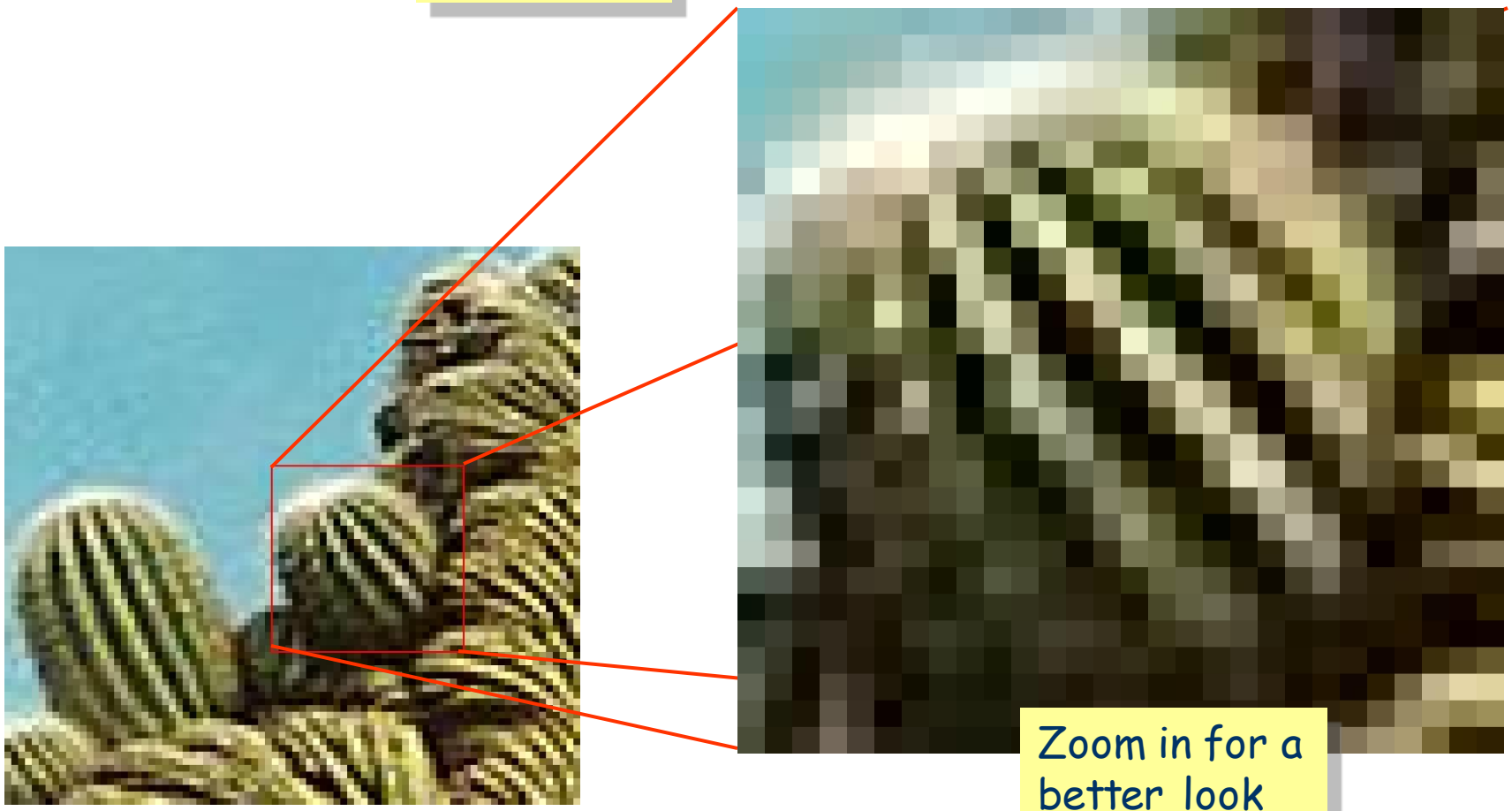
Zoom in on a section for a closer look at the process



Example: resize to 3/7 of the original

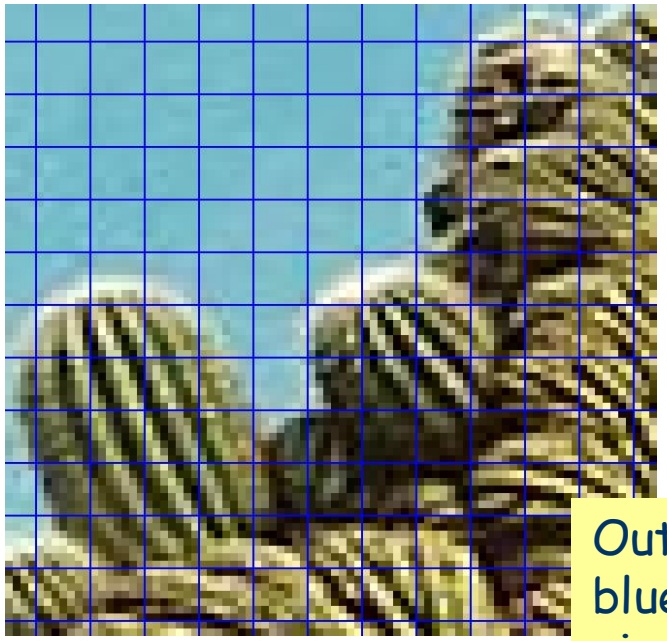
Nearest Neighbor Interpolation

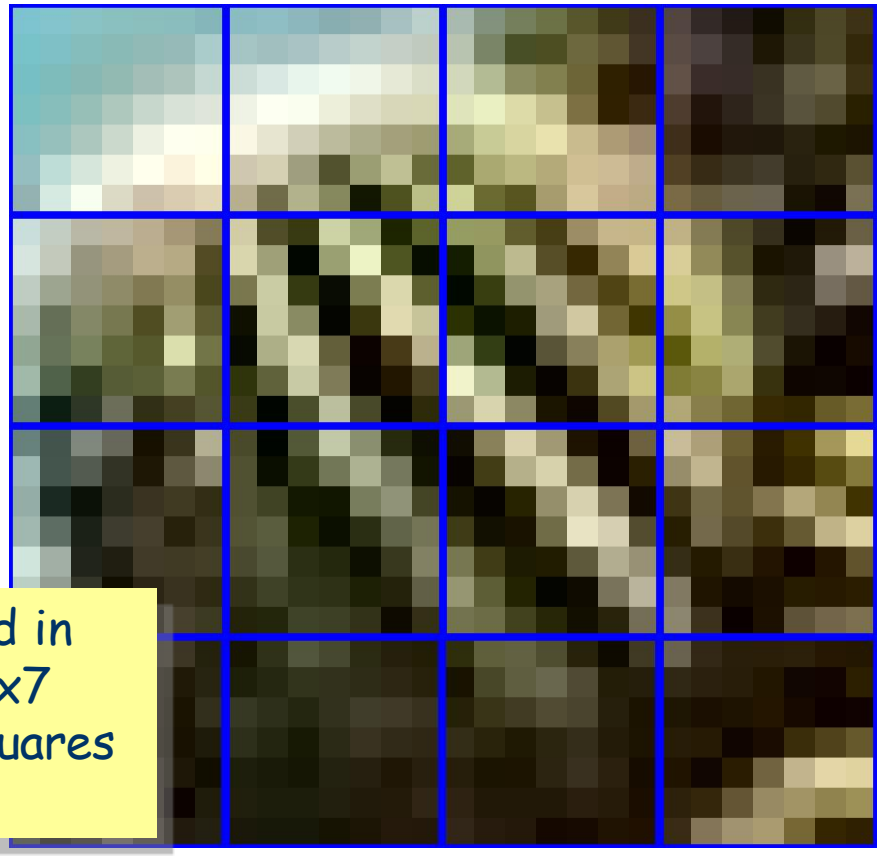
3/7 resize



Nearest Neighbor Interpolation

3/7 resize

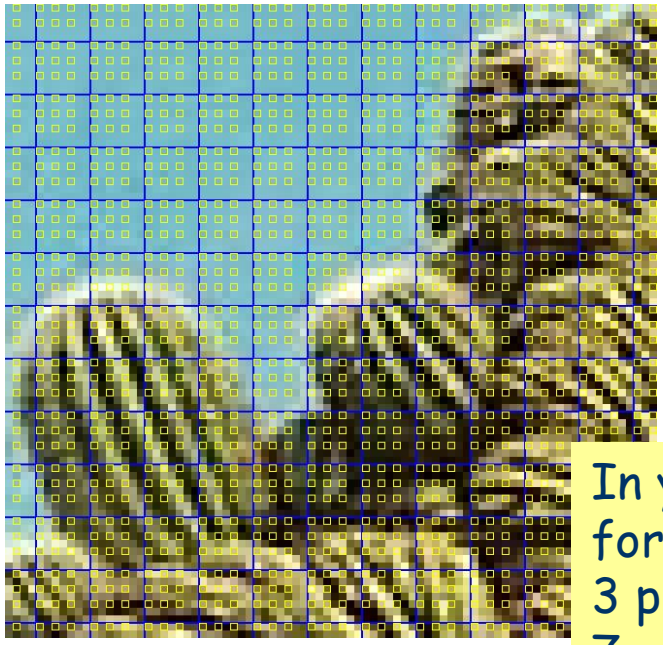


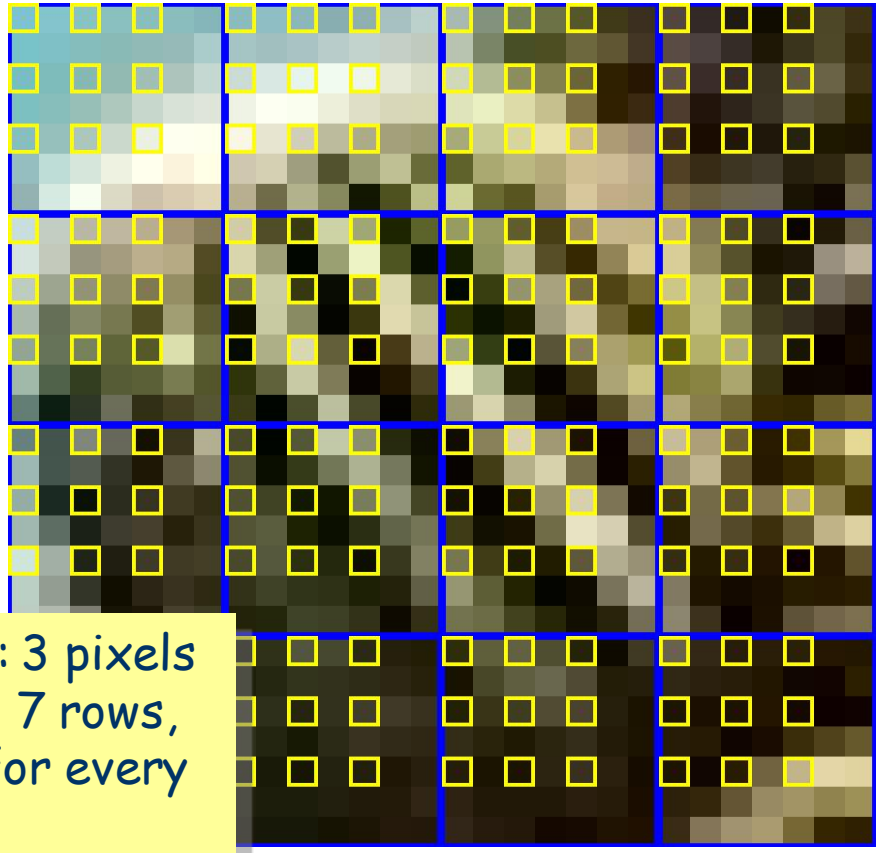


Outlined in blue: 7x7 pixel squares

Nearest Neighbor Interpolation

3/7 resize

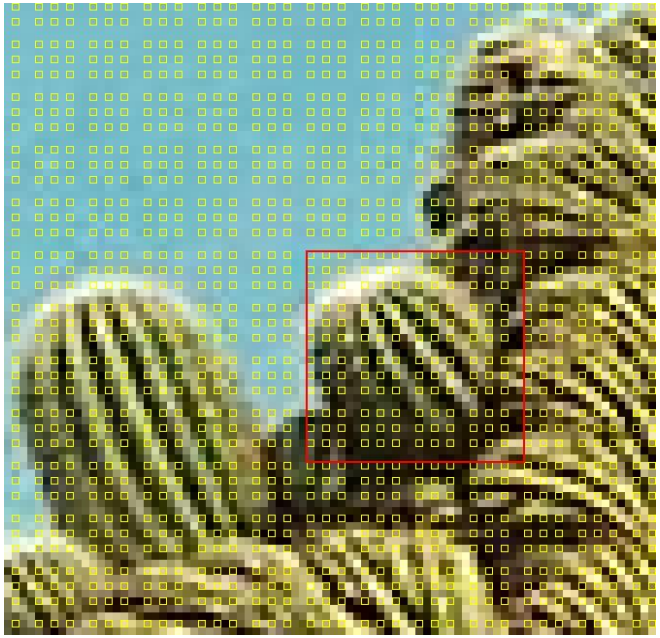




In yellow: 3 pixels for every 7 rows, 3 pixels for every 7 cols.

Nearest Neighbor Interpolation

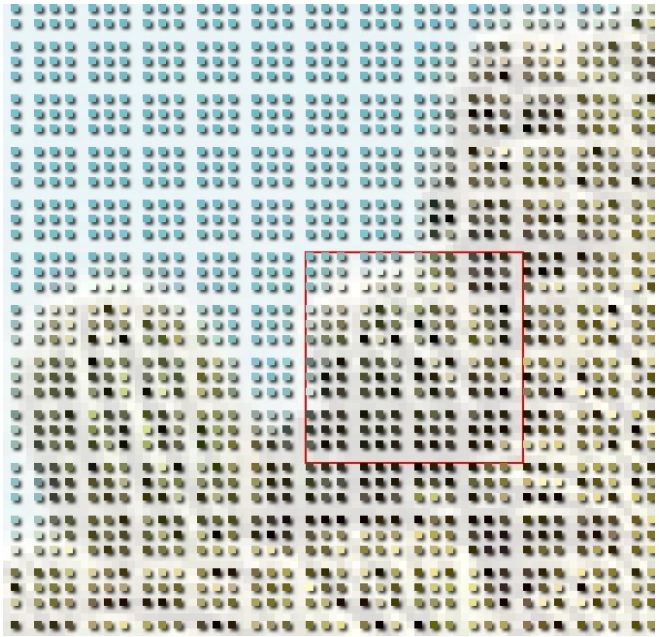
3/7 resize



Keep the highlighted pixels…

Nearest Neighbor Interpolation

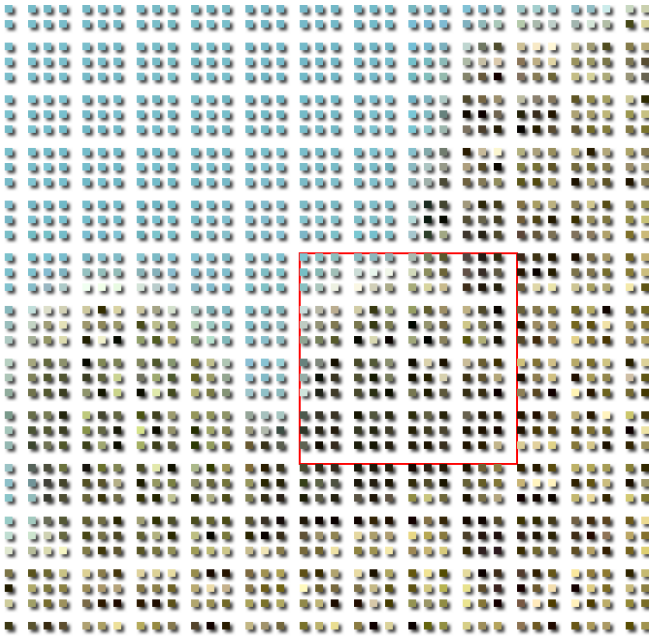
3/7 resize

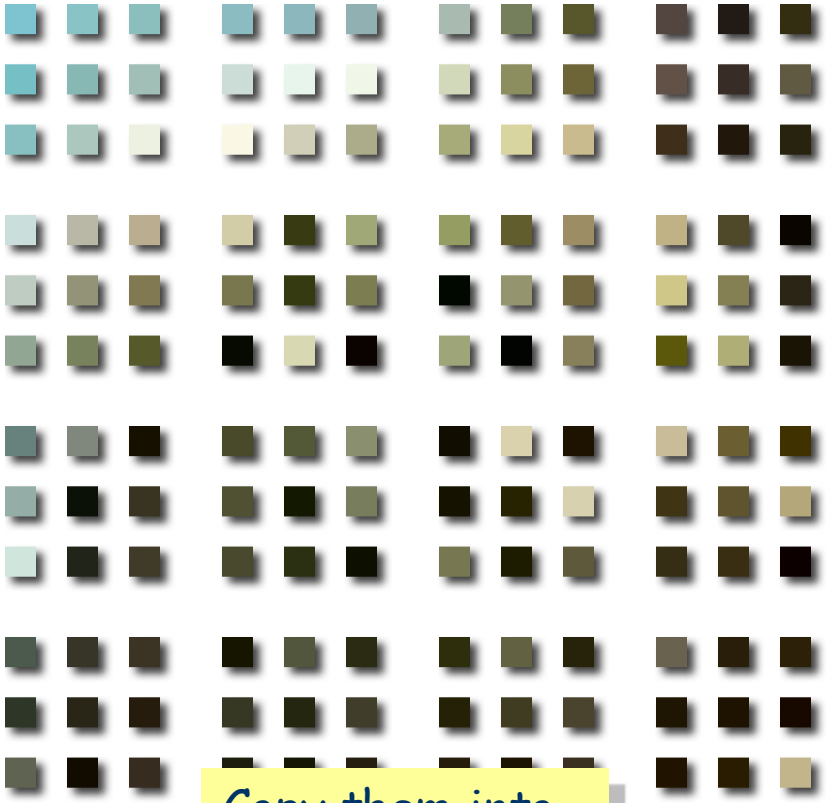


… don't keep the others.

Nearest Neighbor Interpolation

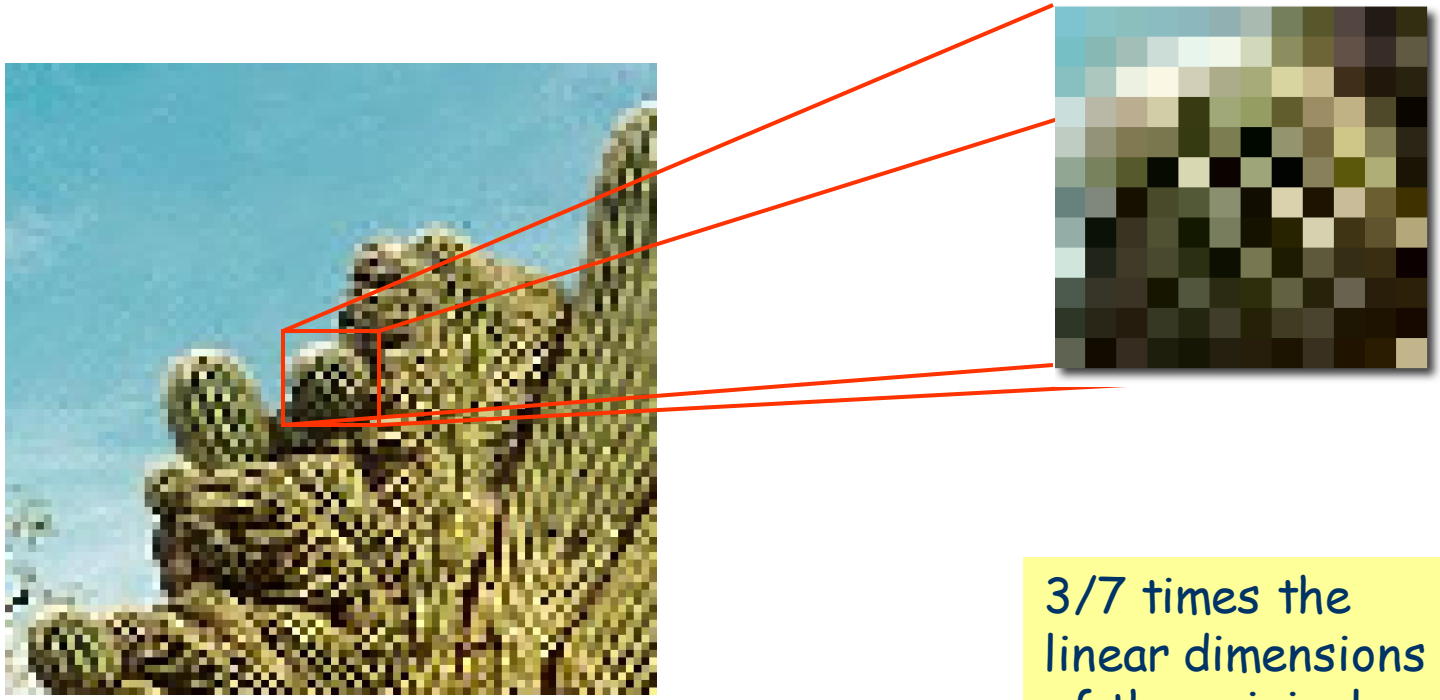
3/7 resize





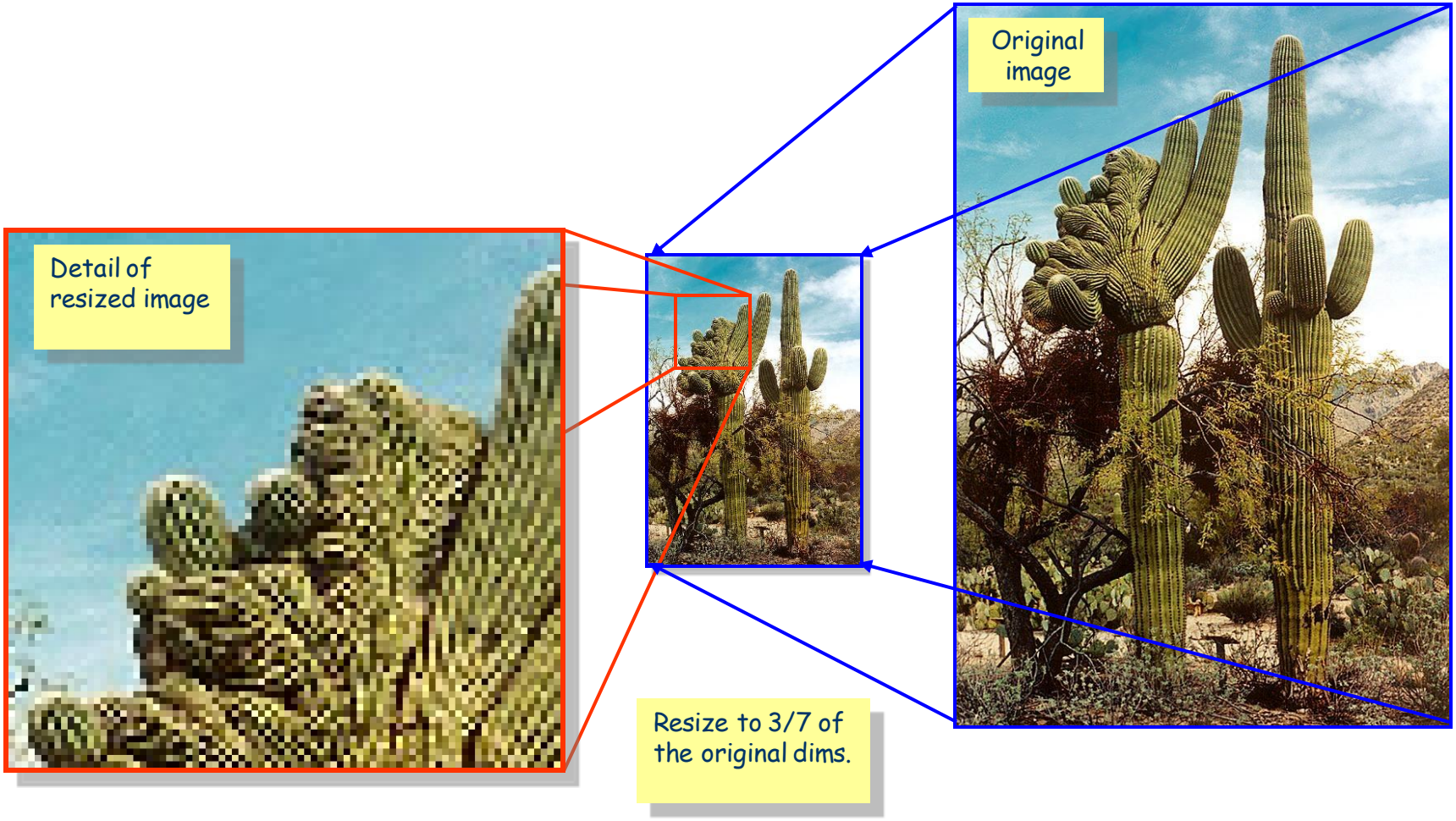
Copy them into a new image.

Nearest Neighbor Interpolation



3/7 times the linear dimensions of the original

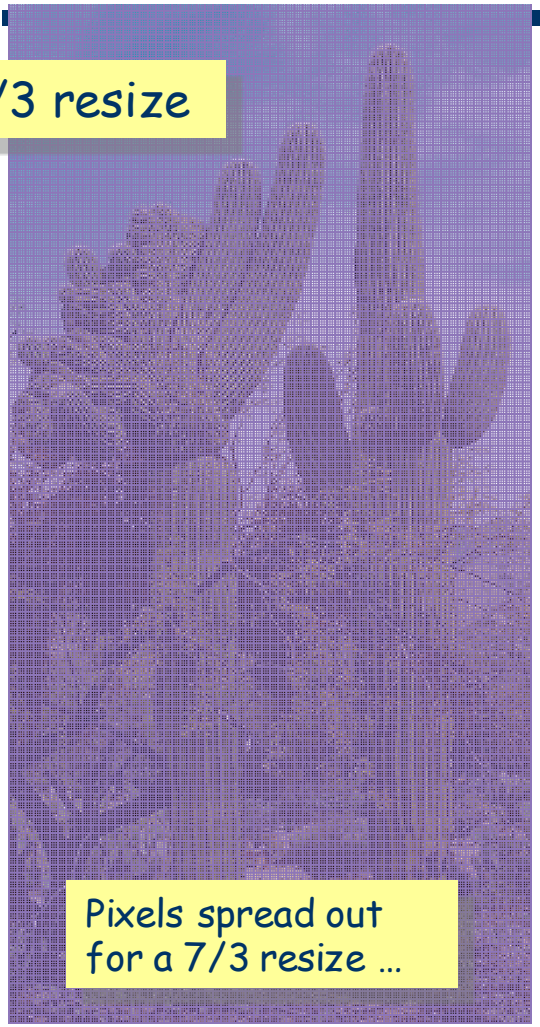
Nearest Neighbor Interpolation



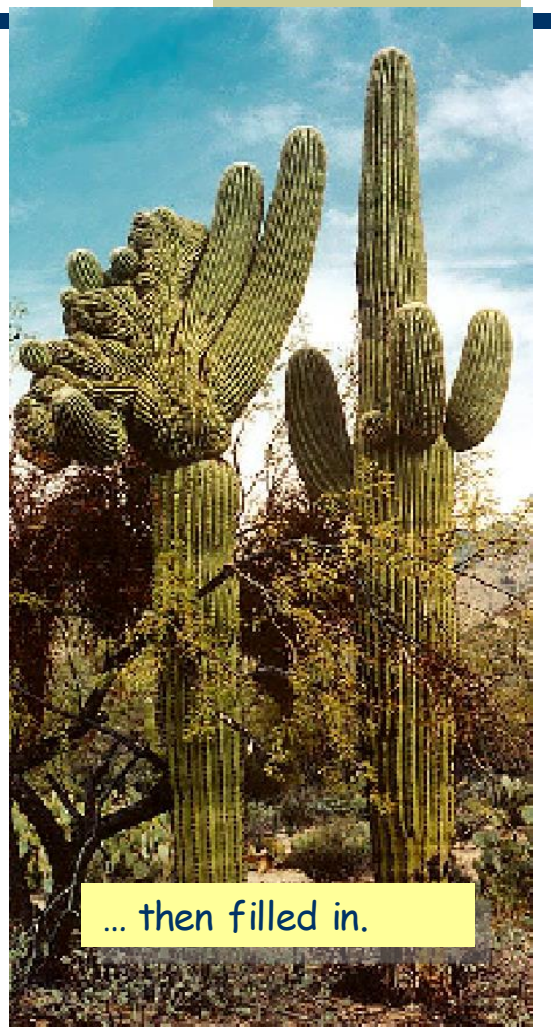
Nearest Neighbor Interpolation

Original image

7/3 resize

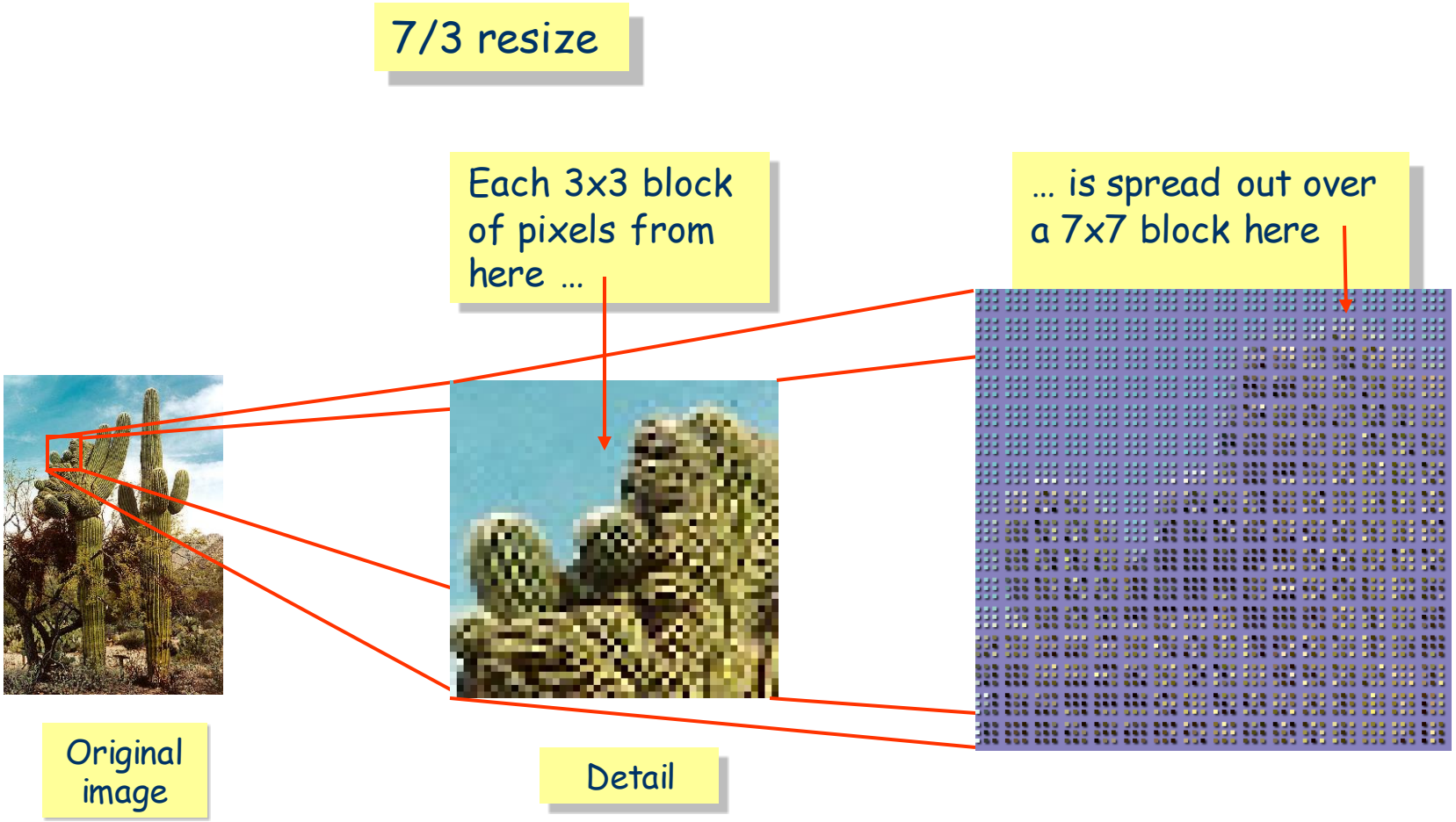


Pixels spread out for a 7/3 resize …



… then filled in.

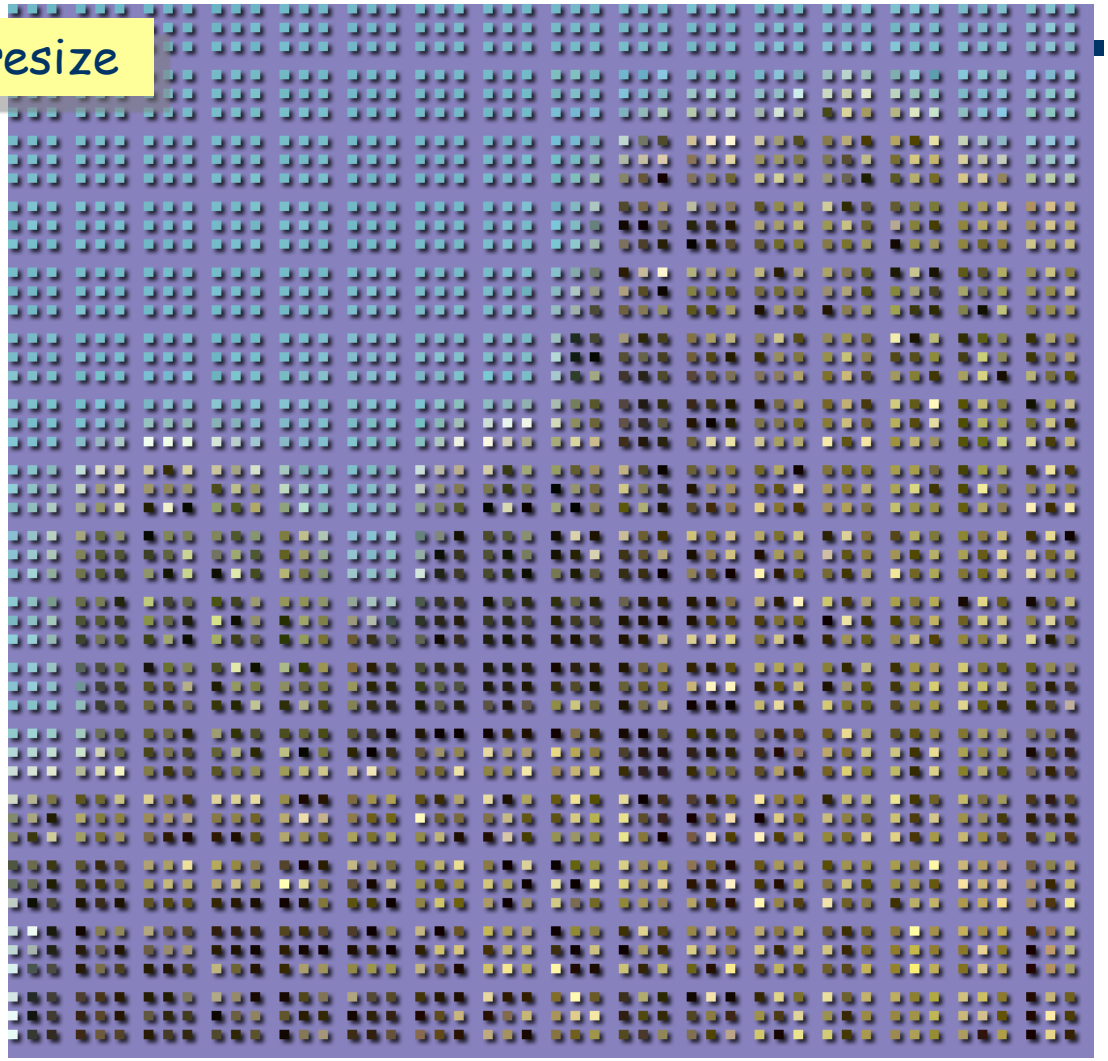
Nearest Neighbor Interpolation



Nearest Neighbor Interpolation

7/3 resize

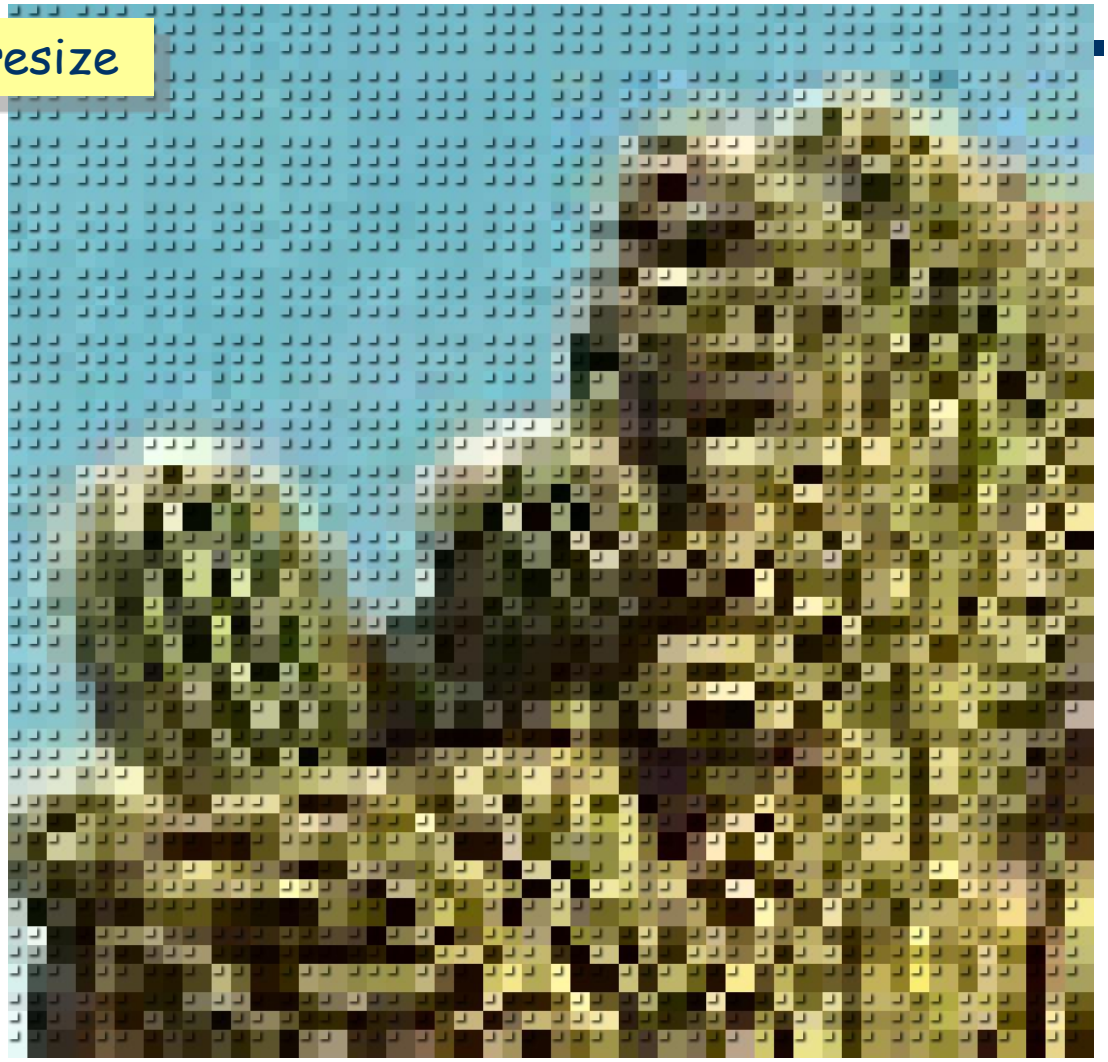
3x3 blocks distributed over 7x7 blocks



Nearest Neighbor Interpolation

7/3 resize

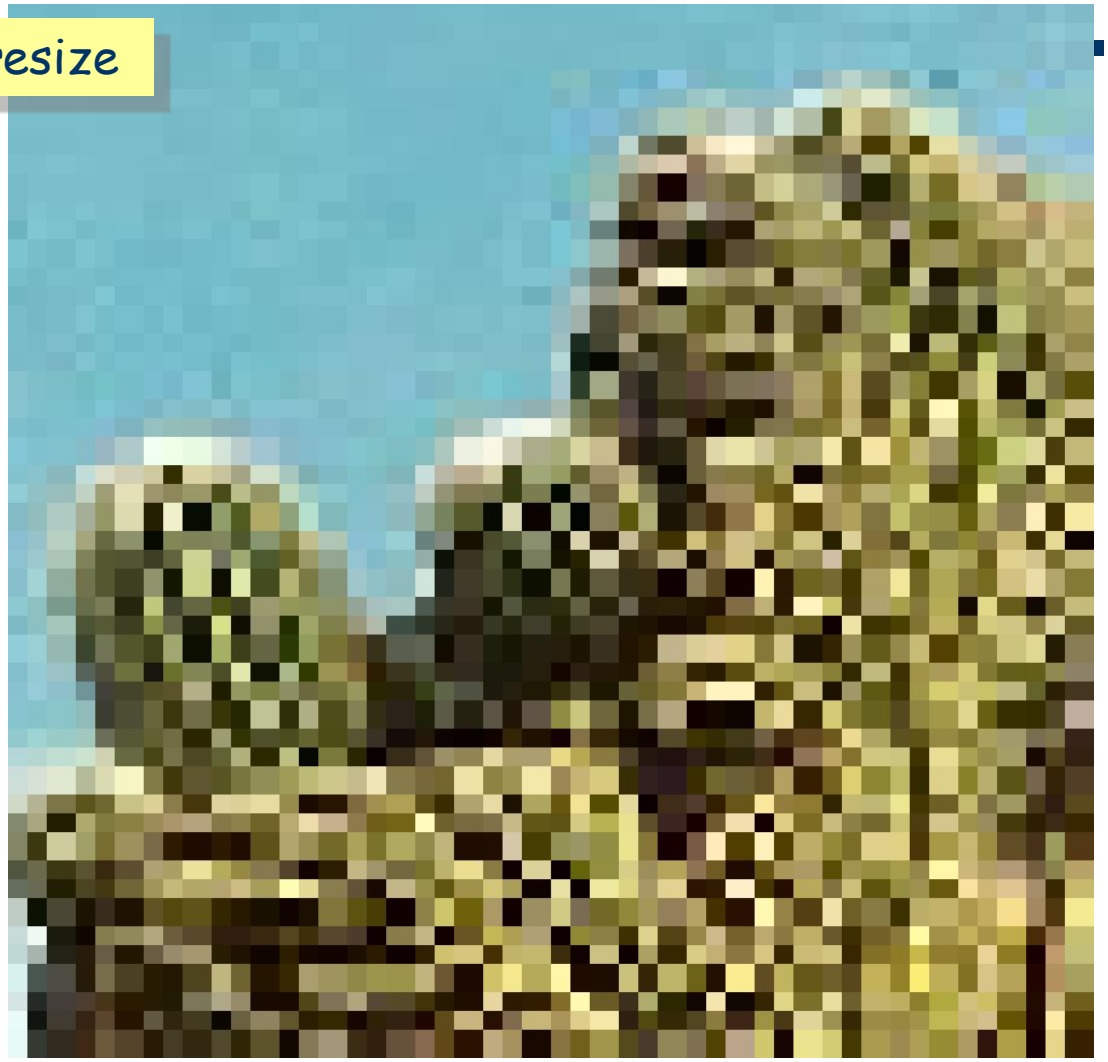
Empty pixels filled with color from non-empty pixel



Nearest Neighbor Interpolation

7/3 resize

Empty pixels filled with color from non-empty pixel



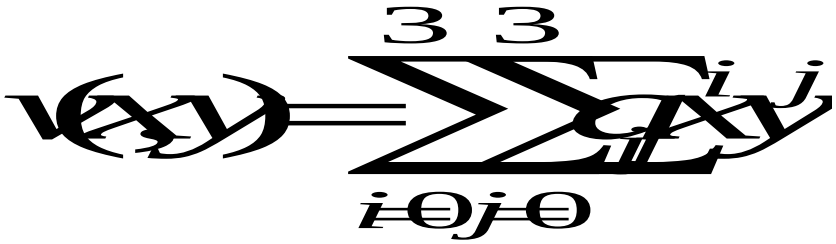
Nearest Neighbor Interpolation

Original image

Image Interpolation

- Nearest neighbour interpolation
  - Simple but produces undesired artefacts
- Bilinear Interpolation

- Bicubic Interpolation



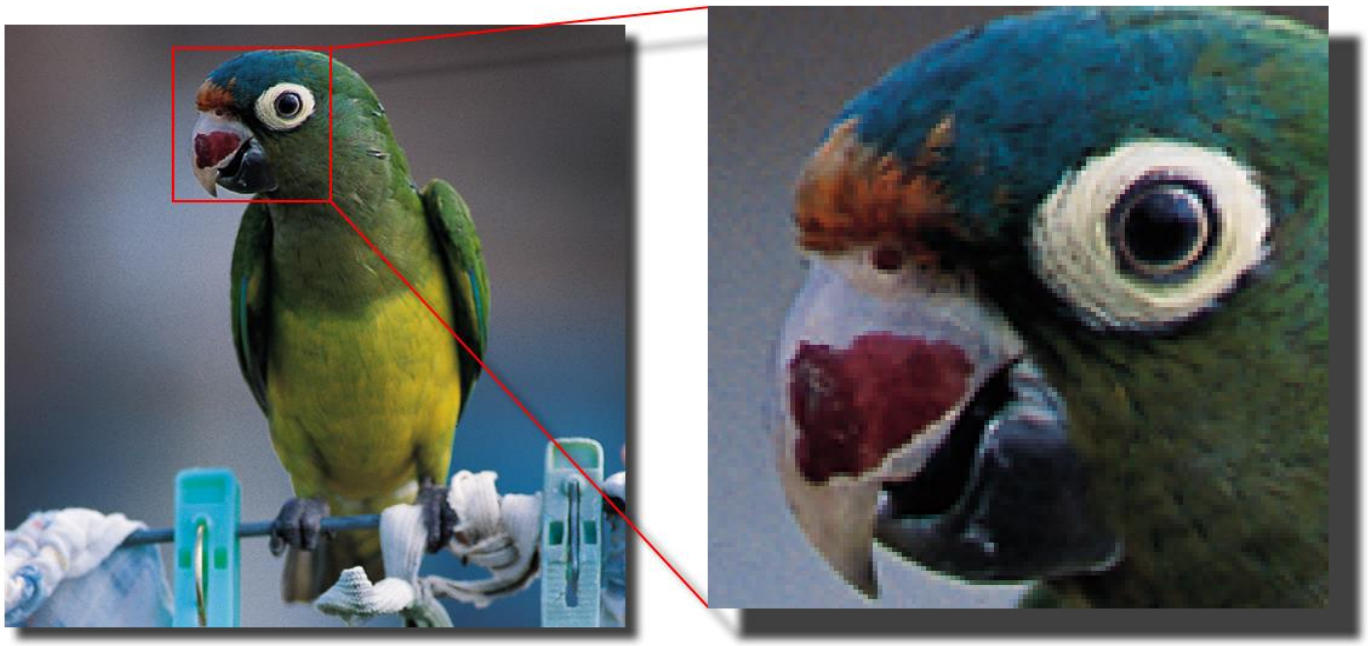
Interpolation: Comparison

We'll enlarge this image by a factor of 4 …

… via bilinear interpolation and compare it to a nearest neighbor enlargement.

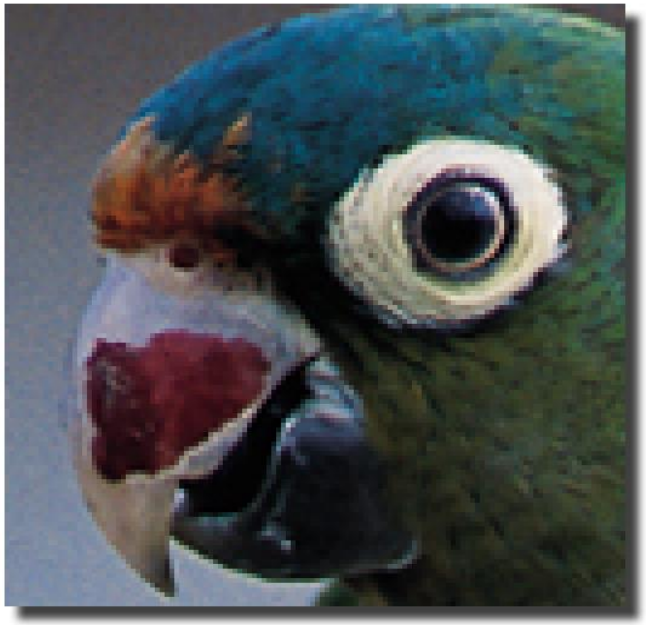
Interpolation: Comparison

Original Image



To better see what happens, we'll look at the parrot's eye.

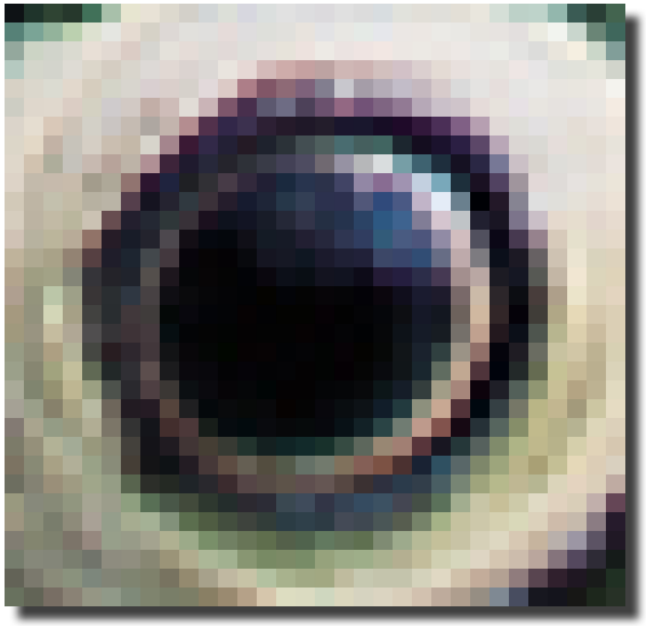
Interpolation: Comparison



Pixel replication

Bilinear interpolation

Interpolation: Comparison



Pixel replication

Bilinear interpolation

Acknowledgements

- Digital Image Processing", Rafael C. Gonzalez & Richard E. Woods, Addison-Wesley, 2002
- Peters, Richard Alan, II, Lectures on Image Processing, Vanderbilt University, Nashville, TN, April 2008
- Brian Mac Namee, Digitial Image Processing, School of Computing, Dublin Institute of Technology
- Computer Vision for Computer Graphics, Mark Borg

Material in these slides has been taken from the following resources