LAST CLASS

Connected Components

- **Recursive algorithm**

- Let is assume that region pixels have the value 0 (black) and that background pixels have the value 255 (white).

(1) Scan the image to find an unlabeled 0 (pixel and assign it a new label L.
(2) Recursively assign a label L to all of its 0 neighbors.
(3) Stop if there are no more unlabeled 0 pixels.
(4) Go to step 1.

- **Sequential algorithm**

- The sequential algorithm usually requires two passes over the image.

- It works with only two rows of an image at a time.

(1) Scan the image left to right, top to bottom.

(2) If the pixel is 0, then:

(2.1) If only one of its upper and left neighbors has a label, then copy the label.

(2.2) If both have the same label, then copy the label.

(2.3) If both have different labels, then copy the upper's label and enter the labels in the equivalence table as equivalent labels.

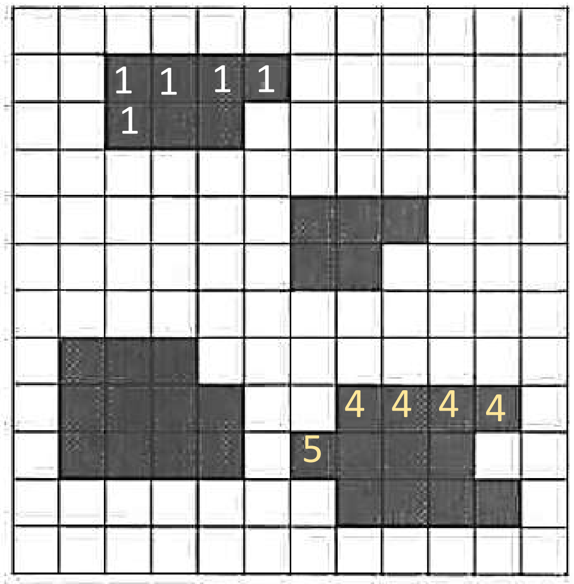
(2.4) Otherwise assign a new label to this pixel and enter this label in the equivalence table.

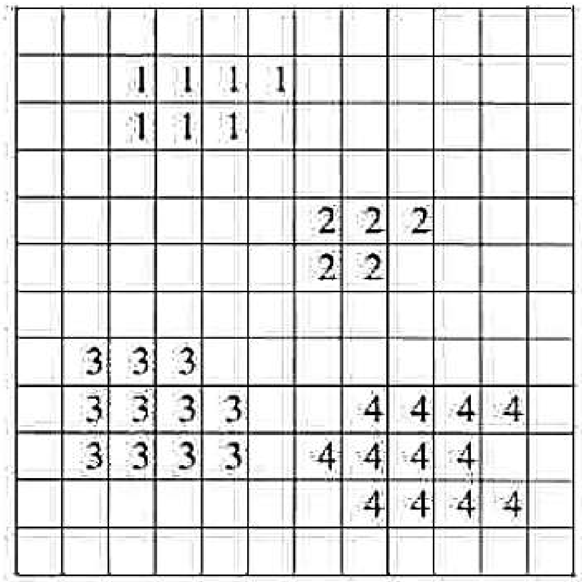
(3) If there are no more pixels to consider, then go to step 2.

(4) Find the lowest label for each equivalent set in the equivalence table.

(5) Scan the image. Replace each label by the lowest label in its equivalent set.

• **Sequential algorithm**





Distance Measures

The following are the different Distance measures:

Euclidean Distance :

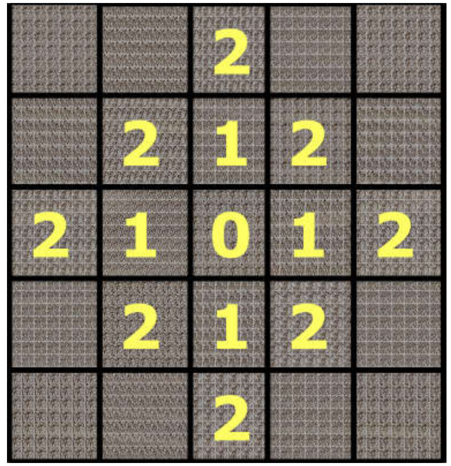
$$D_e(p,q) = [(x-s)^2 + (y-t)^2]^{1/2}$$

Cityblock Distance :

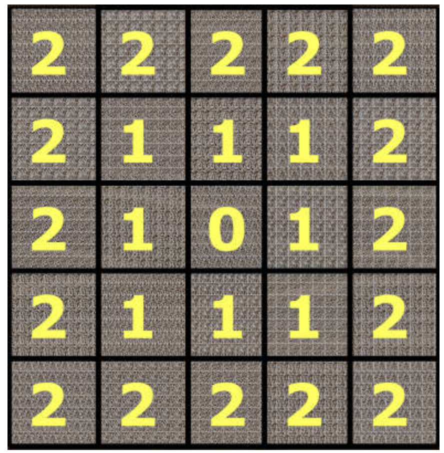
$$D_4(p,q) = |x-s| + |y-t|$$

Chess-board Distance :

$$D_8(p,q) = max(|x-s|,|y-t|)$$



Cityblock Distance



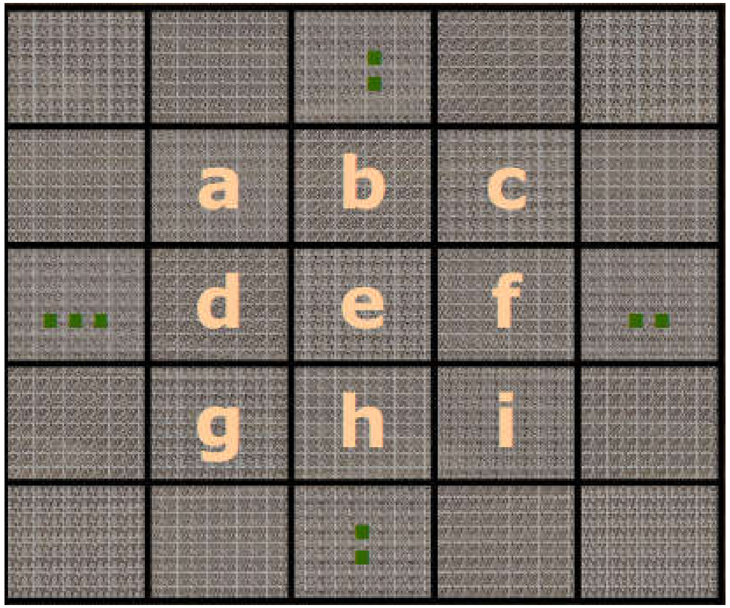
Chess-board Distance

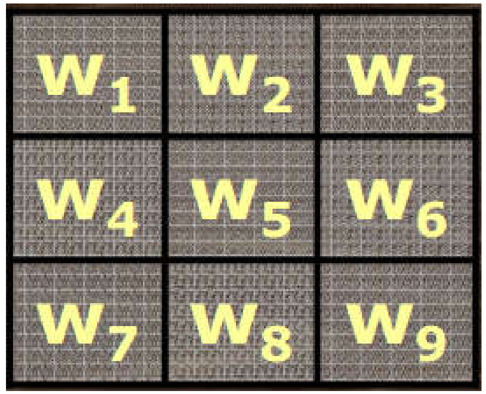
Arithmetic/Logic Operations:

- – Addition : p + q
- – Subtraction : p –q
- – Multiplication : p*q
- – Division : p/q
- – AND : p AND q
- – OR : p OR q
- – Complement: NOT(q)

<u>**Neighborhood based arithmetic/Logic**</u>:

Value assigned to a pixel at position 'e' is a function of its neighbors and a set of window functions.





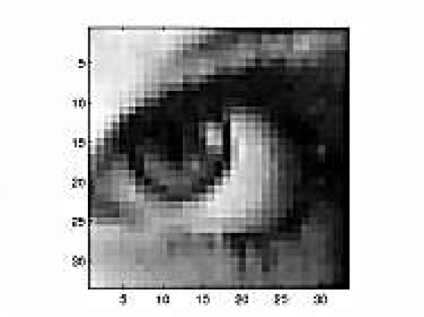
$$p = (w_1 a + w_2 b + w_3 c + w_4 d + w_5 e + w_6 f + w_7 g + w_8 h + w_9 i)$$

$$= \sum w_i f_i$$

Image Interpolation

Image interpolation occurs in all digital images at some stage

- Resizing (resampling)
- Remapping (geometrical tansformations- rotation, change of perspective,...)
- Inpainting (restauration of *holes*)
- Morphing, nonlinear transformations



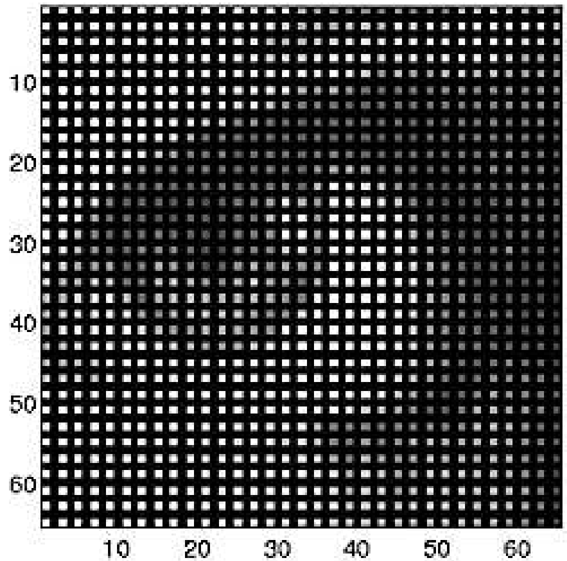
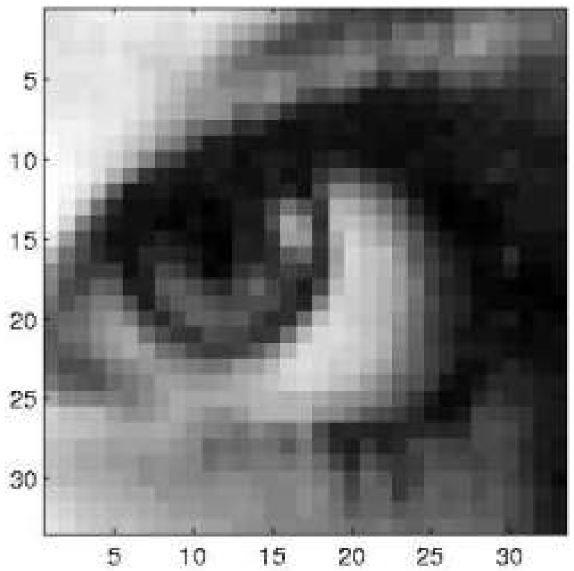


Image interpolation occurs in all digital images at some stage

- Resizing (resampling)
- Remapping (geometrical tansformations- rotation, change of perspective,...)
- Inpainting (restauration of *holes*)
- Morphing, nonlinear transformations



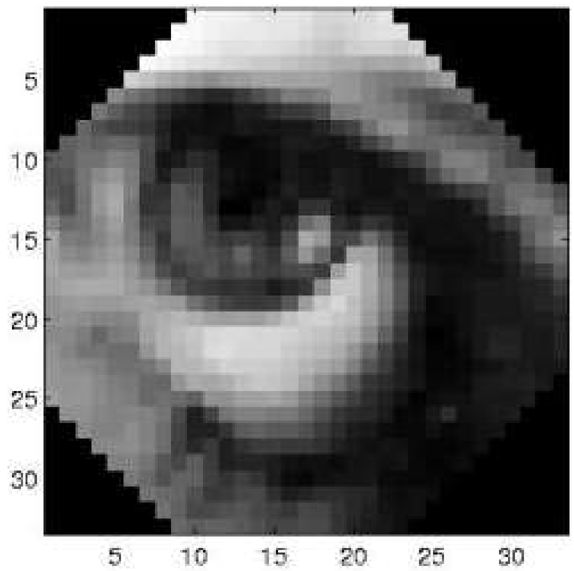
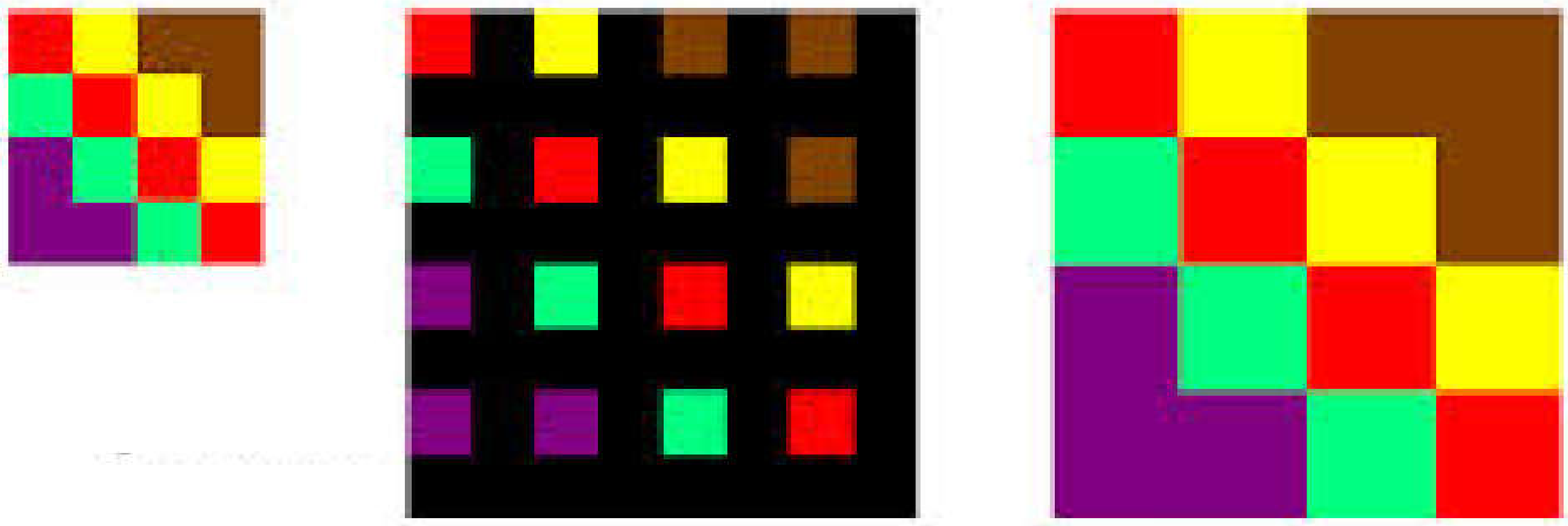


Image interpolation occurs in all digital images at some stage

- Resizing (resampling)
- Remapping (geometrical tansformations- rotation, change of perspective,...)
- Inpainting (restauration of *holes*)
- Morphing, nonlinear transformations

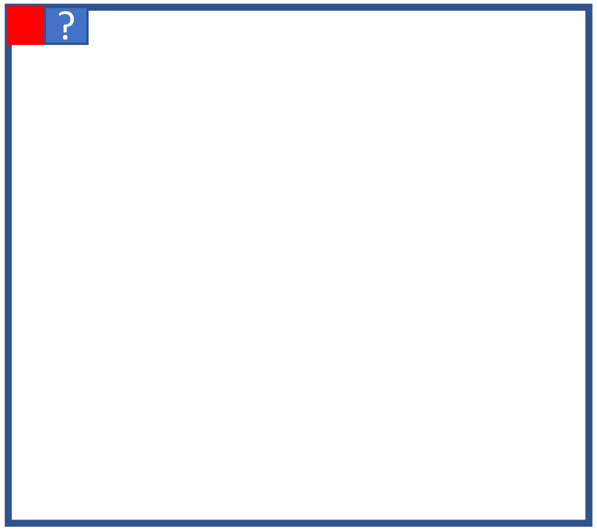
Nearest Neighbor

- Most basic method
- Requires the least processing time
- Only considers one pixel: the closest one to the interpolated point
- Has the effect of simply making each pixel bigger



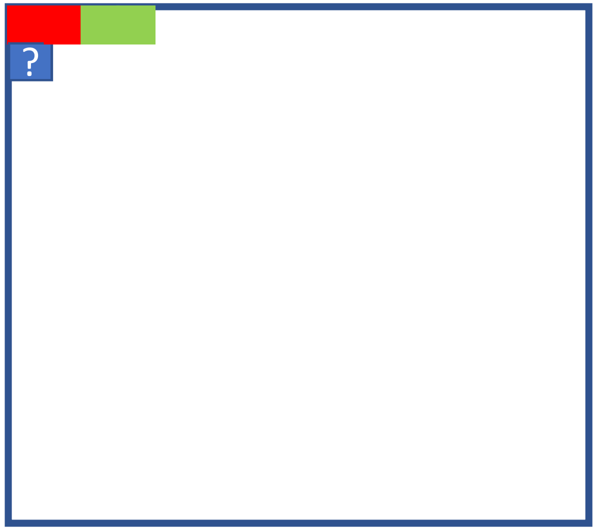
Nearest Neighbor

Nearest Neighbor

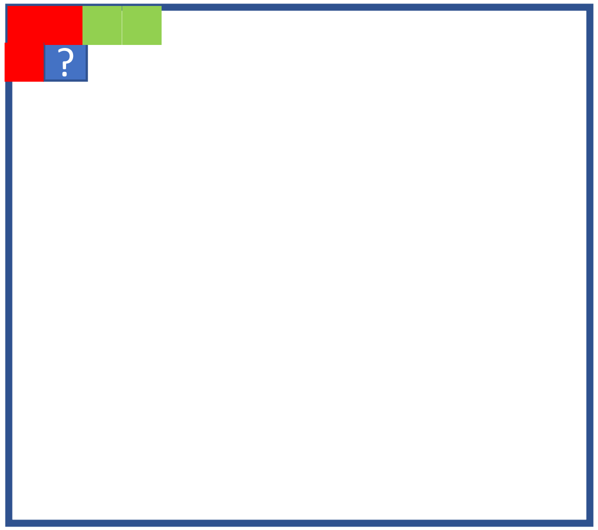


Nearest Neighbor

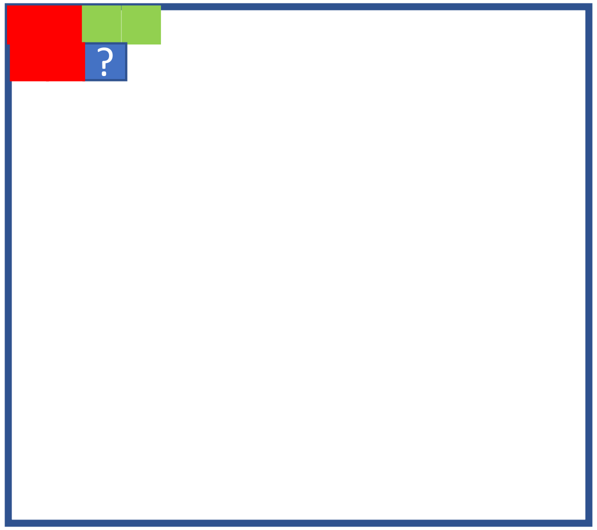
Nearest Neighbor



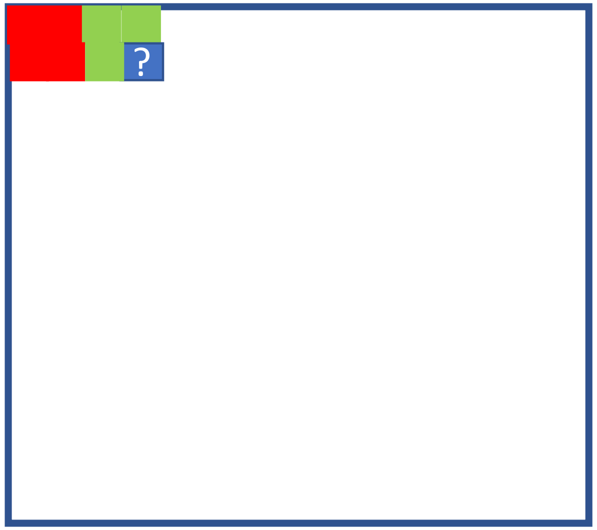
Nearest Neighbor



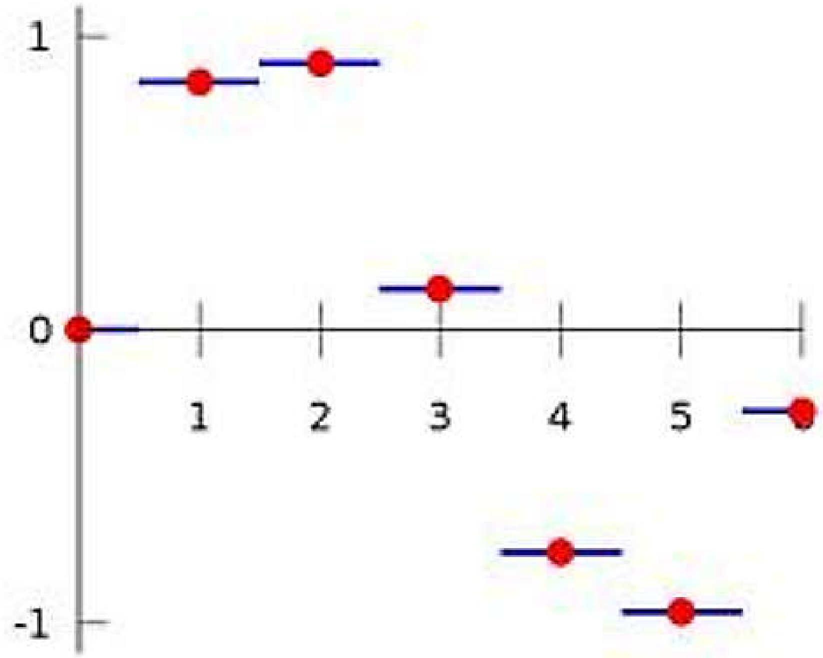
Nearest Neighbor



Nearest Neighbor

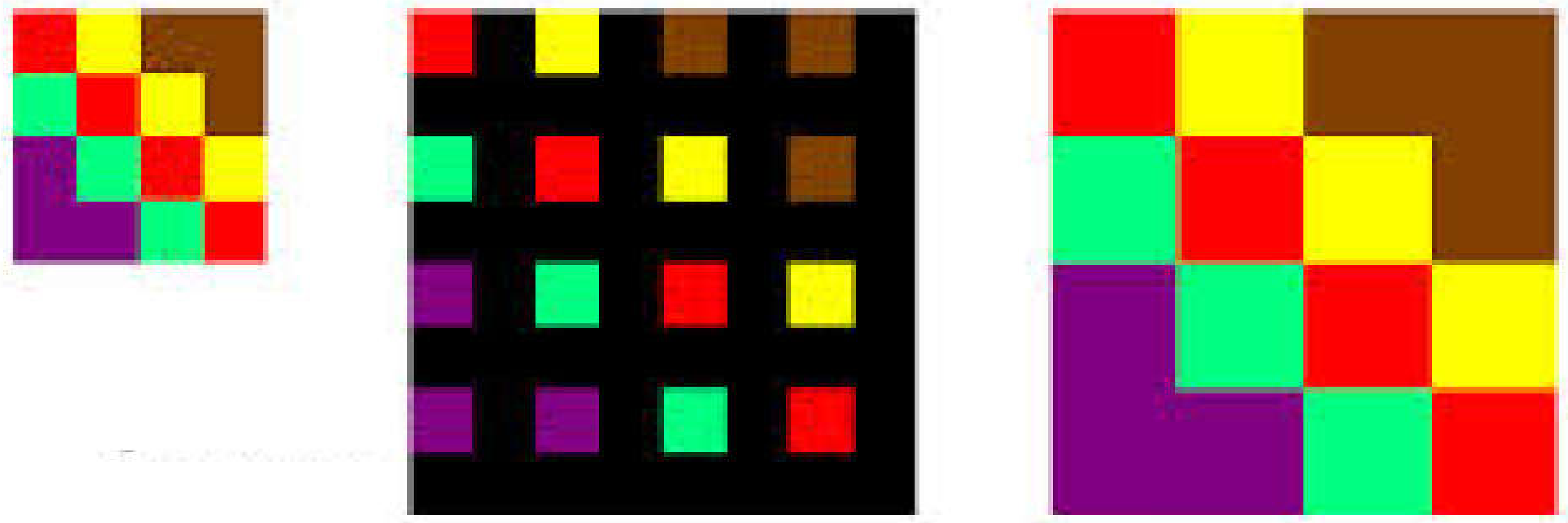


Nearest Neighbor: 1D Equivalence



Nearest Neighbor

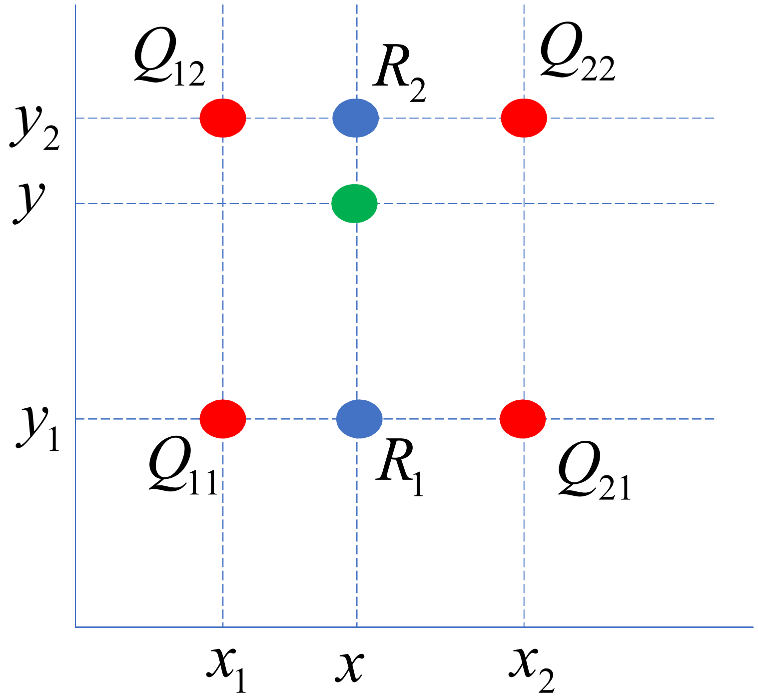
- Most basic method
- Requires the least processing time
- Only considers one pixel: the closest one to the interpolated point
- Has the effect of simply making each pixel bigger



Bilinear Interpolation

Suppose that we want to find the value of the unknown function f at the point (x, y).

Assume that we know the value of f at the four points $Q_{11} = (x_1, y_1)$, $Q_{12} = (x_1, y_2)$, $Q_{21} = (x_2, y_1)$, and $Q_{22} = (x_2, y_2)$.



We first do linear interpolation in the *x*-direction. This yields

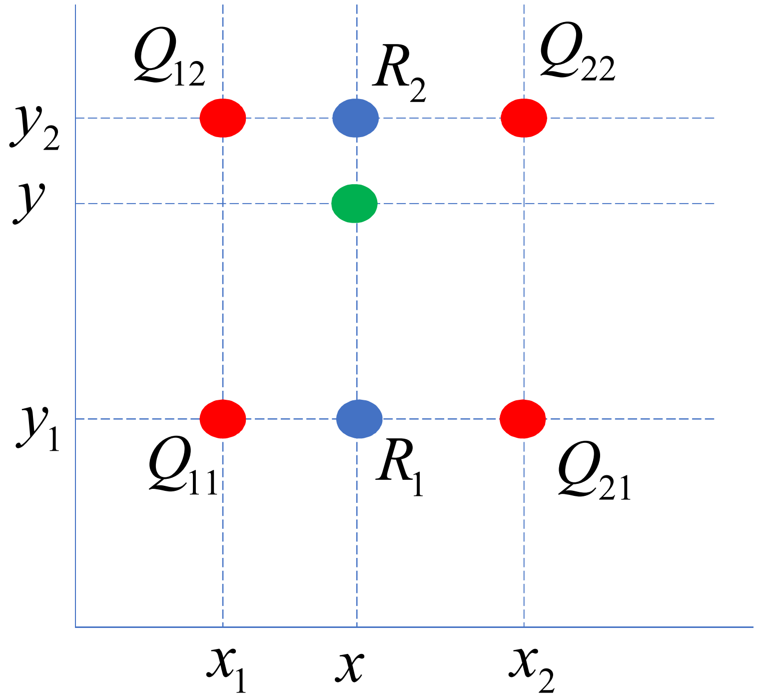
$$f(x, y_1) \approx \frac{x_2 - x}{x_2 - x_1} f(Q_{11}) + \frac{x - x_1}{x_2 - x_1} f(Q_{21}),$$

$$f(x, y_2) \approx \frac{x_2 - x}{x_2 - x_1} f(Q_{12}) + \frac{x - x_1}{x_2 - x_1} f(Q_{22}).$$

Bilinear Interpolation

Suppose that we want to find the value of the unknown function f at the point (x, y).

Assume that we know the value of f at the four points $Q_{11} = (x_1, y_1)$, $Q_{12} = (x_1, y_2)$, $Q_{21} = (x_2, y_1)$, and $Q_{22} = (x_2, y_2)$.



We first do linear interpolation in the *x*-direction. This yields

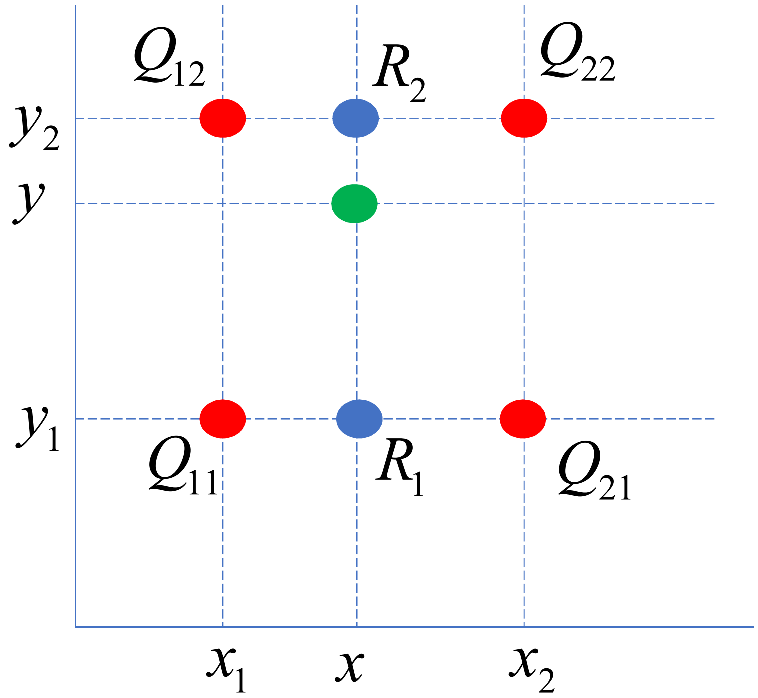
$$f(x, y_1) \approx \cancelto{0}{\frac{x_2 - x}{x_2 - x_1}} f(Q_{11}) + \cancelto{1}{\frac{x - x_1}{x_2 - x_1}} f(Q_{21}),$$

$$f(x, y_2) \approx \cancelto{0}{\frac{x_2 - x}{x_2 - x_1}} f(Q_{12}) + \cancelto{1}{\frac{x - x_1}{x_2 - x_1}} f(Q_{22}).$$

$$\boxed{x \to x_2}$$

Bilinear Interpolation

$$f(x, y_1) \approx \frac{x_2 - x}{x_2 - x_1} f(Q_{11}) + \frac{x - x_1}{x_2 - x_1} f(Q_{21}),$$
$$f(x, y_2) \approx \frac{x_2 - x}{x_2 - x_1} f(Q_{12}) + \frac{x - x_1}{x_2 - x_1} f(Q_{22}).$$



We proceed by interpolating in the *y*-direction to obtain the desired estimate:

$$\begin{aligned} f(x, y) &\approx \frac{y_2 - y}{y_2 - y_1} f(x, y_1) + \frac{y - y_1}{y_2 - y_1} f(x, y_2) \\ &= \frac{y_2 - y}{y_2 - y_1} \left(\frac{x_2 - x}{x_2 - x_1} f(Q_{11}) + \frac{x - x_1}{x_2 - x_1} f(Q_{21}) \right) + \frac{y - y_1}{y_2 - y_1} \left(\frac{x_2 - x}{x_2 - x_1} f(Q_{12}) + \frac{x - x_1}{x_2 - x_1} f(Q_{22}) \right) \end{aligned}$$

Bilinear Interpolation

Alternative Algorithm

An alternative way to write the solution to the interpolation problem is

$$f(x,y) \approx a_0 + a_1 x + a_2 y + a_3 xy,$$

where the coefficients are found by solving the linear system

$$\begin{bmatrix} 1 & x_1 & y_1 & x_1 y_1 \\ 1 & x_1 & y_2 & x_1 y_2 \\ 1 & x_2 & y_1 & x_2 y_1 \\ 1 & x_2 & y_2 & x_2 y_2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} f(Q_{11}) \\ f(Q_{12}) \\ f(Q_{21}) \\ f(Q_{22}) \end{bmatrix},$$

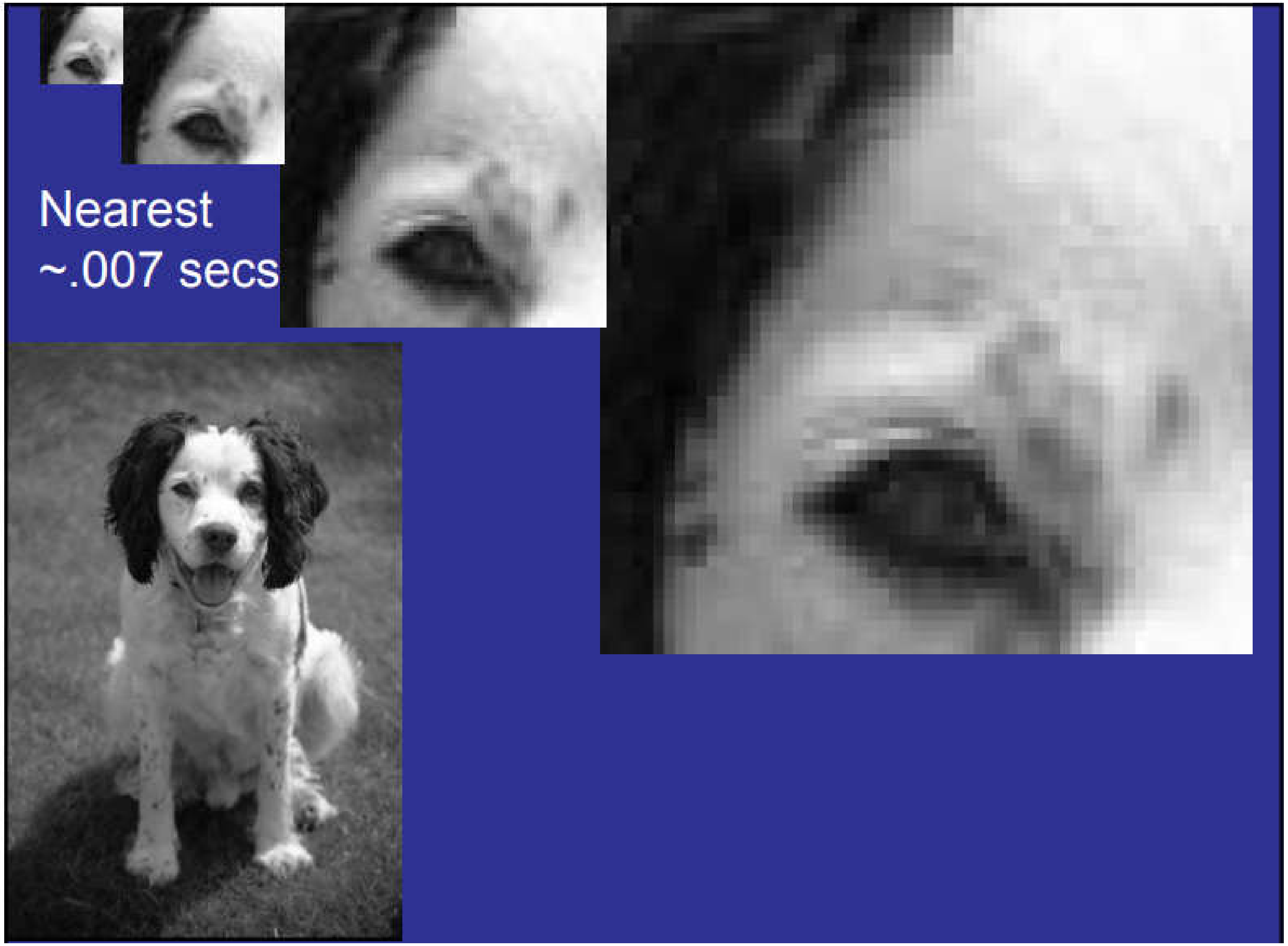
Bicubic Interpolation

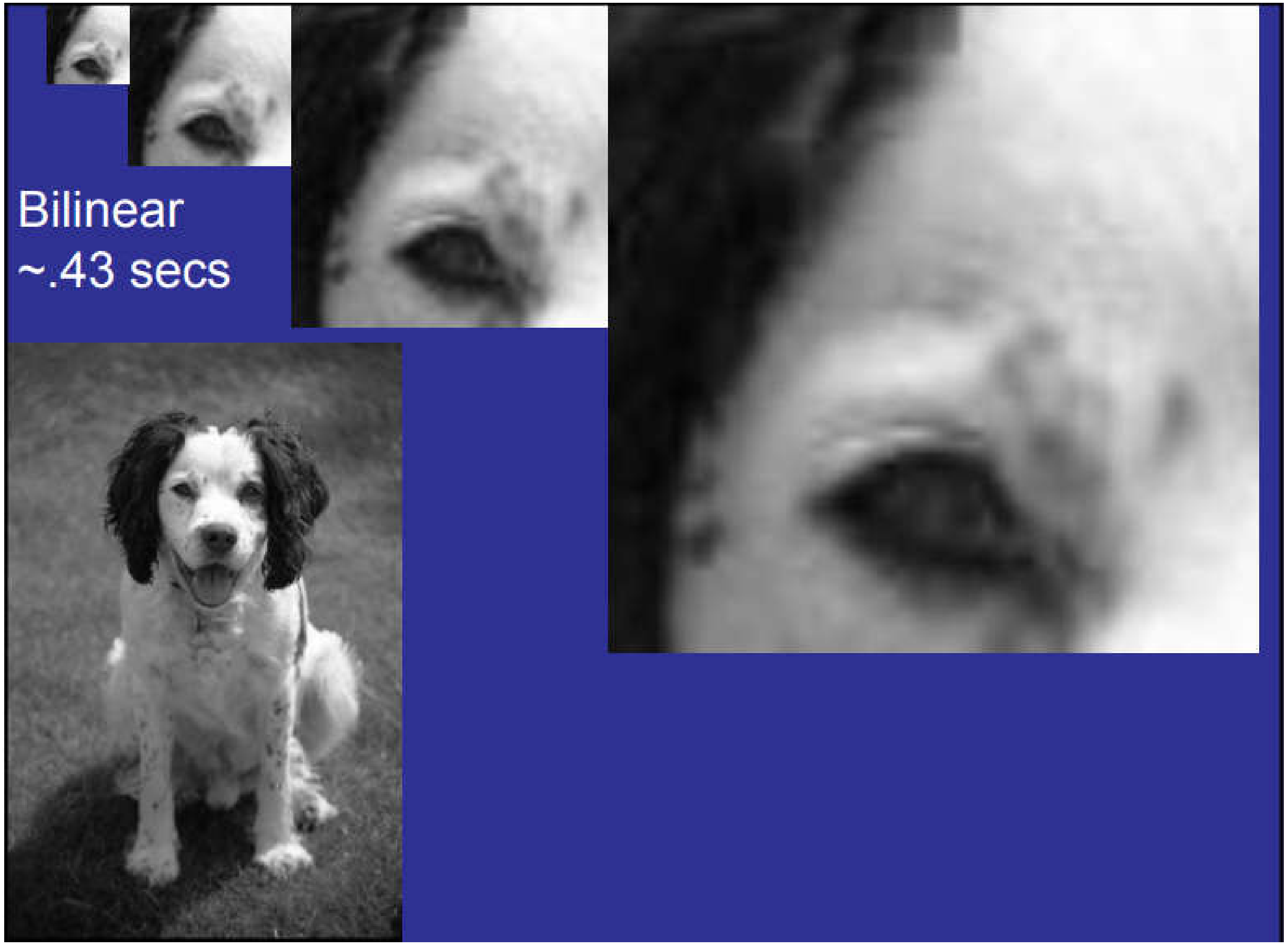
In Bilinear interpolation, we assume

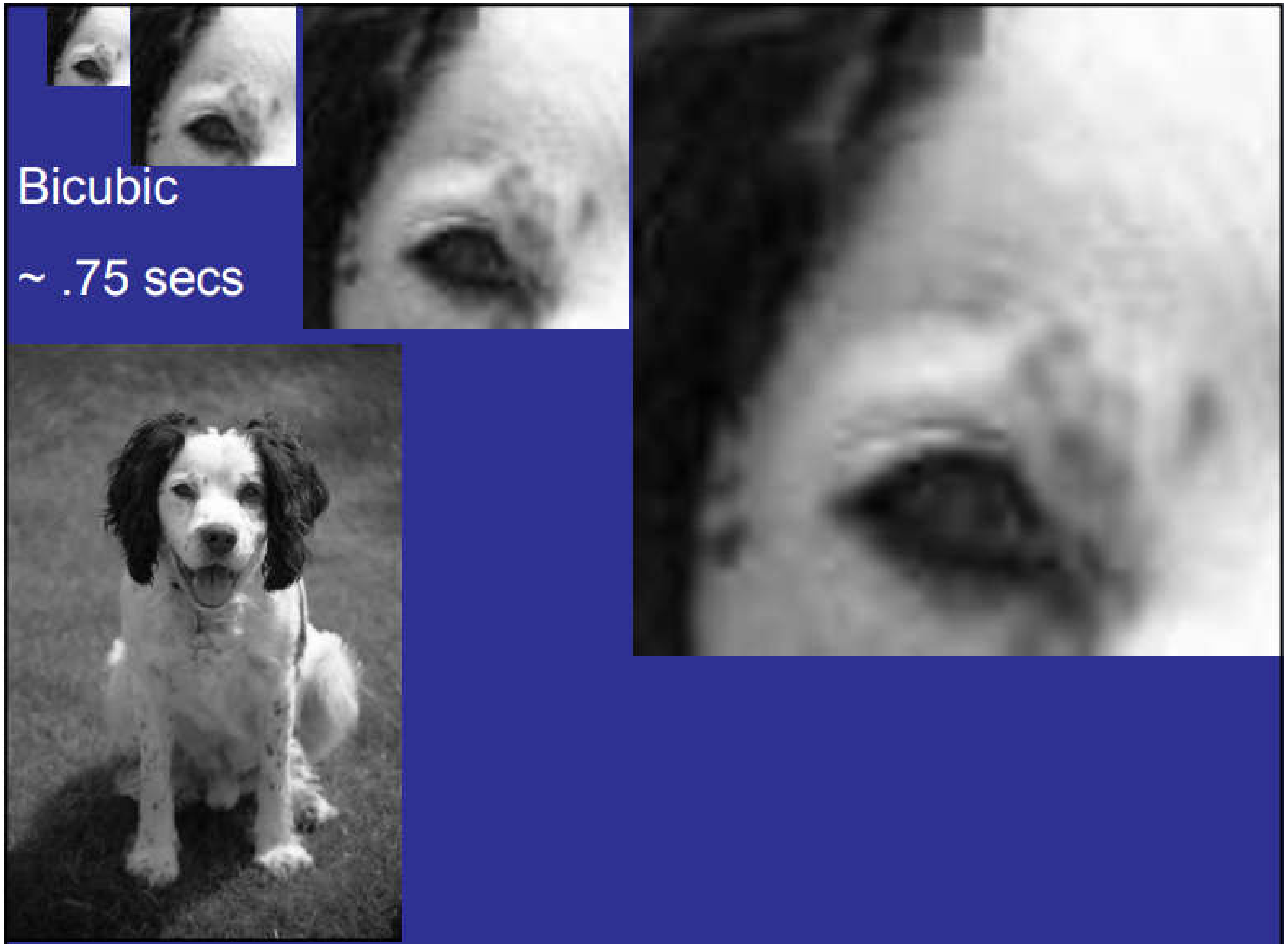
$$f(x,y) \approx a_0 + a_1 x + a_2 y + a_3 xy \quad \approx \sum_{j=0}^{1} \sum_{i=0}^{1} a_{ij} x^i y^j$$

In **Bicubic** interpolation, we assume

$$f(x,y) \approx \sum_{j=0}^{3} \sum_{i=0}^{3} a_{ij} x^i y^j$$


Nearest
~.007 secs


Bilinear
~.43 secs

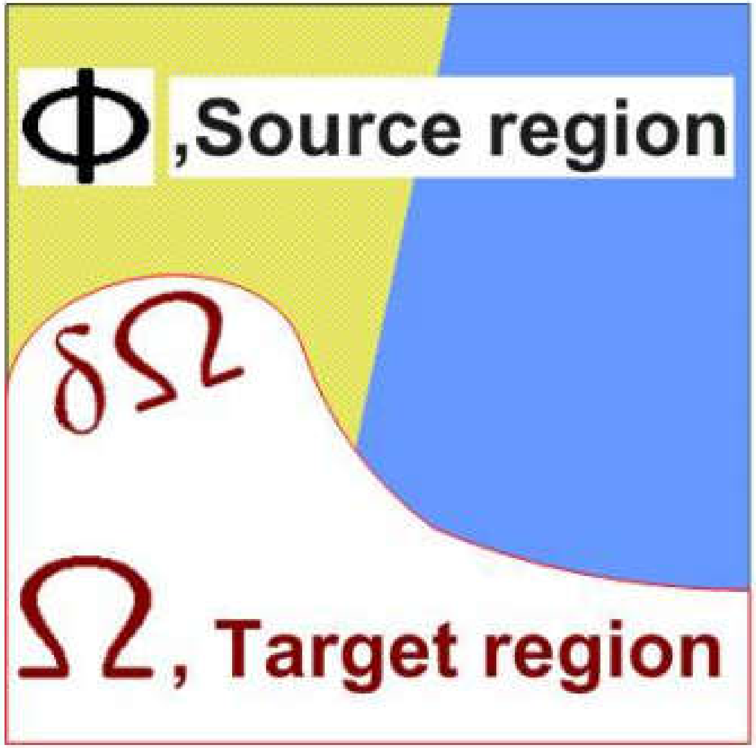

Bicubic
~ .75 secs

Inpainting

Inpainting


what a beautiful underwater image what a beautiful
underwater image what a beautiful underwater
image what a beautiful underwater image what a
beautiful underwater image what a beautiful under
water image what a beautiful underwater image
what a beautiful underwater image what a beautiful
underwater image what a beautiful underwater
image what a beautiful underwater image what a
beautiful underwater image what a beautiful under
water image what a beautiful underwater image
what a beautiful underwater image what a beautiful
underwater image what a beautiful underwater
image what a beautiful underwater image what a
beautiful underwater image what a beautiful under
water image what a beautiful underwater image
image what a beautiful underwater image what a

Inpainting



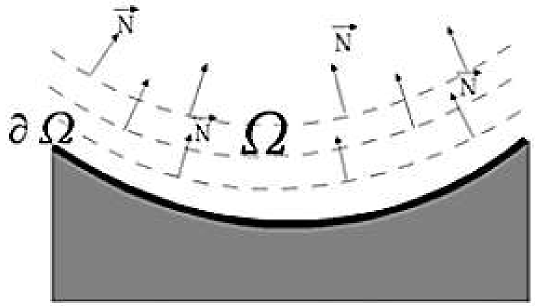
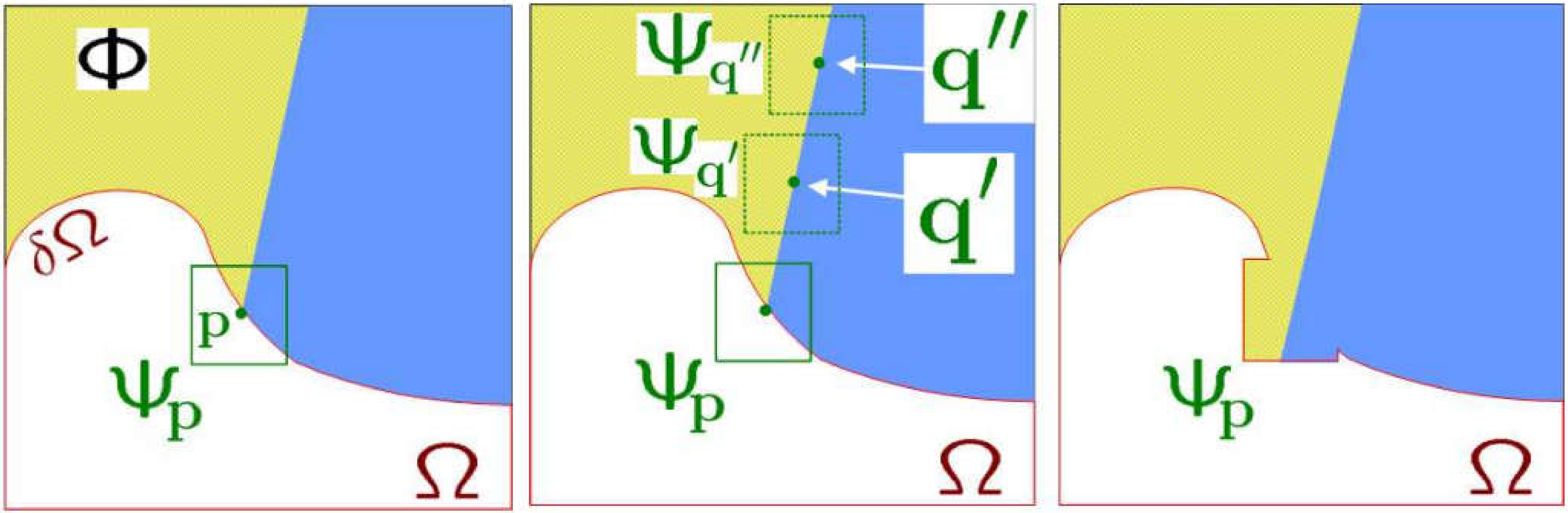


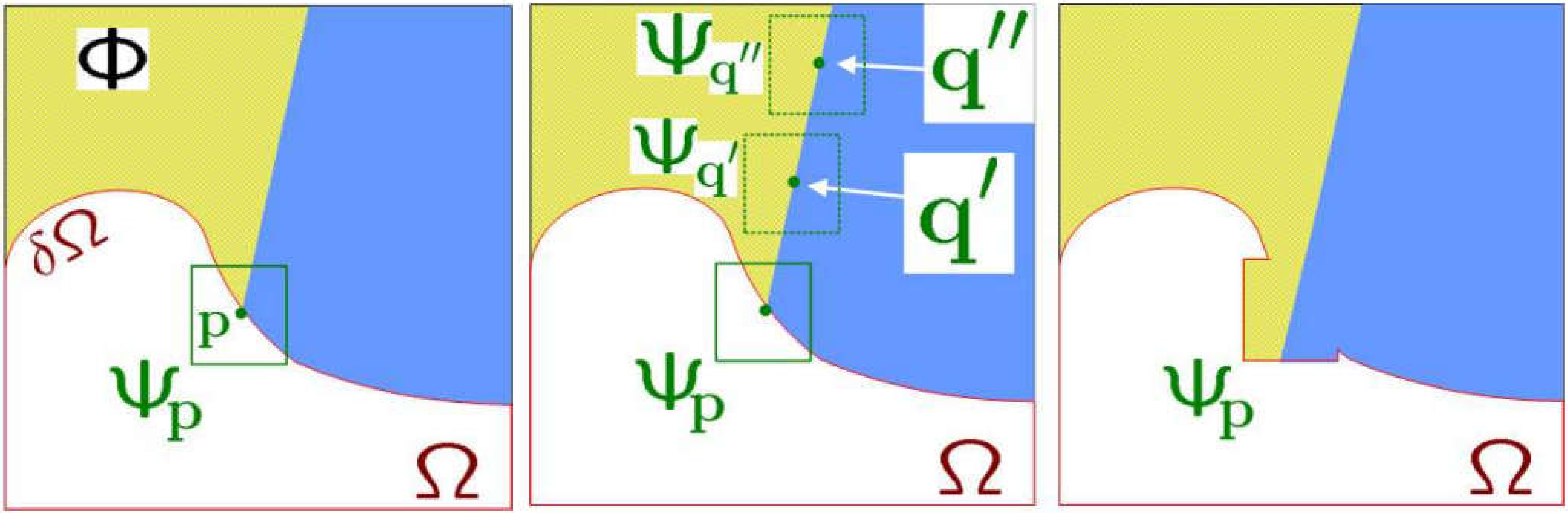
Figure 1: Propagation direction as the normal to the signed distance to the boundary of the region to be inpainted.

Figure 2: Unsuccessful choice of the information propagation direction. Left: detail of the original image, region to be inpainted is in white. Right: restoration.

<u>**Inpainting**</u>



Inpainting



Priority matters!

Inpainting

Region Filling and Object Removal by Exemplar-Based Image Inpainting, Criminisi, Perez and Toyama, IEEE TIP, 2004

Assignment 1

- Implement Criminisi et al. (2004) paper in group of four students.
- Implement using MATLAB
- Submit Report, working code(s) with images
- Do not use absolute file path in your code.
- **DO NOT USE MEX FILES or .p FILES**
- **DO NOT COPY!!**
- Add ‘Read Me’ file if required.

- **Submission Deadline: 17th October 2020, 5 PM.**
- **Submission Portal: Will be communicated through mail**

Report

- Introduction, a very brief literature survey about the problem, the algorithm that you are implementing, results and discussion
- Must be a PDF file
- Preferably in LATEX

  https://www.overleaf.com/learn/latex/Learn_LaTeX_in_30_minutes

  http://www.docs.is.ed.ac.uk/skills/documents/3722/3722-2014.pdf