MA3111: Mathematical Image Processing
Digital Image Fundamentals

Suh-Yuh Yang (楊肅煜)

Department of Mathematics, National Central University
Jhongli District, Taoyuan City 320317, Taiwan

First version: April 25, 2021/Last updated: September 21, 2024

Outline of "digital image fundamentals"

In this lecture, we will give a brief introduction to the topics:

- *The basic concepts of digital image processing.*
- *The basic image processing operations in* MATLAB, *including image reading, display, and storage back into the disk, etc.*

The material of this lecture is based on

- Chapter 1 and Chapter 2 in [GW2018].
- A series of four articles released by Dr. Anil Kumar Maini at `https://www.electronicsforu.com/?s=matlab+image+processing`

What is digital image processing?

The field of *digital image processing* refers to processing digital images by means of a digital computer.

- A grayscale image may be viewed as a real-valued function defined on a 2-D domain $\overline{\Omega} \subseteq \mathbb{R}^2$, i.e., $f : \overline{\Omega} \to \mathbb{R}_0^+$.

  In an image f, the value $f(x, y)$ is called the *intensity or gray level* of the image at point (x, y).

  A color image can be viewed as a vector-valued function $f : \overline{\Omega} \to (\mathbb{R}_0^+)^3$, e.g., $f(x, y) = (f_R(x, y), f_G(x, y), f_B(x, y))$.

- When x and y and intensity values $f(x, y)$ are all finite discrete quantities, the image is referred to as *a digital image (數位影像).*

  A digital image is composed of a finite number of elements, each of which has a particular location and value. These elements are called *image elements or pixels (像素).*

Digital image processing

- Interest in digital image processing methods stems from two principal application areas:

  (1) *improvement of pictorial information for human interpretation;*
  (2) *processing of image data for tasks such as storage, transmission, and extraction of pictorial information.*

  The need to extract information from images and interpret their content has been the driving factor in the development of modern image processing techniques and tools.

- Image processing is a multidisciplinary field. It overlaps with other areas such as *image analysis/understanding (影像分析/影像了解) and computer vision (電腦視覺).*

image processing $\Longrightarrow$ **image analysis/understanding**
$\Longrightarrow$ **computer vision**

Three types of image processing

This course focuses on the first two levels:

- **Low-level process:** operations such as image preprocessing to reduce noise, contrast enhancement, and image sharpening.

  Both its inputs and outputs are images.

- **Mid-level process:** tasks such as segmentation (partitioning an image into regions or objects), description of those objects to make them to a form suitable for computer processing, and classification (recognition) of individual objects.

  Its inputs generally are images, but its outputs are attributes extracted from those images (e.g., edges, contours, etc.).

- **High-level process:** "making sense" of an ensemble of recognized objects, performing the cognitive functions normally associated with human vision.

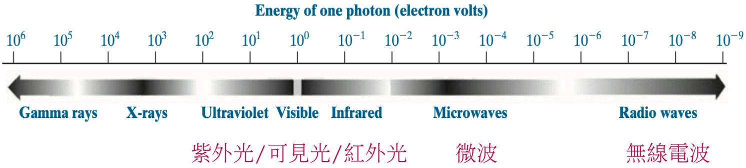
Applications of image processing

Image processing finds use in numerous fields:

- *Medicine:* digital radiography (數位X線攝影), positron emission tomography (PET, 正電子放射斷層掃描), computerized axial tomography (CAT/CT, 電腦軸向斷層掃描), magnetic resonance imaging (MRI, 核磁共振成像), and functional magnetic resonance imaging (fMRI).
- *Industry:* safety systems, quality control and automated guided vehicle control.
- *Military:* detection of soldiers or vehicles, missile guidance, and object recognition and reconnaissance.
- *Consumer electronics:* digital cameras and camcorders, high-definition TVs, monitors, DVD players, personal video recorders, and cell phones.

Electromagnetic spectrum

Imaging machines cover almost the entire electromagnetic (EM) spectrum, ranging from gamma rays to radio waves.



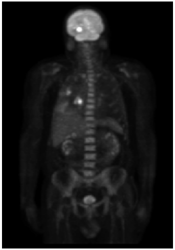
The electromagnetic spectrum arranged according to energy per photon

Gamma (γ)-ray imaging

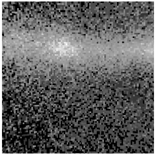
Major uses of imaging based on gamma rays include nuclear medicine and astronomical observations.

- *Nuclear medicine:* the approach is to inject a patient with a radioactive isotope that *emits γ rays as it decays.* Images are produced from the emissions collected by γ-ray detectors.
- *Positron emission tomography (PET):* Patient is given a radioactive isotope that *emits positrons as it decays.* When a positron meets an electron, both are annihilated and two γ rays are given off.

emits γ rays



PET



Cygnus loop(天鵝座星環)

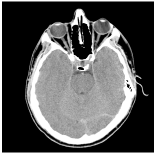
X-ray imaging: W. C. Roentgen (倫琴1895)

X-ray is used in medical diagnostics, industry and astronomy.

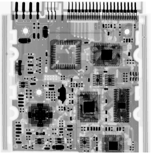
- *The intensity of the X-rays* is modified by absorption as they pass through the patient, placing between an X-ray source and a film sensitive to X-ray energy.
- *Angiography (contrast enhancement radiography):* The catheter is threaded into the blood vessel and guided to the area to be studied. When the catheter reaches the site under investigation, an X-ray contrast medium is injected through the tube.
- *Computerized axial tomography (CAT/CT):* The ensemble of such images constitutes a 3-D rendition of the inside of the body.

chest

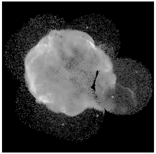
aortic angiogram (主動脈血管造影)



head CT



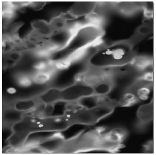
circuit

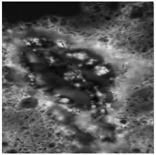


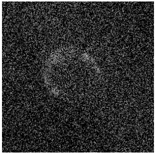
Cygnus loop

Imaging in the ultraviolet band (紫外光頻帶成像)

- *Applications of ultraviolet light:* include lithography, industrial inspection, microscopy, lasers, biological imaging, and astronomical observations.
- *Fluorescence microscopy (螢光顯微鏡):* The ultraviolet light itself is not visible, but when a photon of ultraviolet radiation collides with an electron in an atom of a fluorescent material, it elevates the electron to a higher energy level. Subsequently, the excited electron relaxes to a lower level and emits light in the form of a lower-energy photon in the visible (red) light region.

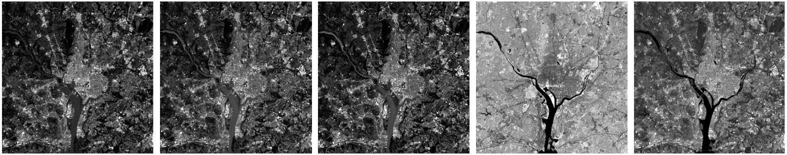

normal corn


corn smut

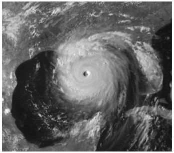

Cygnus loop(天鵝座星環)

Imaging in the visible and infrared bands

- *Remote sensing:* The primary function of LANDSAT is to obtain and transmit images *(thematic bands)* of the Earth from space, for purposes of monitoring environmental conditions.



- *Weather observation and prediction:* are also major applications of multispectral imaging from satellites.
- *Infrared imaging:* the Nighttime Lights of the World data set, which provides a global inventory of human settlements.



Imaging in the microwave band

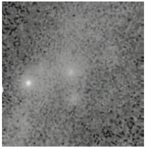
- *The principal application of imaging in the microwave band is radar.*
- The unique feature of imaging radar is its ability to collect data over virtually any region at any time, regardless of weather or ambient lighting conditions.

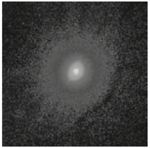
Spaceborne radar image of mountainous region in southeast Tibet

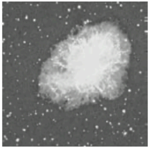
Imaging in the radio band

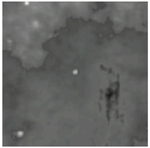
- *Magnetic resonance imaging (MRI):* This technique places a patient in a powerful magnet and passes radio waves through the individual's body in short pulses. Each pulse causes a responding pulse of radio waves to be emitted by the patient's tissues. MRI can produce images in any plane.

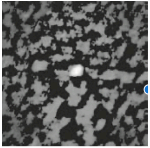

- *Astronomy:* Images of the Crab Pulsar (in the center of each image) covering the electromagnetic spectrum.


Gamma


X-ray

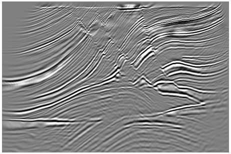

Optical

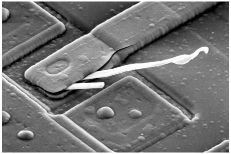

Infrared


Radio

Other imaging modalities

- **Acoustic imaging:** imaging using "sound" finds application in geological exploration *(hundreds of Hz),* industry, and medicine *(ultrasound, millions of Hz).*
    (1) The most important commercial applications of image processing in geology are in mineral and oil exploration.
    (2) *Ultrasound system:* a computer, ultrasound probe consisting of a source, a receiver, and a display, transmiting high-frequency (1 to 5 MHz) sound pulses into the body.
- **Electron microscopes:** function as their optical counterparts, except that they use a focused beam of electrons instead of light to image a specimen. EMs are capable of high magnification ($10000\times$ or more), light microscopy is limited to $1000\times$.





MATLAB **toolbox for image processing**

- MATLAB: *(matrix laboratory)* is a platform for solving scientific computing problems. It is a proprietary programming language developed by *MathWorks (Cleve Moler).*
- *Image Processing Toolbox* is a collection of functions that extends the capability of the MATLAB numeric computing environment for image processing, analysis, visualization and algorithm development.
- MATLAB *IPT* can be used to perform image segmentation, image enhancement, noise reduction, geometric transformations, image registration, and 3D image processing operations, etc.

Grayscale and color images

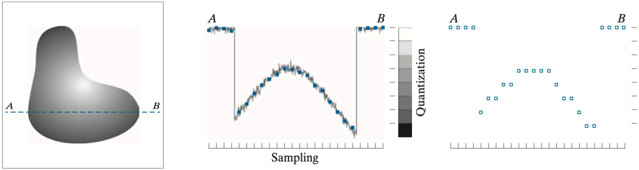
- A grayscale image is a real-valued function $f : \overline{\Omega} \to \mathbb{R}_0^+$, $f(x, y)$ is the *intensity* at the spatial coordinates (x, y).

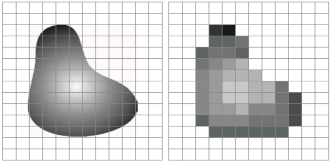
  A color image can be viewed as a vector-valued function $f : \overline{\Omega} \to (\mathbb{R}_0^+)^3$, e.g., $f(x, y) = (f_R(x, y), f_G(x, y), f_B(x, y))$.
- When x and y and the intensity values $f(x, y)$ are all finite discrete quantities, the image is referred to as *a digital image.*
- Digitizing the coordinate values is referred to as *sampling (取樣),* while digitizing the intensity values is called *quantization (量化).*
- *The result of sampling and quantization is a real matrix (矩陣) for a grayscale image and a real tensor (張量) for a color image.*

Sampling and quantization

To create a digital image, we need to convert the continuous sensed data into a digital format. This requires two processes: *sampling and quantization.*

- The method of sampling is determined by the sensor arrangement.
- Quantization of the sensor outputs completes the process.



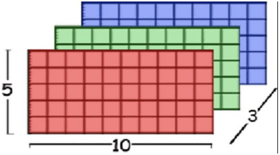


Digital images

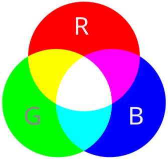
- A digitized grayscale image, *still denoted by f*, is represented as a matrix,

$$f = \begin{bmatrix} f_{11} & f_{12} & \cdots & f_{1N} \\ f_{21} & f_{22} & \cdots & f_{2N} \\ \vdots & \vdots & \cdots & \vdots \\ f_{M1} & f_{M2} & \cdots & f_{MN} \end{bmatrix}_{M\times N} \in (\mathbb{R}_0^+)^{M\times N}.$$

- A digitized color image is represented as a tensor, e.g.,



$\in (\mathbb{R}_0^+)^{5\times 10\times 3}$.



- Commonly used color spaces: RGB (red, green, blue), HSV (hue, saturation, value), etc.

Common types of digital images

- **Gray-level (monochrome) image:** encoded as a 2D array, with each pixel having 8 bits; 0 means black and 255 means white; intermediate values indicate varying shades of gray.
- **Binary image ("mask"):** represented as a 2D array; use 1 bit per pixel; 0 means black and 1 means white.
- **RGB image:** each color pixel is represented as (R, G, B); an RGB color image corresponds to a 3D array (a tensor) of dimensions $M \times N \times 3$, M height, N width, and 3 is the color components.

  For RGB images of class *double*, the range of values is $[0.0, 1.0]$; for *uint8*, range is $[0, 255]$; for *uint16*, range is $[0, 65535]$.

Basic image processing commands in MATLAB

- In MATLAB, a digital grayscale image f is indeed represented as a matrix and a color image is represented as a tensor.
- Images are read in MATLAB environment using `imread`.

  The `imread` function reads pixel values from an image file and returns a matrix of all pixel values.

grayscale and RGB images of penguins

Commands (functions) in MATLAB environment

- Read image named `filename.xxx` from the disk and then set as a variable (matrix) `G` in MATLAB:

  `>>G=imread('filename.xxx');`

- Display the variable `G` in an image form:

  `>>imshow(G)`

- Writes the variable `G` into the disk as an image in the new specified filename `newname.xxx` in the current directory:

  `>>imwrite(G,'newname.xxx');`

- **Example:** Read a RGB (color) image of penguins:

  `>>G=imread('Penguins_RGB.jpg');`

Examples

```
>>H=rgb2gray(G);
```
$\leftarrow 0.2989 * R + 0.5870 * G + 0.1140 * B$

```
>>imwrite(H, 'Penguins_gray.jpg');
>>F=imread('Penguins_gray.jpg');
>>imshow(F)
```

To convert a grayscale image to an RGB image is quite a hard problem! (image colorization)

```
>>size(F)        ← or [m,n] = size(F)
ans =     554  636
>>size(G)        ← or [m,n,p] = size(G)
ans =     554  636  3
>>whos F
Name  Size     Bytes   Class  Attributes
F     554x636  352344  uint8
>>whos G
Name  Size        Bytes    Class  Attributes
G     554x636x3   1057032  uint8
```

Image display

- Images are displayed on the MATLAB desktop using `imshow`.
- `imshow(F)`: `F` is an image array of data type uint8 or double.
  uint8: restricts the values of integers between 0 and 255. (1 byte)
  double: values between 0 and 1, 0 black and 1 white. (8 bytes)
  Any value between 0 and 1 is displayed as grayscale. Any value greater than 1 is displayed as white, and a value less than zero is displayed as black.
  uint8 $\leftrightarrow$ double: `B = im2double(A); A = im2uint8(B);`
- `imshow(F, [low high])`: displays as black if $F(m,n) \leq$ `low` and as white if $F(m,n) \geq$ `high`.

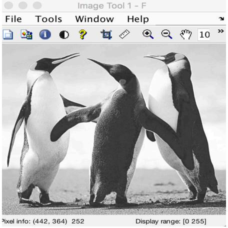
`imshow(F)`

`imshow(F, [50 100])`

Image display (cont'd)

The image tool in the image processing toolbox provides a more interactive environment for viewing and navigating within images, displaying detailed information about pixel values, measuring distances and other useful operations.

```
>>F=imread('Penguins_gray.jpg');
>>imtool(F)
```



The status text at the bottom of the main window shows the column/row location and the value of the pixel lying under the mouse cursor.

Image display (cont'd)

Multiple images can be displayed within one figure using `subplot`.

```
>>A=imread('Penguins_gray.jpg');
>>B=imread('Penguins_RGB.jpg');
>>figure
>>subplot(1,2,1),imshow(A)
>>subplot(1,2,2),imshow(B)
```

Writing images

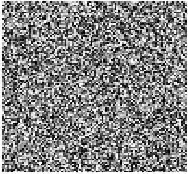
Images are written to the current directory using `imwrite`.

```
>>imwrite(f,'filename');
```

Writes image data `f` to the file `'filename'` in current folder.

Supports graphic file formats including `gif`, `hdf`, `jpeg`, `jpg`, `pbm`, `bmp`, `pgm`, `png`, `pnm`, `ppm`, `tif`, `tiff`, etc.

```
>>F=rand(100,100); % 100x100 matrix with values in [0,1]
>>imwrite(F,'random.png');
>>FF=imread('random.png');
>>imshow(FF)
```



Writing images (cont'd)

```
>>F=imread('Penguins_gray.jpg');
```

% save it in jpg format with quality parameter 10 (poor quality)

```
>>imwrite(F,'Penguins_gray_10.jpg','quality',10);
```

% save it in jpg format with quality parameter 75 (default)

```
>>imwrite(F,'Penguins_gray_75.jpg','quality',75);
```

% save it in jpg format with quality parameter 90 (high quality)

```
>>imwrite(F,'Penguins_gray_90.jpg','quality',90);
```

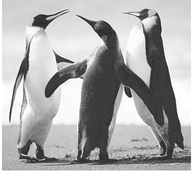
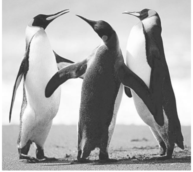




Image information

```
>>imfinfo('Penguins_gray_75.jpg')
ans=
    ⋮
    FileSize:  49327
    Format:  'jpg'
    Width:  636
    Height:  554
    ColorType:  'grayscale'
    ⋮
```

```
>>K=imfinfo('Penguins_gray_75.jpg');
>>[K.FileSize,K.Height,K.Width]
ans=
    49327  554  636
```

Retinex theory: retina + cortex

The classic Retinex theory models the *human color perception* (E. Land, Scientific American 1977). It assumes that the observed image f can be decomposed into two components:

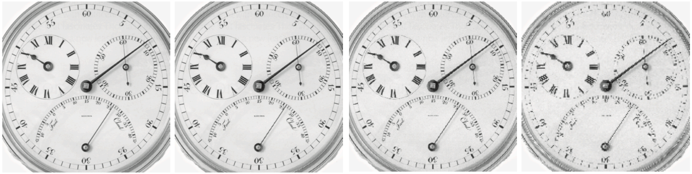
$$f(x,y) = I(x,y) \circ R(x,y),$$

where I represents illumination (照明), R represents reflectance (反射), and $\circ$ represents element-wise multiplication.

- The nature of I is determined by the illumination source and R is determined by the characteristics of the imaged objects.
- *The illumination component is commonly assumed to be smooth or piecewise smooth and retain main structures, while the reflectance component is expected to contain rich image detailed structures.*
- The theory is applicable to images formed via transmission of the illumination through a medium, such as a chest X-ray.

Spatial resolution and intensity resolution

- **Spatial resolution:** *dots/pixels per unit distance.* Dots per unit distance is a measure of image resolution used in the printing and publishing industry. To be meaningful, measures of spatial resolution must be stated with respect to spatial units. *In the U.S., this measure usually is expressed as dots per inch (dpi).*



930 dpi *300 dpi* *150 dpi* *72 dpi*

- **Intensity resolution:** *The number of bits used to quantize intensity.* For example, it is common to say that an image whose intensity is quantized into 256 levels has 8 bits of intensity resolution.

Image quality

Image quality is defined in terms of *spatial resolution and quantization.*

- *Spatial resolution is the pixel density over the image, which is expressed qualitatively as pixels/dots per inch (ppi/dpi). The greater the spatial resolution, the more are the pixels used to display the image.*
- `imresize(x,1/2)`: halves image size, choosing even indices.

  `imresize(x,2)`: all the pixels are repeated to produce an image of the double size of the original, but with half the resolution in each direction.

  `>>A=imread('Penguins_RGB.jpg');` % size: $554 \times 636 \times 3$

  `>>B=imresize(A,1/2);` % size: $277 \times 318 \times 3$

  `>>C=imresize(A,2);` % size: $1108 \times 1272 \times 3$

Image A  *Image B*  *Image C* 

Change the resolution

The following set of commands reduces the resolution of image:

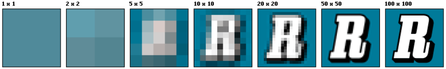
```
>>A1=imresize((imresize(A,1/2)),2);
>>A2=imresize((imresize(A,1/4)),4);
>>A3=imresize((imresize(A,1/8)),8);
```

image A1

image A2

image A3

Image A1: has half the image resolution but the same image size
Image A2: has one-fourth the image resolution but the same image size
Image A3: has one-eighth the image resolution but the same image size



Resolution illustration (Wikipedia)

Image quantization

Image quantization can be described as a mapping process by which groups of data points (several pixels within a range of gray values) are mapped to a single point (a single gray level).

```
>>A=imread('Penguins_gray.jpg');
>>B=grayslice(A,64);     % reduces the quantz. levels to 64
>>imshow(B,gray(64))
>>C=grayslice(A,8);     % reduces the quantz. levels to 8
>>imshow(C,gray(8))
```

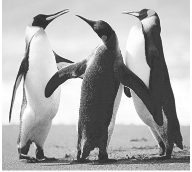


image A

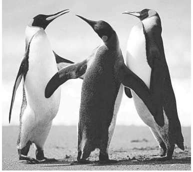


image B

image C

Image interpolation (影像插補)

- Interpolation is the technique of using known data to estimate values at unknown locations.
- Interpolation is used in image tasks such as resizing (zooming, shrinking), remapping (rotating, geometrically correcting digital images), inpainting (restoration of holes), morphing (nonlinear transformations).

$\longrightarrow$ $\longleftarrow$

Morphing: nonlinear transformations

Lagrange interpolation theorem

- **Lagrange interpolation:** *Let* $f : [a,b] \to \mathbb{R}$ *be a real-valued function and* $x_0, x_1, \cdots, x_n \in [a,b]$ *be* $n+1$ *distinct numbers. Then for* $n+1$ *values* $y_0 = f(x_0), y_1 = f(x_1), \cdots, y_n = f(x_n)$, *there exists a unique polynomial* p_n *of degree at most* n *such that*

$$p_n(x_i) = y_i \quad \forall\, 0 \le i \le n,$$

where

$$p_n(x) = y_0\ell_0(x) + y_1\ell_1(x) + \cdots + y_n\ell_n(x) \quad (\textit{Lagrange form})$$

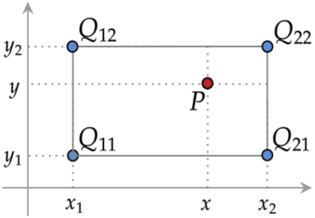
and

$$\ell_i(x) := \prod_{j=0, j\neq i}^{n} \frac{x - x_j}{x_i - x_j}, \quad 0 \le i \le n.$$

- **Interpolation error:** *Assume that* $f \in C^{n+1}[a,b]$. *Then for each* x *in* $[a,b]$ *there corresponds a point* $\xi_x \in (a,b)$ *such that*

$$f(x) - p_n(x) = \frac{1}{(n+1)!} f^{(n+1)}(\xi_x) \prod_{i=0}^{n} (x - x_i).$$

Bilinear interpolation



(cited from "omni calculator")

Let $Q_{11} = (x_1, y_1)$, $Q_{12} = (x_1, y_2)$, $Q_{21} = (x_2, y_1)$, and $Q_{22} = (x_2, y_2)$. Then by the Lagrange linear interpolation, we have

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

Bilinear interpolation (cont'd)

By the Lagrange linear interpolation again, we have

$$
\begin{aligned}
f(x,y) \approx p_{1,1}(x,y) &= \frac{y-y_2}{y_1-y_2} f(x,y_1) + \frac{y-y_1}{y_2-y_1} f(x,y_2) \\
&= \frac{y-y_2}{y_1-y_2}\left(\frac{x-x_2}{x_1-x_2} f(Q_{11}) + \frac{x-x_1}{x_2-x_1} f(Q_{21})\right) \\
&\quad + \frac{y-y_1}{y_2-y_1}\left(\frac{x-x_2}{x_1-x_2} f(Q_{12}) + \frac{x-x_1}{x_2-x_1} f(Q_{22})\right) \\
&= \frac{1}{(x_1-x_2)(y_1-y_2)}\Big(f(Q_{11})(x-x_2)(y-y_2) \\
&\quad + f(Q_{21})(x-x_1)(y_2-y) + f(Q_{12})(x_2-x)(y-y_1) \\
&\quad + f(Q_{22})(x-x_1)(y-y_1)\Big) \\
&= \frac{1}{(x_1-x_2)(y_1-y_2)} \begin{bmatrix} x_2-x \\ x-x_1 \end{bmatrix}^{\top} \begin{bmatrix} f(Q_{11}) & f(Q_{12}) \\ f(Q_{21}) & f(Q_{22}) \end{bmatrix} \begin{bmatrix} y_2-y \\ y-y_1 \end{bmatrix}.
\end{aligned}
$$

A direct approach: bilinear and bicubic interpolations

- For bilinear interpolation, a direct approach is given by

$$f(x,y) \approx p_{1,1}(x,y) = a + bx + cy + dxy,$$

where the four coefficients are determined from the four equations in four unknowns a, b, c, d:

$$\begin{aligned} f(Q_{11}) &= a + bx_1 + cy_1 + dx_1y_1, \\ f(Q_{12}) &= a + bx_1 + cy_2 + dx_1y_2, \\ f(Q_{21}) &= a + bx_2 + cy_1 + dx_2y_1, \\ f(Q_{22}) &= a + bx_2 + cy_2 + dx_2y_2. \end{aligned}$$

- For bicubic interpolation, a direct approach is given by

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

where the 16 coefficients are determined from the 16 equations with 16 unknowns, using the function values of the 16 nearest neighboring points.

Interpolation for image resizing

Below we apply the interpolation technique to image resizing (shrinking and zooming), which are basically *image resampling (影像重新取樣) methods.*

Suppose that an image of size 500×500 *pixels has to be enlarged* 1.5 *times to* 750×750 *pixels. A simple way is*

(1) To create an imaginary 750×750 grid with the same pixel spacing as the original image, then shrink it so that it exactly overlays the original image.

(2) The pixel spacing in the shrunken 750×750 grid will be less than the pixel spacing in the original image.

(3) To assign an intensity value to any point in the overlay, we look for its closest pixel in the underlying original image and assign the intensity of that pixel to the new pixel in the 750×750 grid. (nearest neighbor interpolation)

(4) Expand it back to the specified size to obtain the resized image.

Step (3) can be replaced by the bilinear or bicubic interpolation!

A comparison

- *Nearest neighbor interpolation* is simple but, it has the tendency to produce undesirable artifacts, e.g., severe distortion of straight edges.
- *Bilinear interpolation* gives much better results than nearest neighbor interpolation, with a modest increase in computational burden.
- *Bicubic interpolation* does a better job of preserving fine detail. Bicubic interpolation is the standard used in commercial image editing applications, such as "Adobe Photoshop."

Interpolation from 165 × 166 *pixels to* 2136 × 2140 *pixels.*
(left) nearest neighbor; (middle) bilinear; (right) bicubic