الوسائط المتعددة و برمجتها

السنة الثالثة

قسم تقنيات الحاسوب

المحاضرة الرابعة

إعداد

م.يوسف دعبول

Chapter 2

Digital Image Fundamentals

Image Operations on a Pixel Basis

Numerous references are made to operations between images, such as dividing one image by another.

In some discussions, it is advantageous to use a more traditional matrix notation to denote a digital image and its elements:

$$\mathbf{A} = \begin{bmatrix} a_{0,0} & a_{0,1} & \cdots & a_{0,N-1} \\ a_{1,0} & a_{1,1} & \cdots & a_{1,N-1} \\ \vdots & \vdots & & \vdots \\ a_{M-1,0} & a_{M-1,1} & \cdots & a_{M-1,N-1} \end{bmatrix}. \tag{2.4-2}$$

In Eq. (2.4-2), images were represented in the form of matrices

when we refer to an operation like "dividing one image by another," we mean specifically that the division is carried out between *corresponding* pixels in the two images.

Image Operations on a Pixel Basis

Thus, for example, if f and g are images, the first element of the image formed by "dividing" f by g is simply the first pixel in f divided by the first pixel in g; of course, the assumption is that none of the pixels in g have value 0.

Other arithmetic and logic operations are similarly defined between corresponding pixels in the images involved.

Image Combination

Arithmetic combination is applied on a pixel- by-pixel basis.

– The two images must have comparable dimensions.

– If not then image1 (w1× w2), image2 (h1× h2)

– The new image will have dimensions w × h

$w = \min (w_1 , w_2)$

$h = \min (h_1 , h_2)$

•Two types of image combination:

•*arithmetic (image math)* → grayscale images

•*Logical (Boolean)* → binary images

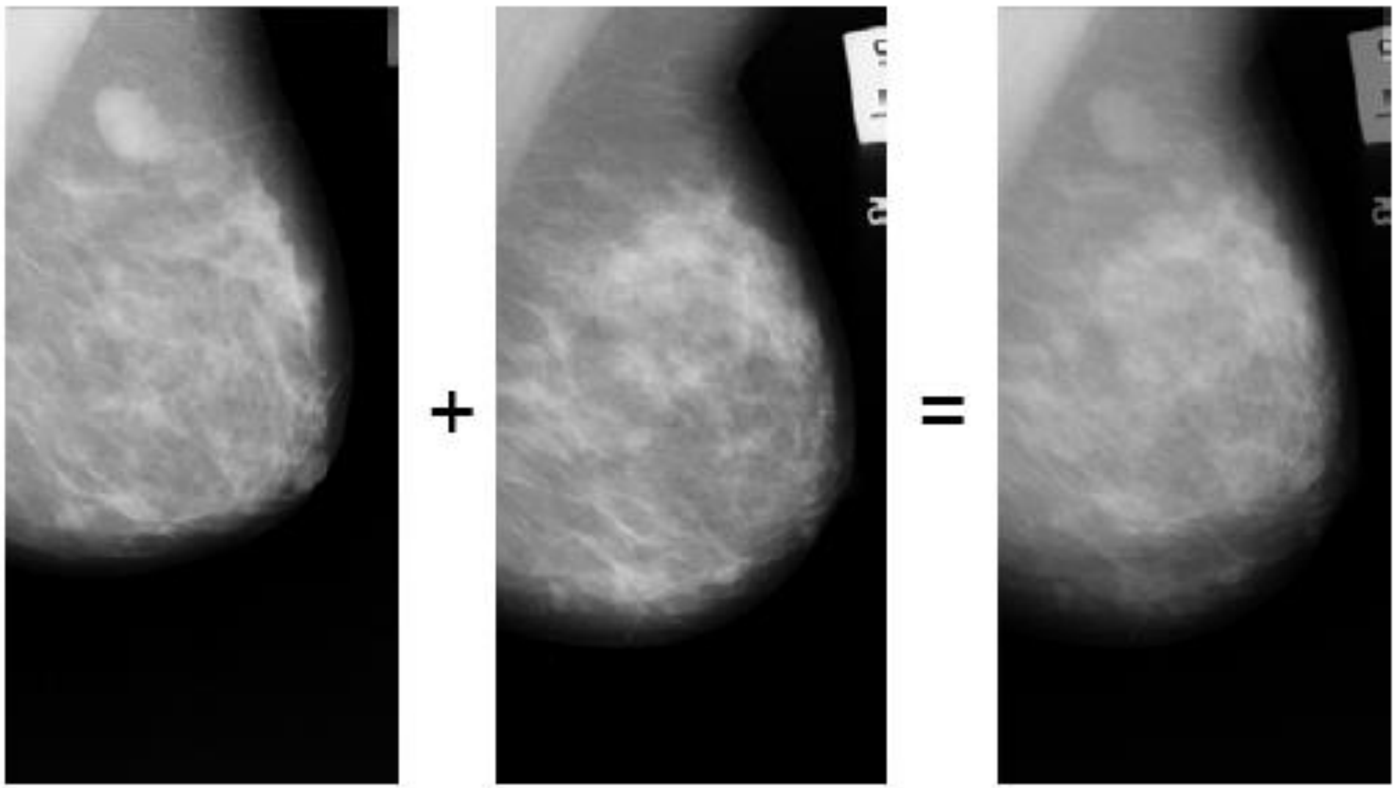
Image Addition

- Image addition superimposes information:
  - •Pixels in the resulting image have values in the range 0-510.
  - •Normalize the resulting image.divided by two (image averaging or converted to 16-bit.)
  - •Primarily used for noise removal "Alpha blending".
  - •Give more emphasis to one image than the other

$$g(x,y) = \alpha f_1(x,y) + (1-\alpha) f_2(x,y)$$

  - • When α =0 .5, g(x,y) becomes a simple, even-weighted average.
  - •Every pixel can have its own α stored in a separate α-channel.

Image Addition



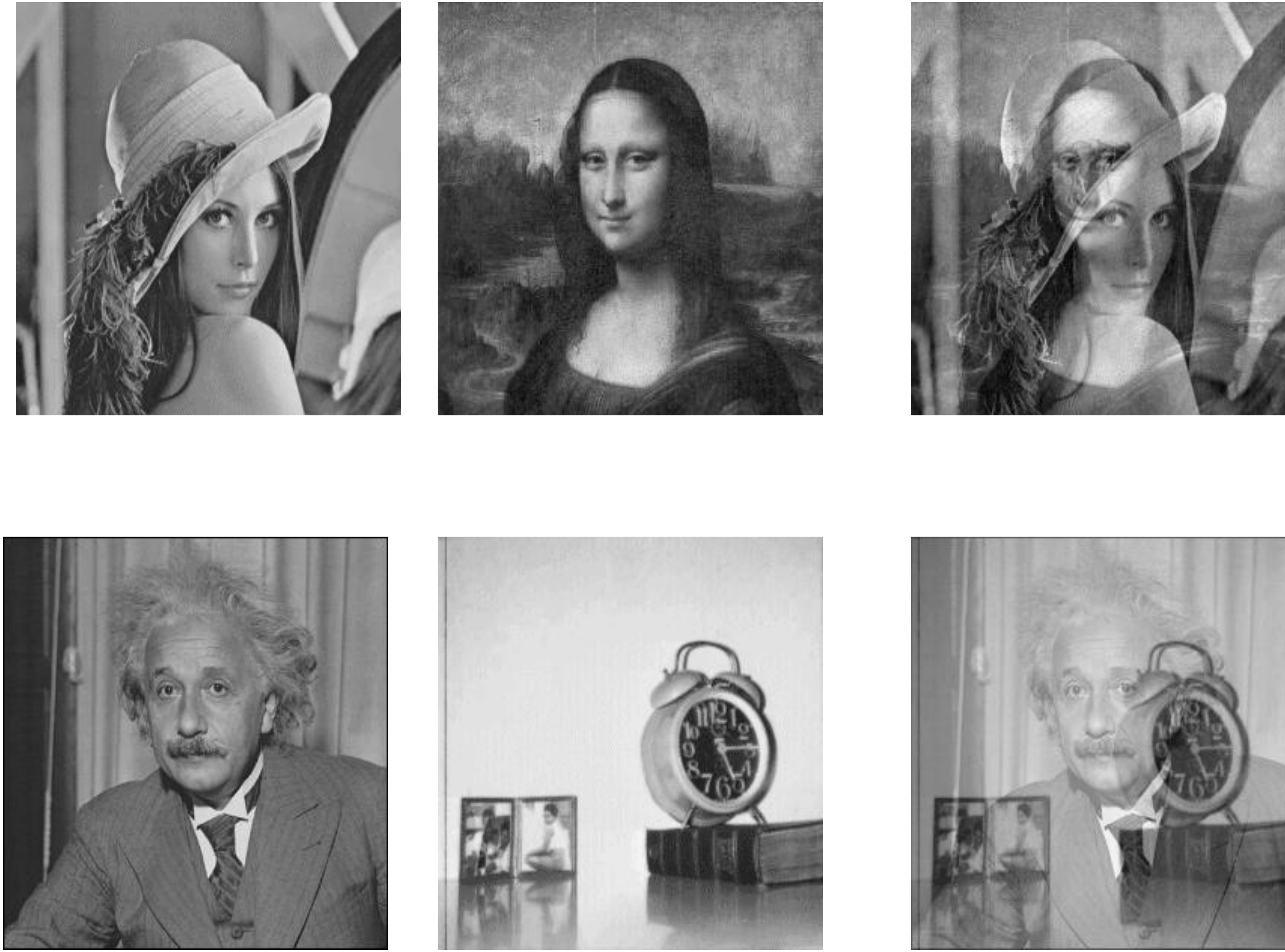


Image Subtraction

- Image subtraction calculates the differences between images
  - Used primarily for change detection
- Pixels in the resulting image have values in the range –255 to +255

$$g(x, y) = |f_1(x, y) - f_2(x, y)|$$

- Changes will be indicated by pixels in the difference image which have non-zero values.
  - The difference image will contain only features that change

Image Subtraction

- Sensor noise, slight intensity changes, and various other factors result in small differences which are of no significance.
- It is usual to apply a threshold to the difference image.
- Object motion can be measures through subtraction

  e.g. track the motion of cells in response to chemical cues.

Image Subtraction

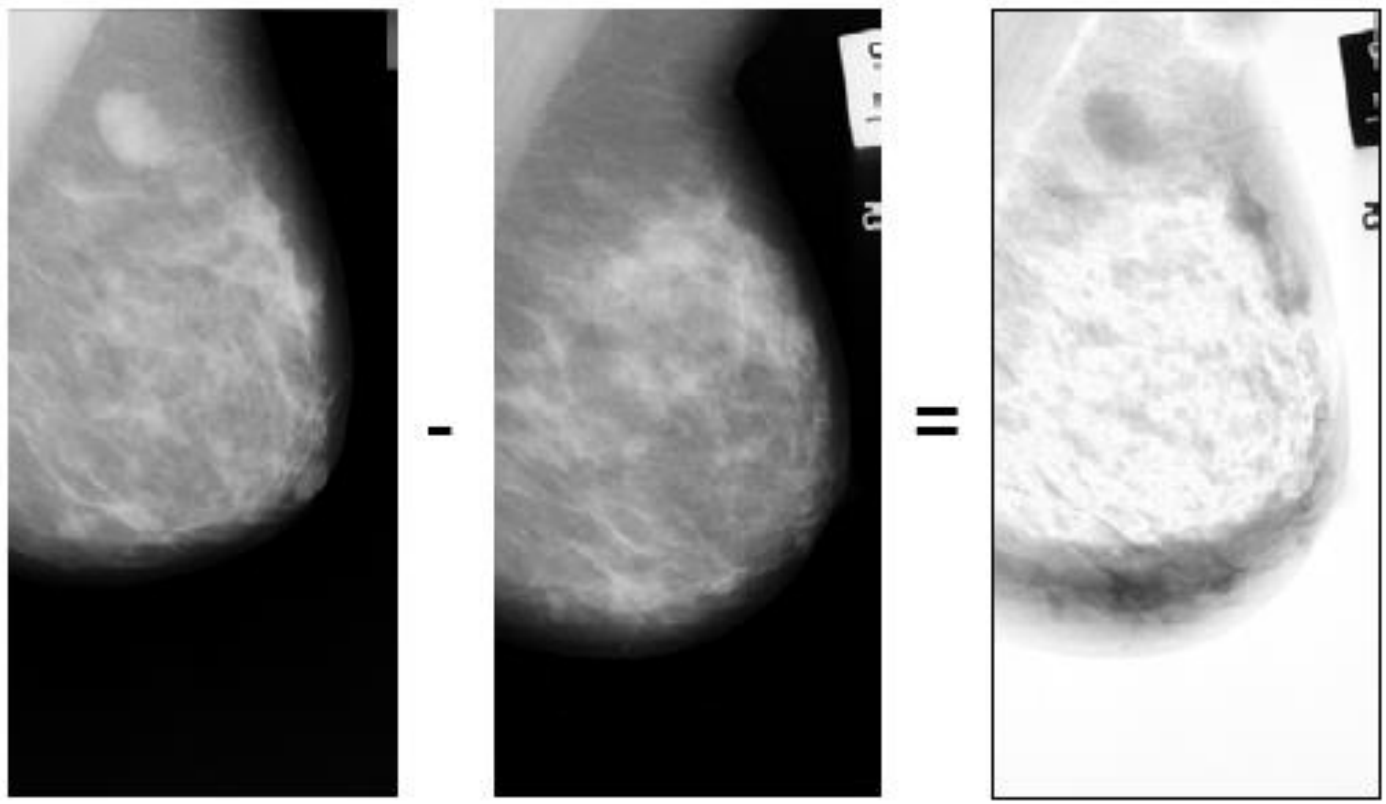


Image Division

- Image division is used for removing backgrounds when linear detectors or cameras are used.
  - For meaningful results use floating-point arithmetic
  - Produces a ratio image in which the pixels should be rescaled and rounded → normalise

$$g(x,y) = f_1(x,y)/f_2(x,y)$$

  - Pixels of 0 intensity are removed from f_2, adding a constant of unity to produce

$$f_2 \equiv f_2 + C, \quad C = 1$$

Image Division

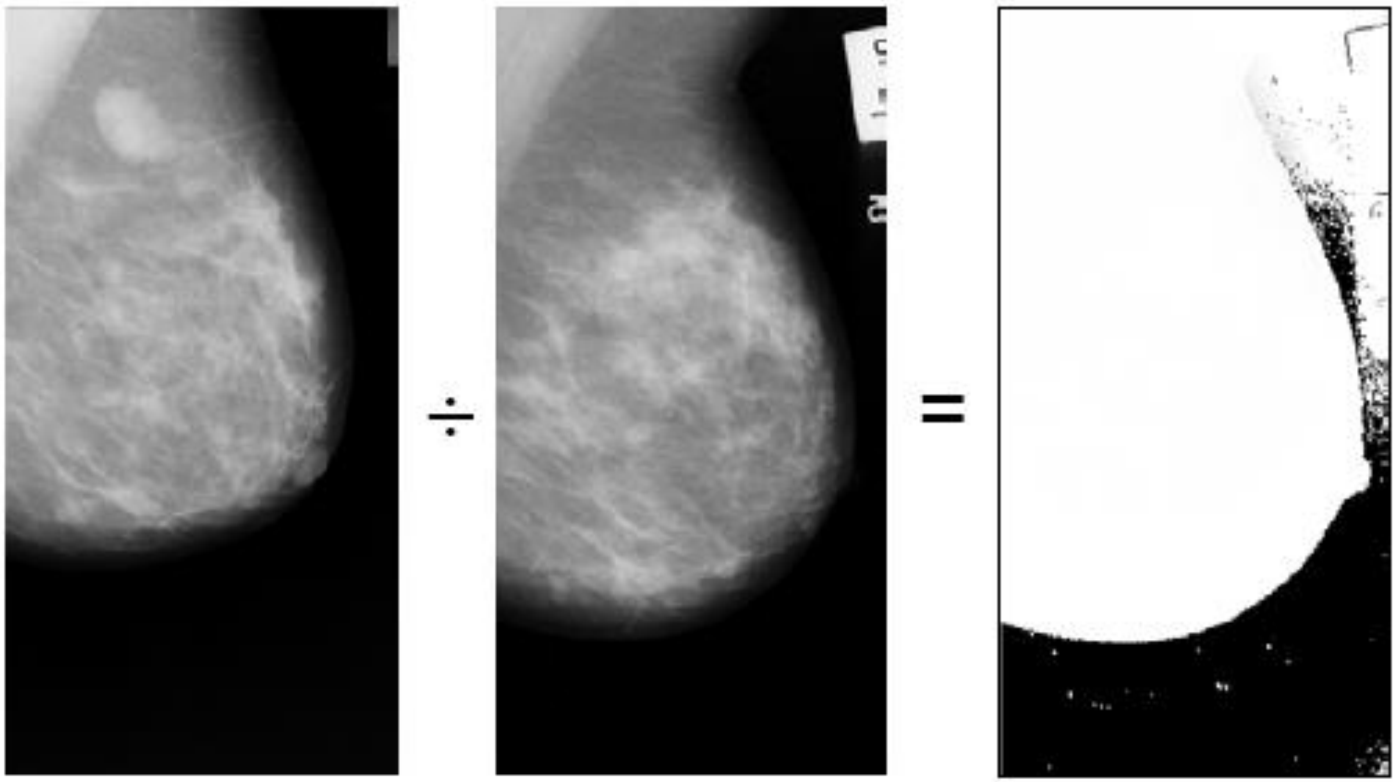


Image Multiplication

- Image multiplication is used for superimposing information

$$g(x,y) = f_1(x,y) \times f_2(x,y)$$

  – Results in an extreme range of values: 0→255 becomes 0→ 65,000
  – Loss of precision in rescaling

  e.g. Combine edge and direction information from Sobel edge detection

  e.g. Add fluorescence or other emission images to a reflection or transmission image

Image Multiplication

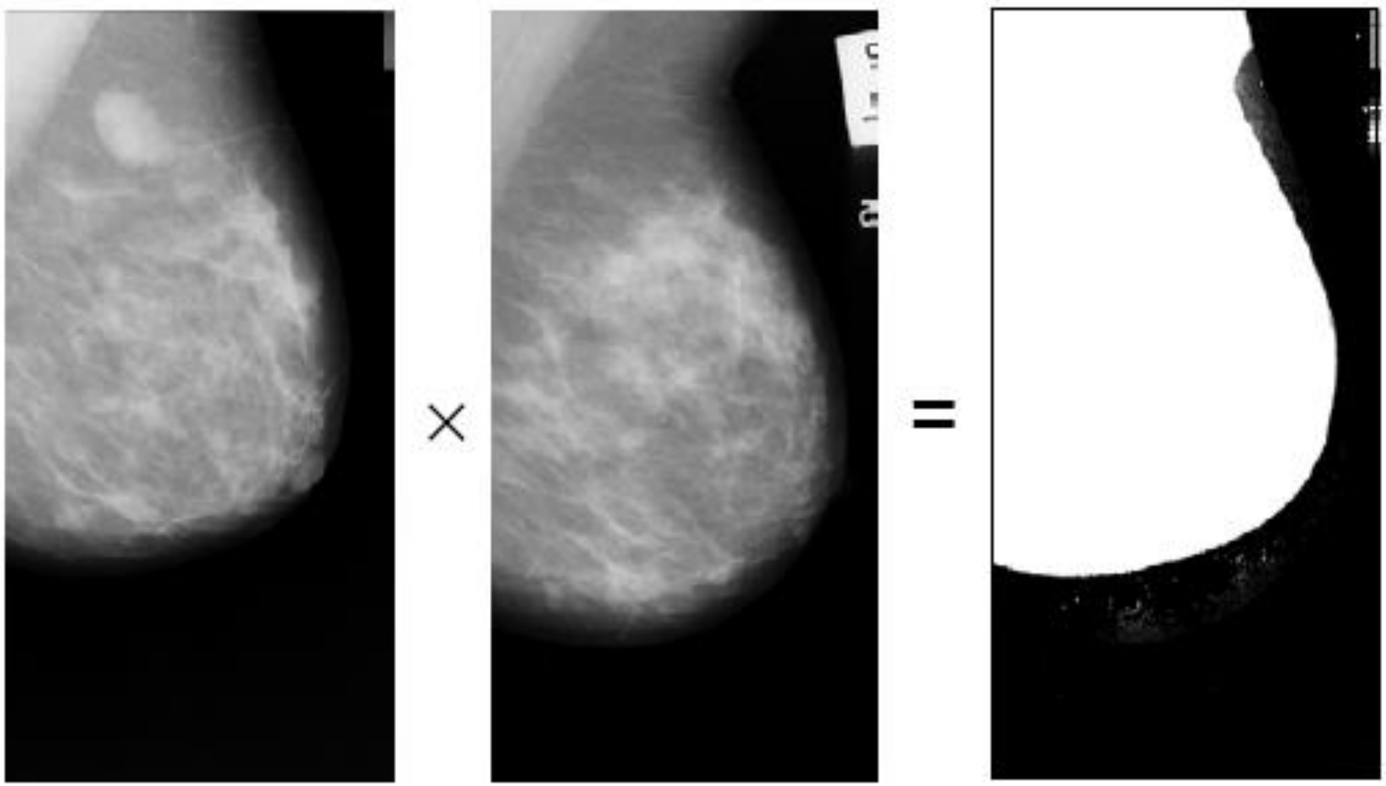


Image Minimum & Maximum

- Image combination using min (or max) involves retaining the darker (or lighter) intensity values at each location

$$g(x, y) = \min(f_1(x, y), f_2(x, y))$$

e.g. To build up a confocal scanning light microscope (CSLM) image with greater depth of field

Image Minimum

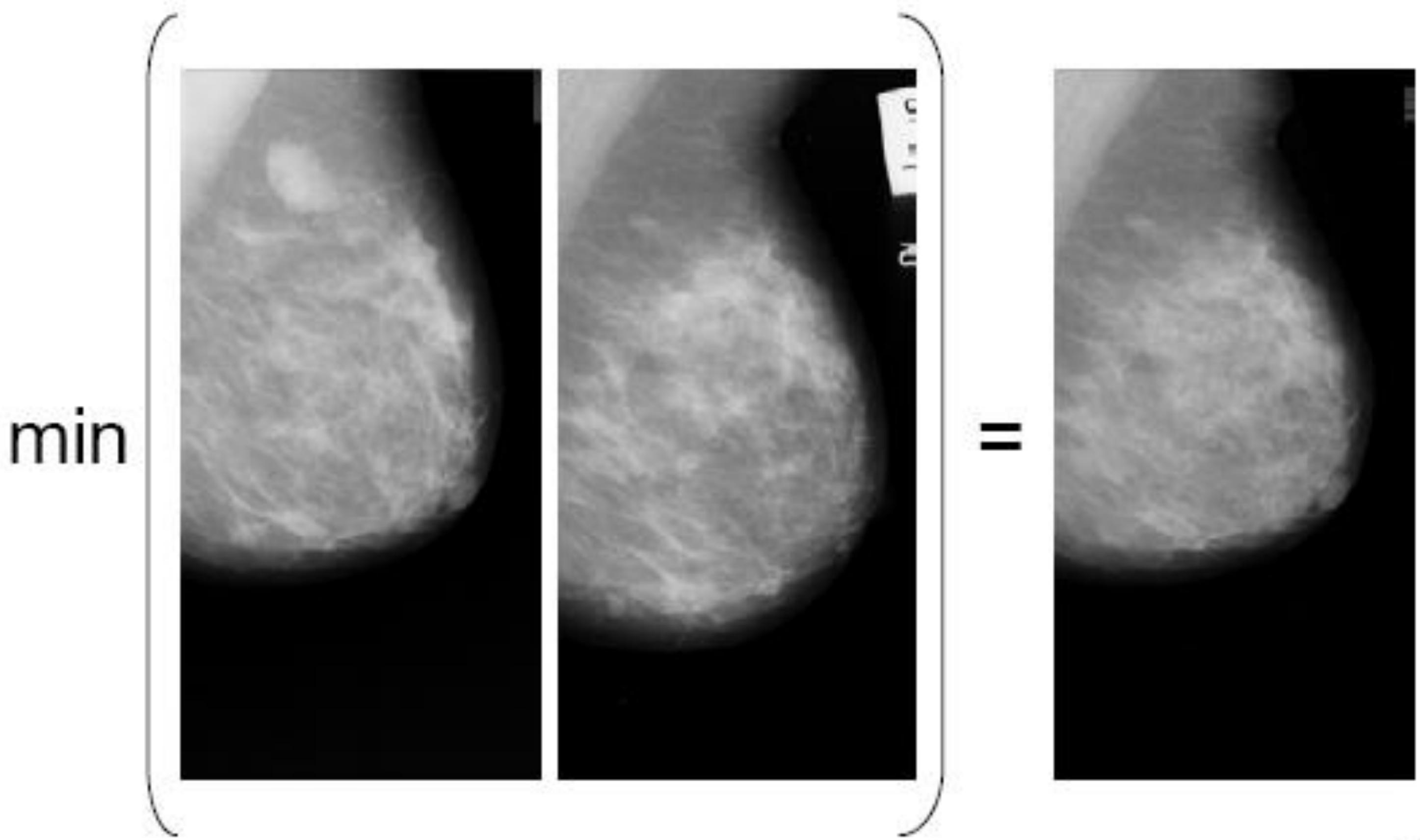
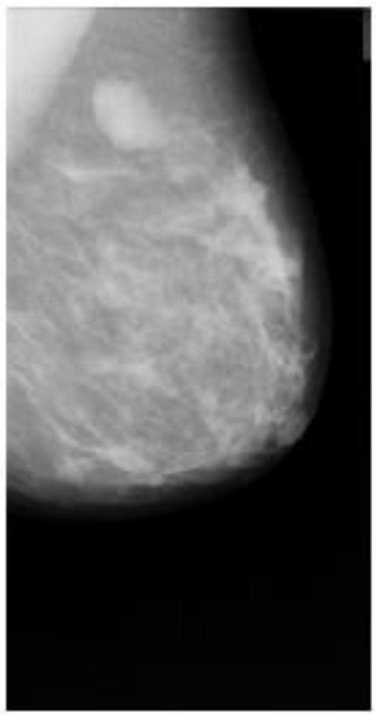
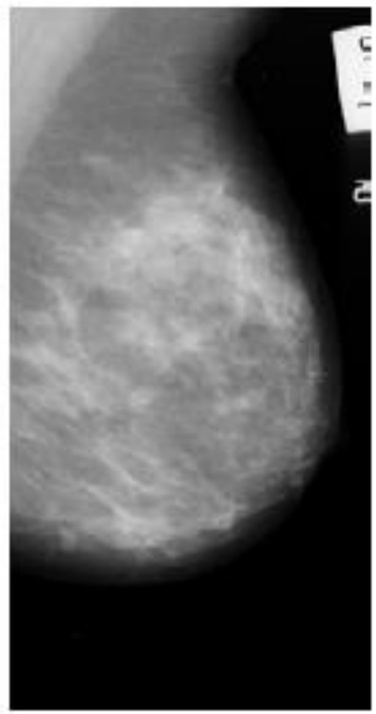
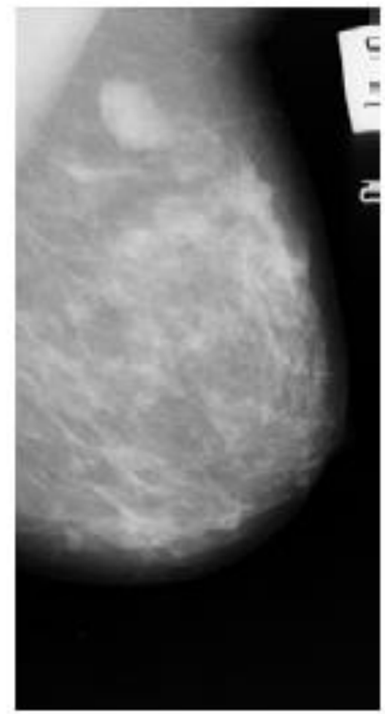
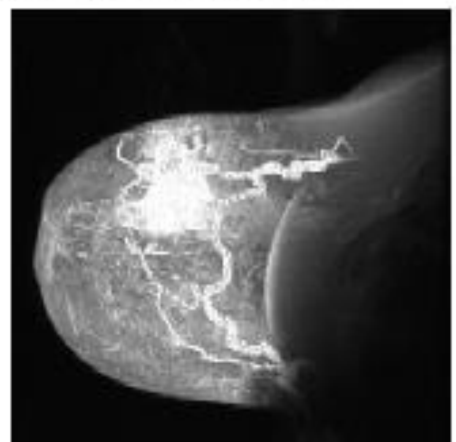


Image Maximum

max (


) =


Additional Effects

- Combining *n* images
  - Shifting each image slightly before performing combination, produces a perspective view of the surface.
  - Use the max function to combine images
    - e.g. maximum intensity projections



Logical Combination

- Logical or boolean combinations are usually applied to binary images

$$g(x,y) = f_1(x,y) \odot f_2(x,y)$$

- Operations
  - AND, OR, XOR, NOT

AND

- The pixel at location (x,y) is 1 if it is 1 in both images $f_1(x,y)$ and $f_2(x,y)$.
  - All pixels common to both images

$$g(x,y) = (f_1 \text{ AND } f_2) = 1$$
$$\text{if } f_1(x,y) = f_2(x,y) = 1$$

OR

- The pixel at location (x,y) is 1 if it is 1 in either of the images $f_1(x,y)$ or $f_2(x,y)$.

$$g(x,y) = (f_1 \text{ OR } f_2) = 1$$
$$\text{if } f_1(x,y) = 1 \text{ OR } f_2(x,y) = 1$$

XOR

- Exclusive-OR
- The pixel at location (x,y) is 1 if it is 1 in either of the images $f_1(x,y)$ or $f_2(x,y)$, but not if it is 1 in both.

$$g(x,y) = (f_1 \text{ XOR } f_2) = 1$$
$$\text{if } f_1(x,y) = 1 \text{ AND } f_2(x,y) = 0,$$
$$\text{or } f_1(x,y) = 0 \text{ AND } f_2(x,y) = 1$$