ELIC 629
Digital Image Processing

Winter 2006

Image Enhancement in the Spatial Domain:

Basic Gray Level Transformations: image negatives, log transformations, power and piecewise-linear transformations, histogram processing

Bill Kapralos

Tuesday, January 31 2006

Overview (1):

- **Before We Begin**
  - Administrative details
  - Review → some questions to consider
- **Linear and Non-Linear Operators**
  - Linear operators
  - Non-linear operators
- **Image Enhancement in the Spatial Domain**
  - Introduction → what is image enhancement ?
  - Background to processing in the spatial domain

Overview (2):

- **Basic Gray-Level Transformations**
  - Introduction

Before We Begin

Administrative Details (1):

- **Miscellaneous Notes**
  - Assignments and lab reports submitted late will result in a penalty
    - 10% per day
    - Not accepted after one week

Administrative Details (2):

- **Lab Three**
  - Will involve the use of a digital camera which I will distribute (along with appropriate equipment) during the lab
  - No lab report required for this lab but there is a lab

Some Questions to Consider (1):

- What is spatial sub-sampling (shrinking) ?
- What is spatial up-sampling (zooming) ?
- What is aliasing ?
- What is interpolation ?
- Define a pixel's 4 & 8-neighbours
- What is adjacency ?
- Define connected, region and boundary
- List and define the three distance measures discussed

Linear & Non-Linear Operators

Linear Operators (1):

- **Mathematical Definition**
  - For any two images "*f*" and "*g*", two scalars "a" and "b" and operator "H" (whose input and output is an image)

$$H(af + bg) = aH(f) + bH(g)$$

  - In words → Result of applying operator H to sum of two images that have been multiplied by some scalar is equal to applying the operator to each image separately, multiplying each image by the appropriate scalar and summing the results

Linear Operators (2):

- **Very Important to Digital Image Processing**
  - Based on well understood theoretical and practical results
  - Usually predictable
  - Can "break" an operator down to several "sub-operators", perform each sub-operator separately and then combine results
    - e.g., useful in networked environments or multi-threaded applications where we can "simultaneously" solve the sub-operators and then combine

Non-Linear Operators (1):

- **Definition**
  - Any operator which does not satisfy the linear operator constraint
  - Any operator that is not linear!
  - Can be more computationally efficient as opposed to linear operators
  - Not well understood theoretically
  - Example: $H \rightarrow$ absolute value of the difference of two images

$$H(f, g) = |f - g| \neq |H(f)| + |H(g)|$$

Non-Linear Operators (2):

- **Further Examples/Comments**
  - How about the following:
    - $H(f) = f^2 \rightarrow$ linear or non-linear ??
    - Can you think of any others ?
    - How do we show (prove) an operator is linear or non-linear ?

Introduction: Image Enhancement in the Spatial Domain

What is Image Enhancement ? (1):

- **Purpose**
  - Process an image so that the resulting image is more suitable than the original for a specific application
  - Important → application specific
    - Application (needs) determine the process applied to the image (or portion of an image)
  - One of the most interesting and visually appealing areas of image processing
    - You get to see the results, typically immediately!

What is Image Enhancement ? (2):

- **Two Broad Categories of Image Enhancement**
  - Spatial domain processing
    - Processing applied to the image plane itself e.g., direct manipulation of the pixel intensity values
  - Frequency domain processing
    - Processing applied to the Fourier spectrum of the image
    - Need to "compute" the Fourier transform of spatial sdomain representation first
    - Once processing done, convert Fourier representation back to spatial domain

What is Image Enhancement ? (3):

- **Application (Viewer) Based**
  - No general method (steps to follow) to image enhancement
    - Dependant on the application
    - Dependant on the viewer who is ultimately the end "user" of the processed image → subjective and once again, my view of a good image may differ from yours!
    - Makes it very difficult to compare/judge image enhancement methods although one thing to consider is processing time

Spatial Processing – Background (1):

- **Denoting a Spatial Domain Process**
  - Denoted by the expression

$$g(x,y) = T[f(x,y)]$$

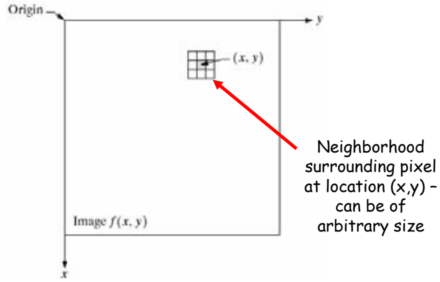
  - f(x,y) → input (original) image
  - g(x,y) → output (processed) image
  - T → a process (operator) on f defined over some neighborhood of spatial position (x,y) but can also operate on a set of images

Spatial Processing – Background (2):

- **Denoting a Spatial Domain Process (cont…)**
  - How do we define a neighborhood over (x,y) ?
    - Most popular approach → use a square or rectangular "sub-image" area (neighborhood) centered about pixel (x,y)
    - Process T is applied to pixels in sub-image only → e.g., g(x,y) is calculated based on the values of the pixels in this sub-image centered about (x,y)
    - Of course, not restricted to square or rectangular neighborhood → can use circular, elliptical etc. but much more complicated!

Spatial Processing – Background (3):

- **Denoting a Spatial Domain Process (cont…)**
  - Graphical illustration of neighborhood



Spatial Processing – Background (4):

- **Kernels or Masks or Templates or Windows**
  - Neighborhoods greater than 1x1
  - A small 2D array (matrix, sub-image, grid etc.)
  - The individual values of the mask (kernel etc.) are known as the mask coefficients
    - Determine the "nature" of the process (e.g., the process performed on the output is based on the value of the coefficients)
    - Mask processing or mask filtering → enhancement techniques based on operators that use a mask, kernel etc. (discussed later…)

Spatial Processing – Background (5):

- **Further Details Regarding Operator "T"**
  - Simplest form of operator T is one which uses a neighborhood of size 1 x 1 (single pixel) - in other words, output (intensity) at g(x,y) is determined by using the intensity at f(x,y) only
    - Known as a gray-level (or intensity or mapping) transformation function denoted by:

$$s = T(r)$$

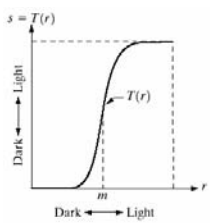
    - r → intensity at input pixel location (x,y)
    - s → intensity at output pixel location (x,y)

Spatial Processing – Background (6):

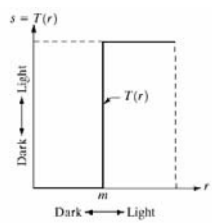
- **Example of a 1x1 Operator**
  - Contrast Stretching → output image with higher contrast intensity
    - Intensities of input image below "m" are compressed into narrow range of "s" toward black (e.g., darkened)
    - Intensities above "m" are brightened
    - Thresholding → In the limiting case, binary image results

Spatial Processing – Background (7):

- **Contrast Stretching**



Values below m are "compressed" into small region of output & values above m are stretched



Thresholding in the limiting case

Basic Gray-Level Transformations

Introduction (1):

- **Simplest Image Enhancement Techniques**
  - Recall → values of pixels before processing known as "r", after processing "s" and related by s = T(r)
  - Three common types of gray-level transformations
    1. Linear (negative and identity transformations)
    2. Logarithmic (log and inverse-log transformations)
    3. Power-law (n^{th} power & n^{th} root transformations)

Introduction (2):

- **Linear, Logarithmic & power Law Operators**