Computer Vision & Digital Image Processing

Intensity Transformations and Spatial Filtering

Intensity Transformations and Spatial Filtering Basics

- Operations take place in the spatial domain
  - Operate directly on pixel values
  - Often more computationally efficient and requires less resources
- General form for operations is:
  \[ g(x, y) = T[f(x, y)] \]
- Where \( f(x, y) \) is the input image, \( g(x, y) \) is an output image, \( T \) is an operator on \( f \) defined over a neighborhood of point \((x, y)\)

Intensity Transformations and Spatial Filtering Basics (continued)

- The operator can apply to a single image or to a set of images
- The point \((x, y)\) shown is an arbitrary point in the image
- The region containing the point is a neighborhood of \((x, y)\)
- Typically the neighborhood is rectangular, centered on \((x, y)\) and is much smaller than the image

Intensity Transformations and Spatial Filtering Basics (continued)

- Spatial filtering
  - Generally involves operations over the entire image
  - Operations take place involving pixels within a neighborhood of a point of interest \((x, y)\)
  - Also involves a predefined operation called a spatial filter
  - The spatial filter is also commonly referred to as:
    - Spatial mask
    - Kernel
    - Template
    - Window
Point Processing

- The smallest neighborhood of a pixel is 1×1 in size
- Here, \( g \) depends only on the value of \( f \) at \((x, y)\)
- \( T \) becomes an intensity transformation function of the form
  \[ s = T(r) \]
- where \( s \) and \( r \) represent the intensity of \( g \) and \( f \) at any point \((x, y)\)
- Also called a gray-level or mapping function

Intensity Transformation Example

Some Basic Intensity Transformation Functions

- Here, \( T \) is a transformation that maps a pixel value \( r \) into a pixel value \( s \)
- Since we are concerned with digital data, the transformation can generally be implemented with a simple lookup table
- Three basic types of transformations
  - Linear (negative and identity transformations)
  - Logarithmic (log and inverse-log transformations)
  - Power-law (nth power and nth root transformations)
**Image Negatives**

- The negative of an image with intensity levels in the range \([0, L-1]\) can be described by:

\[
s = L - 1 - r
\]

**Log Transformations**

- General form:

\[
s = c \log(1 + r)
\]

- \(c\) is a constant and \(r \geq 0\)
- Maps a narrow range of low intensity values in input to a wider output range
- The opposite is true for high intensity input values
- Compresses the dynamic range of images with large variations in pixel values

**Log Transformation Example**

- A plot (image) of the Fourier Spectrum is enhanced usually has a fairly large dynamic range
- The image can be enhanced by applying the log transformation

**Power-Law (Gamma) Transformations**

- Basic form

\[
s = cr^\gamma
\]

- Where \(c\) and \(\gamma\) are positive constants
- Power-law curves with fractional values of \(\gamma\) map a narrow range of dark input values to a wider range of output values
- The opposite is true for higher values of input levels
- There exists a family of possible transformation curves by varying \(\gamma\)
### Gamma Correction

- Many devices used for image capture, display and printing respond according to a power law.
- The exponent in the power-law equation is referred to as **gamma**.
- The process of correcting for the power-law response is referred to as **gamma correction**.
- Example:
  - CRT devices have an intensity-to-voltage response that is a power function (exponents typically range from 1.8-2.5).
  - Gamma correction in this case could be achieved by applying the transformation $s = r^{1/2.5} = r^{0.4}$.

### Gamma Correction Example

**FIGURE 3.7**

(a) Intensity ramp image. (b) Image as viewed on a simulated monitor with gamma of 2.5. (c) Gamma-corrected image. (d) Corrected image as viewed on the same monitor. Compare (d) and (a).

### Gamma Correction (MRI Example)

**FIGURE 3.8**

(A) Magnetic resonance image (MRI) of a human brain slice. (B) Results of applying the transformation $y = x^{1/2.5}$ with $\gamma = 0.5$, $0.4$, and $0.3$, respectively. (Original image courtesy of Dr. Donald R. Stetten, Department of Radiology and Radiological Sciences, Stanford University Medical Center.)
Gamma Correction (Aerial Example)

FIGURE 3.9
(a) Aerial image.  
(b)–(d) Results of 
applying the 
transformation in 
Eq. (3.2.3) with 
c = 1 and 
γ = 3.0, 4.0, and 
5.0, respectively. 
(Original image for 
this example 
courtesy of 
NASA.)

Piecewise-Linear Transformations

- Piecewise functions can be arbitrarily complex
- A disadvantage is that their specification requires significant user input
- Example functions
  - Contrast stretching
  - Intensity-level slicing
  - Bit-plane slicing

Contrast Stretching

- Contrast stretching expands the range of intensity levels in an image so it spans a given (full) intensity range
- Control points \((r_1, s_1)\) and \((r_2, s_2)\) control the shape of the transform \(T(r)\)
- \(r_1=r_2, s_1=0\) and \(s_2=L-1\) yields a thresholding function

Intensity-level Slicing

- Used to highlight a specific range of intensities in an image that might be of interest
- Two common approaches
  - Set all pixel values within a range of interest to one value (white) and all others to another value (black)
  - Brighten (or darken) pixel values in a range of interest and leave all others unchanged
**Bit-plane Slicing**

- Pixels are digital values composed of bits.
- For example, a pixel in a 256-level gray-scale image is comprised of 8 bits.
- We can highlight the contribution made to total image appearance by specific bits.
- For example, we can display an image that only shows the contribution of a specific bit plane.

---

**Bit-plane Slicing (Example)**

- Bit-plane slicing is useful in:
  - Determining the number of bits necessary to quantize an image.
  - Image compression.

---

**Intensity-level Slicing (Angiogram Example)**

![Intensity-level Slicing (Angiogram Example)](image)

**Bit-plane Slicing (continued)**

- Bit-plane slicing is useful in:
  - Determining the number of bits necessary to quantize an image.
  - Image compression.

![Bit-plane Slicing (continued)](image)