2. Digital Image Fundamentals

1. Elements of Visual Perception
2. Light and the Electromagnetic Spectrum
3. Image Sensing and Acquisition
4. Image Sampling and Quantization
5. Some Basic Relationships between Pixels
6. An Introduction to Mathematical Tools Used in Digital Image Processing
2.1 Elements of Visual Perception

- IP: foundation of mathematic and probabilistic formulations
- Human intuition and analysis play a central role
- Choices made based on subjective, visual judgments
- Physical limitations of human vision

- e.g: How human and electronic imaging devices compare in terms of resolution and ability to adapt to changes in illumination?
2.1 Elements of Visual Perception

2.1.1 Structure of the Human Eye

Contracts & expands to control the amount of light entering the eye

Central opening of the Iris: Pupil (diameter: ~ 2 to 8 mm)

Concentric layers of fibrous cells
Absorb ~8% of the visible light spectrum

Innermost membrane of the eye
Light from object imaged on retina
2.1.1 Structure of the Human Eye

Experimentation to illustrate the eye’s *Blind Spot*:
Close your left eye and stare at the cross. Get your head closer (or further) to the image. In a particular position, the dot should disappear. If you get even closer (or further) the dot appears again.
2.1.1 Structure of the Human Eye

Distribution of discrete light receptors over the surface of the retina

2 classes of receptors: cones and rods:

- **Cones**: 6-7 million in each eye, mainly located in the fovea. Highly sensitive to colour, fine details. “Photopic” or bright-light vision

- **Rods**: 75-150 million, distributed. Sensitive to low level of illumination, not involved in colour vision. “Scotopic” or dim-light vision

Distribution of receptors is radially symmetric about the fovea, except the so-called “blind spot”
2.1.1 Structure of the Human Eye

Approximation: fovea $\approx$ square sensor array of size $1.5 \text{ mm} \times 1.5 \text{ mm}$.
Density of cones in this area: 150,000 elements/mm$^2$
=> Number of cones in the region of highest acuity in the eye: $\sim 337,000$ elements.

Just in term of raw resolving power, a CCD can have this number of elements in a receptor array no larger than $5\text{mm} \times 5\text{mm}$.
=> basic ability of the eye to resolve detail is comparable to current electronic imaging sensors (but...)
2.1.2 Image Formation in the Eye

**Photo camera:** lens has *fixed focal length*. Focusing at various distances by *varying distance* between lens and imaging plane (location of film or chip)

**Human eye:** converse. *Distance* lens-imaging region (retina) is *fixed*. *Focal length* for proper focus obtained by *varying* the shape of the lens.
2.1.3 Brightness Adaptation and Discrimination

Eye's ability to discriminate between different intensity levels
Range of light intensity levels to which the human visual system can adapt: on the order of $10^{10}$

“Subjective Brightness”

Range of subjective brightness the eye can perceive when adapted to this level $B_a$
Perceived intensity is not a simple function of actual intensity

The visual system tends to undershoot or overshoot around the boundary of regions of different intensities

Mach bands

FIGURE 2.7 Illustration of the Mach band effect. Perceived intensity is not a simple function of actual intensity.
Simultaneous contrast phenomenon: a region’s perceived brightness does not depend simply on its intensity.
**Optical illusions:**

The eye fills in non-existing info or wrongly perceives geometrical properties of objects.
2.2 Light and the Electromagnetic Spectrum

Energy of one photon (electron volts)

|            | 10^6 | 10^5 | 10^4 | 10^3 | 10^2 | 10^1 | 10^-1 | 10^-2 | 10^-3 | 10^-4 | 10^-5 | 10^-6 | 10^-7 | 10^-8 | 10^-9 |
|------------|------|------|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|-------|
| Frequency  | 10^21| 10^20| 10^19| 10^18| 10^17| 10^16| 10^15| 10^14| 10^13| 10^12| 10^11| 10^10| 10^9  | 10^8  | 10^7  |
| Wavelength | 10^-12| 10^-11| 10^-10| 10^-9| 10^-8| 10^-7| 10^-6| 10^-5| 10^-4| 10^-3| 10^-2| 10^-1| 1     | 10^1  | 10^2  | 10^3  |

Visible spectrum

0.43 µm

0.79 µm

**FIGURE 2.10** The electromagnetic spectrum. The visible spectrum is shown zoomed to facilitate explanation, but note that the visible spectrum is a rather narrow portion of the EM spectrum.
Wavelength ($\lambda$) and frequency ($\nu$) related by:  

$$\lambda = \frac{c}{\nu}$$  

c $\approx 2.998 \times 10^8 \text{ m/s}$  

$\lambda$ in microns ($\mu\text{m}=10^{-6} \text{ m}$) or nanometers ($\text{nm}=10^{-9} \text{ m}$)

Energy (eV):  

$$E = h\nu$$  

(h: Planck’s constant)
• Light void of colour = *monochromatic* (or *achromatic*) light
  => only attribute: *intensity* or *gray level*

• Range of measured values = *gray scale*

• Monochromatic images = *gray-scale images*

Chromatic light source: frequency + radiance, luminance, brightness

• *Radiance* = total amount of energy that flows from the light source (W)

• *Luminance* (in lumens, lm) = measure of the amount of energy an observer *perceives* from a light source

• *Brightness* = subjective descriptor of light perception practically impossible to measure
2.3 Image Sensing and Acquisition

Transform of illumination energy into digital images:

The incoming energy is transformed into a voltage by the combination of input electrical power and sensor material.

Output voltage waveform = response of the sensor(s)

A digital quantity is obtained from each sensor by *digitizing* its response.
Ex: Photodiode
Made of silicon
Output voltage waveform proportional to light
Filter in front: increase selectivity
2.3.1 Image acquisition using a single sensor

Arrangement for high precision scanning

Lead screw

In-expensive (but slow) way to obtain high-resolution images
2.3.2 Image acquisition using sensor strips

Ex of use: airborne imaging applications

Ring configuration
Medical (CAT) and industrial imaging
(cross-sectional (slice) images of 3D objects

FIGURE 2.14 (a) Image acquisition using a linear sensor strip. (b) Image acquisition using a circular sensor strip.
2.3.3 Image acquisition using sensor arrays

- Illumination source reflected from a scene element
- Imaging system collects the incoming energy and focus it onto an image plane (sensor array)
- Response of each sensor proportional to the integral of the light energy projected
- Sensor output: analog signal → digitized

NB1: Motion not necessary
NB2: Predominant arrangement for digital cameras (e.g. CCD array)
2.3.3 Image acquisition using sensor arrays

**CCD cameras**: widely used in modern applications: private consumers, industry, astronomy…

CCD: Charge Couple Device

Rectangular grid of electron-collection sites laid over a thin silicon wafer

Image readout of the CCD one row at a time, each row transferred in parallel to a serial output register
2.3.3 Image acquisition using sensor arrays

Alternative to CCD cameras: **CMOS technology**

CMOS: Complementary Metal-Oxyde-Semiconductor

CMOS can potentially be implemented with fewer components, use less power and provide data faster than CCDs.

CCD: more mature technology

NB: a CMOS-based camera can be significantly smaller than a comparable CCD camera.
CCD vs CMOS

CCD: when exposure complete, transfers each pixel’s charge packet sequentially to a common output structure, which converts the charge to a voltage, buffers it and sends it off-chip.

CMOS imager: the charge-to-voltage conversion takes place in each pixel.

From:
CCD vs CMOS

- **Responsivity** *(amount of signal the sensor delivers per unit of input optical energy):* CMOS imagers marginally superior to CCDs

- **Dynamic range** *(ratio of a pixel’s saturation level to its signal threshold):* CCDs have advantage by factor of 2 in comparable circumstances

- **Uniformity** *(consistency of response for different pixels under identical illumination conditions):* CMOS imagers “traditionally worse”

- **Shuttering** *(ability to start and stop exposure arbitrarily):* standard feature of virtually all consumer and industrial CCDs

CCD vs CMOS

- **Speed**: CMOS arguably has the advantage over CCDs (all camera functions can be placed on the image sensor)

- **Windowing**: CMOS has ability to read out a portion of the image sensor (=> elevated frame or line rates for small ROI\(^{(1)}\)). CCDs generally more limited

- **Antiblooming** *(ability to gracefully drain localized overexposure without compromising the rest of the image in the sensor)*: CMOS generally has natural blooming immunity, CCDs require specific engineering

- **Reliability**: CMOS have advantage (all circuit functions can be placed on a single integrated circuit chip)

\(^{(1)}\) ROI = Region of Interest

2.3.4 A Simple Image Formation Model

Images denoted by two-dimensional functions \( f(x,y) \)

Value of amplitude of \( f \) at \((x,y)\): positive scalar quantity

Image generated by physical process: intensity values are proportional to the energy radiated by a physical source  \( \Rightarrow 0 < f(x,y) < \infty \)

\( f(x,y) \) may be characterized by 2 components:

1. The amount of source illumination \textit{incident} on the scene: \textit{illumination} \( i(x,y) \)
2. The amount of illumination \textit{reflected} by the objects of the scene: \textit{reflectance} \( r(x,y) \)

\[ f(x,y) = i(x,y) \, r(x,y) \], where \( 0 < i(x,y) < \infty \) and \( 0 < r(x,y) < 1 \)

\textit{total absorption} \quad \textit{total reflectance}
2.3.4 A Simple Image Formation Model

Example of typical ranges of illumination \( i(x,y) \) for visible light (average values):

- Sun on a clear day: \( \sim 90,000 \text{ lm/m}^2 \), down to 10,000 \( \text{lm/m}^2 \) on a cloudy day
- Full moon on a clear evening: \( \sim 0.1 \text{ lm/m}^2 \)
- Typical illumination level in a commercial office: \( \sim 1000 \text{ lm/m}^2 \)

Typical values of reflectance \( r(x,y) \):

- 0.01 for black velvet
- 0.65 for stainless steel
- 0.8 for flat white wall paint
- 0.9 for silver-plated metal
- 0.93 for snow
2.3.4 A Simple Image Formation Model

**Monochrome image**

Intensity \( l \): \( L_{\text{min}} \leq l \leq L_{\text{max}} \). In practice: \( L_{\text{min}} = i_{\text{min}} r_{\text{min}} \) and \( L_{\text{max}} = i_{\text{max}} r_{\text{max}} \)

Typical limits for indoor values in the absence of additional illumination:
\( L_{\text{min}} \approx 10 \) and \( L_{\text{max}} \approx 1000 \)

\([L_{\text{min}}, L_{\text{max}}]\) is called the gray (or intensity) scale

Common practice: shift to \([0, L-1]\), where \( l=0 \) is considered black and \( l=L-1 \) is considered white
2.4 Image Sampling and Quantization

2.4.1 Basic Concepts in Sampling and Quantization

Digitizing the coordinate values = Sampling

Digitizing the amplitude values = Quantization
2.4.1 Basic Concepts in Sampling and Quantization

Method of **sampling** determined by the sensor arrangement:

- **Single sensing element combined with motion**: spatial sampling based on number of individual mechanical increments
- **Sensing strip**: the number of sensors in the strip establishes the sampling limitations in one image direction; in the other: same value taken in practice
- **Sensing array**: the number of sensors in the array establishes the limits of sampling in both directions
The quality of a digital image is determined to a large degree by the number of samples and discrete intensity levels used in sampling and quantization.

However image content is also an important consideration in choosing these parameters.
2.4.2 Representing Digital Images

Continuous image: function of 2 continuous variables $f(s,t)$

$\rightarrow$ digital image by sampling and quantization

$\rightarrow$ 2D array $f(x,y)$, M rows and N columns, $(x,y)$ = discrete coordinates

$x = 0, 1, 2, \ldots, M-1$ and $y = 0, 1, 2, \ldots, N-1$

Section of the real plane spanned by the coordinates of an image = spatial domain

$x$ and $y$ are called spatial variables or spatial coordinates
2.4.2 Representing Digital Images

Representation useful for gray-scale images

NB: Origin and axes
→ TV + matrix

F of size 600x600 here
= 360,000 numbers…
Useful for algorithms
2.4.2 Representing Digital Images

Sampling $\Rightarrow \quad (x,y) \rightarrow f(x,y) = z$
$\mathbb{Z}^2 \rightarrow \mathbb{R}$

Quantization $\Rightarrow \quad (x,y) \rightarrow f(x,y) = z \in [0,L-1]$
$\mathbb{Z}^2 \rightarrow \mathbb{Z}$

The digitization process requires decisions on the values of $M$, $N$ and $L$ (number of discrete intensity levels)

No (theoretical) restrictions on $M$ and $N$ other than: $M > 0$ and $N > 0$

Due to storage and hardware, typically: $L = 2^k$

Assume that discrete levels are equally spaced and integers in $[0,L-1]$
Dynamic range = ratio of maximum measurable intensity to minimum detectable intensity level in the system
Rule: upper limit determined by saturation, lower limit determined by noise

Contrast = difference in intensity between the highest and the lowest intensity levels in an image

High dynamic range => high contrast expected
Low dynamic range => dull, washed-out gray look
Number b of bits required to store an image:
\[ B = M \times N \times k \]
\[ M = N \Rightarrow b = N^2 k \]

Image with \(2^k\) intensity levels => “\(k\)-bit image” (ex: 256 → 8-bit image)

<table>
<thead>
<tr>
<th>(N/k)</th>
<th>(1 (L = 2))</th>
<th>(2 (L = 4))</th>
<th>(3 (L = 8))</th>
<th>(4 (L = 16))</th>
<th>(5 (L = 32))</th>
<th>(6 (L = 64))</th>
<th>(7 (L = 128))</th>
<th>(8 (L = 256))</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>1,024</td>
<td>2,048</td>
<td>3,072</td>
<td>4,096</td>
<td>5,120</td>
<td>6,144</td>
<td>7,168</td>
<td>8,192</td>
</tr>
<tr>
<td>64</td>
<td>4,096</td>
<td>8,192</td>
<td>12,288</td>
<td>16,384</td>
<td>20,480</td>
<td>24,576</td>
<td>28,672</td>
<td>32,768</td>
</tr>
<tr>
<td>128</td>
<td>16,384</td>
<td>32,768</td>
<td>49,152</td>
<td>65,536</td>
<td>81,920</td>
<td>98,304</td>
<td>114,688</td>
<td>131,072</td>
</tr>
<tr>
<td>256</td>
<td>65,536</td>
<td>131,072</td>
<td>196,608</td>
<td>262,144</td>
<td>327,680</td>
<td>393,216</td>
<td>458,752</td>
<td>524,288</td>
</tr>
<tr>
<td>512</td>
<td>262,144</td>
<td>524,288</td>
<td>786,432</td>
<td>1,048,576</td>
<td>1,310,720</td>
<td>1,572,864</td>
<td>1,835,008</td>
<td>2,097,152</td>
</tr>
<tr>
<td>1024</td>
<td>1,048,576</td>
<td>2,097,152</td>
<td>3,145,728</td>
<td>4,194,304</td>
<td>5,242,880</td>
<td>6,291,456</td>
<td>7,340,032</td>
<td>8,388,608</td>
</tr>
<tr>
<td>2048</td>
<td>4,194,304</td>
<td>8,388,608</td>
<td>12,582,912</td>
<td>16,777,216</td>
<td>20,971,520</td>
<td>25,165,824</td>
<td>29,369,128</td>
<td>33,554,432</td>
</tr>
<tr>
<td>4096</td>
<td>16,777,216</td>
<td>33,554,432</td>
<td>50,331,648</td>
<td>67,108,864</td>
<td>83,886,080</td>
<td>100,663,296</td>
<td>117,440,512</td>
<td>134,217,728</td>
</tr>
<tr>
<td>8192</td>
<td>67,108,864</td>
<td>134,217,728</td>
<td>201,326,592</td>
<td>268,435,456</td>
<td>335,544,320</td>
<td>402,653,184</td>
<td>469,762,048</td>
<td>536,870,912</td>
</tr>
</tbody>
</table>

\(L = \text{Number of intensity levels}\)
2.4.3 Spatial and Intensity Resolution

Intuitively, \textit{spatial resolution} = measure of the smallest discernible detail in an image

Quantitatively (most common measures): line pairs per unit distance or dots (pixels) per unit distance (printing and publishing industry). In the US: dots per inch (dpi) e.g. newspapers: 75 dpi, magazines: 133 dpi, glossy brochures: 175 dpi, DIP book: 2400 dpi

Key point: to be meaningful, measures of spatial resolution must be stated \textit{w.r.t. spatial units}

\textit{Intensity resolution} = smallest discernible change in intensity level
Most common: 8bit. 16bit when needed. 32 bits rare. Exceptions: 10 or 12 bits
Effects of Sampling

Original image: 3692 x 2812 pixels
72 dpi image: 213 x 162 array
Smaller images zoomed back to the original size

**FIGURE 2.20** Typical effects of reducing spatial resolution. Images shown at: (a) 1250 dpi, (b) 300 dpi, (c) 150 dpi, and (d) 72 dpi. The thin black borders were added for clarity. They are not part of the data.
Aliasing Effect

Example in 1 dimension
Original image: 200x200 pixels

Sampled image: 100x100 pixels
Aliasing Effect

Original image: 622x756 pixels

205x250 pixels

“Moiré pattern”
Effects of Quantization

**FIGURE 2.21**
(a) 452 × 374, 256-level image.
(b)–(d) Image displayed in 128, 64, and 32 gray levels, while keeping the spatial resolution constant.

false contouring
Effects of Quantization

**FIGURE 2.21**
(Continued)
(c)–(h) Image displayed in 16, 8, 4, and 2 gray levels. (Original courtesy of Dr. David R. Pickens, Department of Radiology & Radiological Sciences, Vanderbilt University Medical Center.)
Experiment to study the effects on “image quality”:
Set of these 3 types of images generated by varying N (spatial resolution) and k (intensity resolution) independently
Observers asked to rank them according to their subjective quality
“Isopreference curve” in the Nk-plane
(Point lying on an isopreference curve correspond to images of equal subjective quality)

Isopreference curves tend to become more vertical as details in the image increase

Perceived quality remains the same while $N$ increases but $k$ decreases

$N = $ spatial resolution
$k = $ intensity resolution
2.4.4  Image Interpolation

Used in zooming, shrinking, rotating and geometric corrections
Interpolation applied to image resizing (shrinking and zooming): i.e. image resampling methods

Interpolation = process of using known data to estimate values at unknown locations

e.g. image 500x500 pixels enlarged 1.5 times => 750x750:
• Create an imaginary 750x750 grid with same sample spacing, shrink it so that it fits over the original image (=> pixel spacing reduced)
• Intensity level assignment for one pixel : look for its closest pixel in the original image and assign its intensity (nearest neighbour interpolation)
Problem of this approach: undesirable effects such as distortion of straight edges

Bilinear interpolation: use the 4 nearest neighbours to estimate the intensity at a given location \((x,y)\):
\[ v(x,y) = ax + by + cxy + d \] \((a,b,c,d\) determined from the 4 equations written using the 4 neighbours)

Bicubic interpolation: use the 16 nearest neighbours of a point:
\[ v(x, y) = \sum_{i=0}^{3} \sum_{j=0}^{3} a_{ij} x^i y^j \]
Chapter 2
Digital Image Fundamentals

FIGURE 2.24  (a) Image reduced to 72 dpi and zoomed back to its original size (3692 \times 2812 pixels) using nearest neighbor interpolation. This figure is the same as Fig. 2.20(d). (b) Image shrunk and zoomed using bilinear interpolation. (c) Same as (b) but using bicubic interpolation. (d)–(f) Same sequence, but shrinking down to 150 dpi instead of 72 dpi [Fig. 2.24(d) is the same as Fig. 2.20(c)]. Compare Figs. 2.24(e) and (f), especially the latter, with the original image in Fig. 2.20(a).
2.5 Some Basic Relationships between Pixels

Given an image $f(x,y)$ and pixels p or q

2.5.1 Neighbours of a pixel

- A pixel p at $(x,y)$ has 4 horizontal and vertical neighbours, whose coordinates are: $(x+1,y), (x-1,y), (x,y+1), (x,y-1) \rightarrow \text{set } N_4(p) \text{ (4-neighbours of p)}$
  
  NB: each is a unit distance from p, and some of these locations lie outside the image (borders)

- The 4 diagonal neighbours of p have coordinates: $(x+1,y+1), (x+1,y-1), (x-1,y+1), (x-1,y-1) \rightarrow \text{set } N_D(p)$

- $N_4(p) \cup N_D(p) = N_8(p) : \text{the set of 8-neighbours of p} \begin{bmatrix} 0 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
2.5.2 Adjacency, Connectivity, Regions and Boundaries

Let \( V \) be a set of intensity values used to define adjacency.

- **4-adjacency**: \( p \) and \( q \) with values in \( V \) are 4-adjacent if \( q \in N_4(p) \)
- **8-adjacency**: \( p \) and \( q \) with values in \( V \) are 8-adjacent if \( q \in N_8(p) \)
- **m-adjacency (mixed adjacency)**: \( p \) and \( q \) with values in \( V \) are m-adjacent if
  
  \( q \in N_4(p) \), or
  
  \( q \in N_D(p) \) and \( N_4(p) \cap N_4(q) \) has no pixel with values from \( V \)

<table>
<thead>
<tr>
<th></th>
<th>8-adjacency</th>
<th>m-adjacency</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0 1 1</td>
<td>0 1 1</td>
</tr>
<tr>
<td></td>
<td>0 1 0</td>
<td>0 1 0</td>
</tr>
<tr>
<td></td>
<td>0 0 1</td>
<td>0 0 1</td>
</tr>
<tr>
<td>1</td>
<td>0 0 0</td>
<td>0 0 0</td>
</tr>
<tr>
<td>1</td>
<td>0 1 0</td>
<td>0 1 0</td>
</tr>
<tr>
<td>0</td>
<td>1 0 0</td>
<td>1 0 0</td>
</tr>
<tr>
<td>0</td>
<td>0 1 1</td>
<td>0 1 1</td>
</tr>
<tr>
<td>1</td>
<td>1 1 0</td>
<td>1 1 0</td>
</tr>
<tr>
<td>a</td>
<td>0 0 0</td>
<td>0 0 0</td>
</tr>
<tr>
<td>d</td>
<td>0 1 0</td>
<td>0 1 0</td>
</tr>
<tr>
<td>c</td>
<td>0 1 1</td>
<td>0 1 1</td>
</tr>
<tr>
<td>b</td>
<td>0 0 1</td>
<td>0 0 1</td>
</tr>
</tbody>
</table>

**FIGURE 2.25** (a) An arrangement of pixels. (b) Pixels that are 8-adjacent (adjacency is shown by dashed lines; note the ambiguity). (c) m-adjacency. (d) Two regions that are adjacent if 8-adjacency is used. (e) The circled point is part of the boundary of the 1-valued pixels only if 8-adjacency between the region and background is used. (f) The inner boundary of the 1-valued region does not form a closed path, but its outer boundary does.
Path (or curve) from p \((x,y)\) to q \((s,t)\): sequence of distinct pixels with coordinates:
\((x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n)\), where \((x_0, y_0) = (x, y)\), \((x_n, y_n) = (s, t)\) and for i from 1 to n, \((x_i, y_i)\) and \((x_{i-1}, y_{i-1})\) are adjacent.

n = length of the path

\((x_0, y_0) = (x_n, y_n) \Rightarrow\) closed path

4-, 8-, or m-paths depending on the type of adjacency specified (cf figure)

Let S be a subset of pixels in an image

- P and q are connected in S if path exists between them consisting of pixels in S only
- For any p in S, set of pixels connected to it in S: connected component of S.
  - If only one: S is a connected set

- R is a region of the image if R is a connected set
- Ri and Rj adjacent if Ri U Rj = connected set
- Regions not adjacent are disjoint
2.5.3 Distance Measures

For pixels p, q and r, with coord \((x,y)\), \((s,t)\) and \((v,w)\) resp., D is a *distance function* or *metric* if:

\[
D(p, q) \geq 0 \quad (D(p, q) = 0 \iff p = q)
\]

\[
D(p, q) = D(q, p)
\]

\[
D(p, z) \leq D(p, q) + D(q, z)
\]

*Euclidian distance* between p and q:

\[
D_e(p, q) = \sqrt{(x - s)^2 + (y - t)^2}
\]

*D4 distance* (city-block distance, or *Manhattan distance*):

\[
D_4(p, q) = |x - s| + |y - t|
\]

*D8 distance* (*chessboard distance*, or Tchebychev distance):

\[
D_8(p, q) = \max(|x - s|, |y - t|)
\]

http://en.wikipedia.org/wiki/Taxicab_geometry
2.6 An Introduction to Mathematical Tools Used in Digital Image Processing

2.6.1 Array versus Matrix Operations

*Array operation:* on a *pixel-by-pixel* basis
NB: distinction between array and matrix operations

*Array product* of 2 images A and B:

\[
\begin{bmatrix}
a_{11} & a_{12} \\
a_{21} & a_{22}
\end{bmatrix}
\begin{bmatrix}
b_{11} & b_{12} \\
b_{21} & b_{22}
\end{bmatrix}
= \begin{bmatrix}
a_{11}b_{11} & a_{12}b_{12} \\
a_{21}b_{21} & a_{22}b_{22}
\end{bmatrix}
\]

(compare with the *matrix product*...)

2.6.2 Linear versus Nonlinear Operations

Let $H$ be a general operator producing an output image $g(x,y)$ for a given image $f(x,y)$:

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

$H$ is said to be a \textit{linear operator} if:

$$H[a_i f_i(x,y) + a_j f_j(x,y)] = a_i H[f_i(x,y)] + a_j H[f_j(x,y)]$$

$$= a_i g_i(x,y) + a_j g(x,y)$$
2.6.3 Arithmetic Operations

• \( s(x,y) = f(x,y) + g(x,y) \)

• \( d(x,y) = f(x,y) - g(x,y) \)

• \( p(x,y) = f(x,y) \times g(x,y) \)

• \( v(x,y) = f(x,y) \div g(x,y) \)

Operations performed between corresponding pixels in \( f \) and \( g \) for \( x = 0,1,2\ldots M-1 \) and \( y = 0,1,2\ldots M-1 \), \( M \) and \( N \) being the row and column sizes of the images

NB: \( s, d, p \) and \( v \) are images of the same size
2.6.5 Spatial Operations

3 broad categories:

a) Single-pixel operations
b) Neighbourhood operations
c) Geometric spatial transformations

a) Single-pixel operations

Transformation function $T$:

$$s = T(z)$$
b) Neighbourhood operations

$S_{x,y}$: set of coordinates of a neighbourhood centered on a point $(x,y)$ in an image $f$

Ex: average value of pixels in a rectangular neighbourhood of size $m \times n$ centered on $(x,y)$:

$$g(x, y) = \frac{1}{mn} \sum_{(r, c) \in S_{x,y}} f(r, c)$$
c) Geometric spatial transformations

*Rubber-sheet* transformations (cf. “printing” an image on a sheet of rubber and then stretching the sheet according to a predefined set of rules)

Geometric transformation: 2 basic operations:

1. *Spatial transformation* of coordinates \((x, y) = T\{(v, w)\}\)

2. *Intensity interpolation* that assigns intensity values to the spatially transformed pixels

1. Example of spatial transformation:

\[
(x, y) = T\{(v, w)\} = (v/2, w/2)
\]

Shrinks the original image to half its size in both spatial directions

*Affine transform* general form:

\[
\begin{bmatrix}
x \\
y \\
1
\end{bmatrix}
= \begin{bmatrix}
v \\
w \\
1
\end{bmatrix} T
= \begin{bmatrix}
v & w & 1
\end{bmatrix}
\begin{bmatrix}
t_{11} & t_{12} & 0 \\
t_{21} & t_{22} & 0 \\
t_{31} & t_{32} & 1
\end{bmatrix}
\]

Can scale, rotate, translate, sheer…

(2.6-23)
TABLE 2.2
Affine transformations based on Eq. (2.6–23).

<table>
<thead>
<tr>
<th>Transformation Name</th>
<th>Affine Matrix, T</th>
<th>Coordinate Equations</th>
<th>Example</th>
</tr>
</thead>
</table>
| Identity            | \[
\begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1 \\
\end{bmatrix}
\] | \[x = v\] \[y = w\] | T       |
| Scaling             | \[
\begin{bmatrix}
c_x & 0 & 0 \\
0 & c_y & 0 \\
0 & 0 & 1 \\
\end{bmatrix}
\] | \[x = c_x v\] \[y = c_y w\] | T       |
| Rotation            | \[
\begin{bmatrix}
cos \theta & \sin \theta & 0 \\
-sin \theta & cos \theta & 0 \\
0 & 0 & 1 \\
\end{bmatrix}
\] | \[x = v \cos \theta - w \sin \theta\] \[y = v \cos \theta + w \sin \theta\] | T       |
| Translation         | \[
\begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
t_x & t_y & 1 \\
\end{bmatrix}
\] | \[x = v + t_x\] \[y = w + t_y\] | T       |
| Shear (vertical)    | \[
\begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
s_x & 1 & 0 \\
0 & 0 & 1 \\
\end{bmatrix}
\] | \[x = v + s_x w\] \[y = w\] | T       |
| Shear (horizontal)  | \[
\begin{bmatrix}
1 & s_h & 0 \\
0 & 1 & 0 \\
0 & 0 & 1 \\
\end{bmatrix}
\] | \[x = v\] \[y = s_h w\] \[+ w\] | T       |

NB: Provide framework for *concatenating* together a sequence of operations.
2. Intensity interpolation

a) **Forward mapping**: scan the pixels of \( f \) and at each location \((v,w)\) compute the spatial location \((x,y)\) of the corresponding pixel in \( g \) using Eq. (2.6-23) directly.

b) **Inverse mapping**: scan the output pixel locations and at each \((x,y)\) compute the corresponding location in \( f \) using \((v,w) = T^{-1}(x,y)\) then interpolate among the nearest input pixels to determine the intensity of the output pixel value.
2.6.6 Vector and Matrix Operations

**Multispectral image processing**
e.g.: colour images formed in RGB colour space: Red, Green and Blue component images
Each pixel of an RGB image has 3 components:

\[
z = \begin{bmatrix}
z_1 \\
z_2 \\
z_3 \\
\end{bmatrix}
\]

- Intensity of the pixel in the **red** image
- … in the **green** image
- … in the **blue** image

RGB image of size MxN represented by 3 component images or a total of MN 3-D vectors

⇒ General multispectral case:
  - n component images, n-dimensional vectors

⇒ vector-matrix theory
Example:
Euclidean Distance $D$ between a pixel vector $z$ and a point $a$ in n-dimensional space:

$$D(z, a) = \left[ (z - a)^T (z - a) \right]^{1/2}$$

$$= \left[ (z_1 - a_1)^2 + (z_2 - a_2)^2 + ... + (z_n - a_n)^2 \right]^{1/2}$$

$$= \| z - a \| \quad (\text{Vector Norm})$$
2.6.7 Image Transforms

\[ T(u, v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) r(x, y, u, v) \]

Forward transform of \( f(x,y) \)

Input image

Forward transformation kernel

Recover \( f(x,y) \) :

\[ f(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} T(u, v) s(x, y, u, v) \]
Example of Image Processing in the *transform domain*
References:
• [http://www.sensorecleaning.com](http://www.sensorecleaning.com)
## Appendix: Images Formats (supported by Matlab Image Processing Toolbox)

<table>
<thead>
<tr>
<th>Format Name</th>
<th>Full Name</th>
<th>Description</th>
<th>Recognized Extensions</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIFF</td>
<td>Tagged Image File Format</td>
<td>A flexible file format supporting a variety of image compression standards, including JPEG. (container)</td>
<td>.tif, .tiff</td>
</tr>
<tr>
<td>JPEG</td>
<td>Joint Photographic Experts Group</td>
<td>A standard for compression of images of photographic quality</td>
<td>.jpg, .jpeg</td>
</tr>
<tr>
<td>GIF</td>
<td>Graphics Interchange Format</td>
<td>For 1- through 8-bit images. Frequently used to make small animations on the Internet</td>
<td>.gif</td>
</tr>
<tr>
<td>BMP</td>
<td>Windows Bitmap</td>
<td>Format used mainly for simple uncompressed images</td>
<td>.bmp</td>
</tr>
<tr>
<td>PNG</td>
<td>Portable Network Graphics</td>
<td>Compresses full colour images with transparency (up to 48 bits/pixel)</td>
<td>.png</td>
</tr>
<tr>
<td>XWD</td>
<td>X Window Dump</td>
<td></td>
<td>.xwd</td>
</tr>
</tbody>
</table>