Multimedia Systems
Images and Graphics

Er. Narayan Sapkota, M.Sc.
narayan.sapkota@eemc.edu.np

Everest Engineering College (Senior Lecturer)
Kathmandu University (Visiting Faculty)

January 7, 2025

Syllabus

1. Basic Image Concepts
   - Image Representation
   - Image Format
2. Image Processing Fundamentals
   - Image Synthesis/Generation
   - Image Analysis
   - Image Recognition*
   - Image Transmission
3. Image Enhancement
   - Enhancement by Point Processing
   - Spatial Filtering
   - Color Image Processing

Table of Contents

Basic Image Concepts (1)

Recommendation to Readers

For an in-depth exploration of the topics discussed in this chapter, readers are encouraged to consult Gonzalez's renowned work, Digital Image Processing.[1]

An image is a spatial representation of an object, a two- or three-dimensional scene, or another image. It can be real or virtual and is often considered a continuous function defining a rectangular region of a plane.

An image can be conceptualized as a function representing light intensity values at each point over a planar region. For digital computer processing, this function must be sampled at discrete intervals, where the intensity values are quantized into discrete levels.

Basic Image Concepts (2)

Recorded images may exist in formats like photographs, analog video signals, or digital files. In computer vision, images are typically recorded forms such as video or digital pictures, whereas in computer graphics, images are always digital.

An image can be described as a two-dimensional function, $f(x, y)$, where x and y represent spatial (plane) coordinates. The value of f at any point (x, y) is referred to as the intensity or gray level of the image at that specific location.

Basic Image Concepts (3)

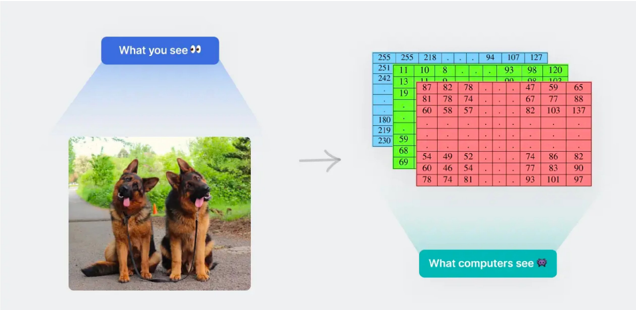

Figure 1: Creation of Digital Image

Table of Contents

Image Representation (1)

[1] A digital image is a numerical representation of a visual scene, designed for computer processing and storage. It is represented by a 2D matrix, where $I(r, c)$ denotes the intensity at position (r, c). These positions are called pixels, and in intensity images, the pixel values represent grayscale levels, encoding the image's brightness. Resolution refers to image detail, determined by the number of pixels, pixel density (PPI/DPI), and bit depth.

Sampling and Quantization

Let $f(s, t)$ be a continuous image function. To convert it to a digital image, we sample it into a 2D array $f(x, y)$ with M rows and N columns, where x and y are discrete coordinates. The coordinates (x, y) are integers, representing positions in the image. The image's section in the real plane is called the spatial domain, with x and y being spatial coordinates.

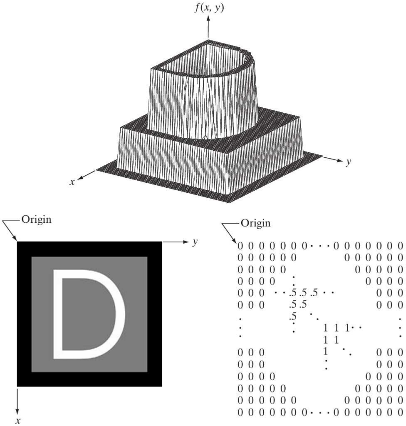

Figure 2: [Top] Image plotted as a surface. [Left] Displayed as a visual intensity array. [Right] Image shown as a 2-D numerical array (0, 0.5, and 1 represent black, gray, and white, respectively).

[1] See Section 2.4.2 of [1] for more details

How Digital Image Representation Works? (1)

Digital image representation starts by sampling a continuous image into discrete elements.

1. First, spatial sampling divides the image into a grid of small, uniform regions called pixels, which represent intensity or color at their respective locations. The resolution depends on the number of pixels in this grid.
2. Next, quantization approximates each pixel's intensity or color to a finite set of levels, determined by the bit depth. For example, a 1-bit image has 2 levels (black and white), an 8-bit image has 256 levels, and a 24-bit image can represent over 16 million colors, typical for true-color images.
3. The third step, color representation, uses different models. Grayscale images use a single intensity value, while RGB images use Red, Green, and Blue channels to define colors. Other models, such as CMYK or HSV, may be used based on the application.

How Digital Image Representation Works? (2)

4. Finally, coordinate representation places pixels on a 2D grid, indexed by row and column positions (x, y), usually starting from the top-left corner. Digital images are stored in formats like BMP, PNG, JPEG, and TIFF, optimized for compression, quality, or editing flexibility.

For example, to sample and quantize a standard (525-line) NTSC television picture with a VGA (Video Graphics Array) controller for re-display without noticeable degradation, a matrix of 640×480 pixels is required. Each pixel is represented by an 8-bit integer, allowing 256 discrete gray levels. This specification results in an array containing 307,200 8-bit numbers, or a total of 2,457,600 bits. In many applications, even finer sampling resolutions are necessary to capture more detail.

The *image file size* depends on the total number of pixels and the number of bits per pixel (quantization). It is given by the formula:

How Digital Image Representation Works? (3)

$$\text{Image File Size} = W \times L \times n \, \text{bits}$$

where:

$$W = \text{width of the image in pixels,}$$

$$L = \text{height of the image in pixels,}$$

$$n = \text{number of bits per pixel (bit depth).}$$

For example, a 640×480 image with 8 bits per pixel requires:

$$640 \times 480 \times 8 = 2,457,600 \, \text{bits} \approx 307.2 \, \text{KB}.$$

Table of Contents

Image Format (1)

There are different kinds of image formats in the literature. We shall consider the image format that comes out of an image frame grabber, i.e., the captured image format, and the format when images are stored, i.e., the stored image format.

1. **Captured Image Format:** The Captured Image Format refers to the specifications of an image at the moment it is acquired by an image capture device, such as a camera or frame grabber. It is determined by two key parameters: spatial resolution and color encoding.

   The spatial resolution is specified as width x height. For example 320 x 240 pixels means 76,800 pixels in the entire image. The resolution determines how much information is captured in the image. A higher resolution provides more pixels to represent fine details, whereas a lower resolution may capture a coarser version of the image.

Image Format (2)

The second key parameter, color encoding, refers to how each pixel's color is represented in terms of bits per pixel (bpp). This determines the number of possible colors or grayscale levels each pixel can represent. For instance, a 1-bit color encoding represents only two colors (black and white), a 9-bit encoding can represent 512 shades or colors, while a 24-bit encoding allows for 16.7 million different colors, typically using an RGB (Red, Green, Blue) model.

These two factors, spatial resolution and color encoding, are determined by the hardware and software capabilities of the image capture device. For example, a SPARCstation using a VideoPig.card for image capturing might produce images with a resolution of 320 x 240 pixels, and the color encoding could vary from 1-bit (binary image), 9-bit (limited color or grayscale), to 24-bit (true color). The captured image format, therefore, defines the image's quality, size, and color depth at the moment of capture, before any further processing or storage takes place.

2. **Stored Image Format:** To store an image, the image is represented in a two-dimensional matrix, where each value corresponds to the data associated with one image pixel. In bitmaps, these values are binary numbers. In color images, the values can be one of the following:
   - Three numbers that normally specify the intensity of the red, green, and blue components.
   - Three numbers representing references to a table that contains the red, green, and blue intensities.
   - A single number that works as a reference to a table containing color triples.
   - An index pointing to another set of data structures, which represents colors.

Image Format (4)

If sufficient memory is available, an image can be stored using uncompressed RGB triples. However, when storage is limited, compression methods are necessary. Storing an image involves saving the color values for each pixel, along with additional information such as width, height, depth, and potentially the creator's name.

To manage image properties, flexible formats like RIFF (Resource Interchange File Format) and BRIM are used. RIFF supports various media types including bitmaps, vector drawings, and audio, while BRIM stores image details like dimensions, authoring information, and history. Popular image formats include PostScript, GIF, XBM, JPEG, TIFF, PBM, and BMP, with JPEG being widely used.

Graphics Formats (1)

Graphics formats are defined using graphics primitives and their associated attributes:

- Graphics Primitives: Include 2D shapes (e.g., lines, rectangles) and 3D objects (e.g., polyhedra).
- Attributes: Such as line style, width, and color effects, determine the final appearance of the graphic.
- High-level Representation: The image is not stored as a pixel matrix, reducing storage requirements and simplifying manipulation.

However, this high-level representation must eventually be converted into a lower-level form for display:

- Bitmap: Represents the image as an array of pixel values, with each pixel using a single bit.
- Pixmap: Uses multiple bits per pixel (e.g., 8 bits for 256 colors).

Graphics packages, like SRGP, can facilitate the conversion from primitives to bitmap or pixmap.

Table of Contents

Image Processing Fundamentals (1)

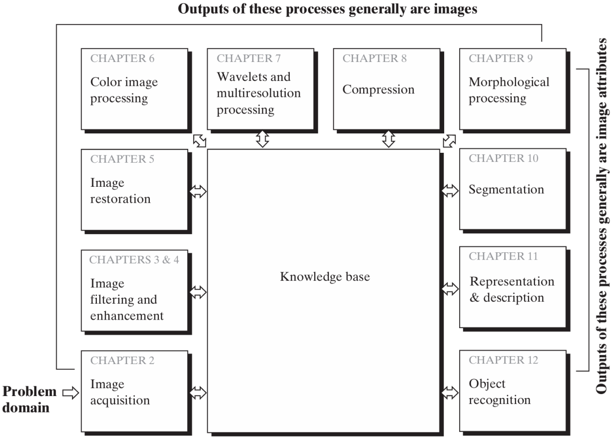

Figure 3: Fundamental steps in digital image processing Ref. [1]

Image Processing Fundamentals (2)

Computer graphics and image processing are closely related fields but involve opposite processes:

- Computer graphics: Focuses on the creation or synthesis of images from computer-based models, generating 2D or 3D graphical representations.
- Image processing: Deals with analyzing images or reconstructing models from 2D and 3D object pictures. It includes tasks like image recognition and analysis, where the input is an image (e.g., a photograph or video frame) and the output could be another image or specific parameters.
- Digital image processing: Most common, but analog image processing is also possible.

In essence, computer graphics is about creating images from models, while image processing is about extracting information or reconstructing models from images.

Table of Contents

Image Synthesis/Generation (1)

Image synthesis plays a crucial role in computer-supported user interfaces and is essential for visualizing 2D, 3D, and higher-dimensional objects. It is widely used in various fields, such as education, science, medicine, construction, advertising, and entertainment, including:

1. User interfaces: Applications like Microsoft Windows manage activities and offer point-and-click options for menu items, icons, and objects.
2. Office automation and electronic publishing: Graphics are integral to creating and distributing documents, including text, tables, and graphs, with hypermedia systems enabling multimedia document viewing.
3. Simulation and animation for scientific visualization and entertainment: Animated movies and scientific simulations, such as studying flow behavior and chemical reactions, rely on 3D modeling. Trends also include visual effects in commercials and movies.

Image Synthesis/Generation (2)

Image synthesis/generation refers to the creation of new images from computer-based models or data. It is an essential process in computer graphics, enabling the creation of realistic 2D, 3D, and even higher-dimensional images for various applications. The synthesis can involve the generation of entire scenes, objects, or textures. The process of image synthesis involves several techniques, including:

- Rasterization: Converts vector graphics (geometric models) into a pixel-based image. This method is commonly used for creating 2D images from 3D models in computer graphics.
- Ray tracing: Simulates the physical behavior of light to create realistic lighting, shadows, reflections, and refractions. It is commonly used in high-quality renderings and animations.

Image Synthesis/Generation (3)

- Procedural generation: Uses algorithms to generate textures, landscapes, and even entire scenes. This approach is widely used in video games and simulations, where large, complex environments are needed without manually modeling each element.
- Machine learning and neural networks: These methods generate images by learning patterns from large datasets. Generative Adversarial Networks (GANs) are particularly popular for creating realistic images that can resemble photographs, art, or synthetic data.
- Photogrammetry: Involves capturing 3D models and textures from multiple 2D images of a scene. This technique is often used in virtual reality (VR), augmented reality (AR), and digital preservation.

Image Synthesis/Generation (4)

How image synthesis works ?

Image synthesis is typically carried out using computer software and hardware tools. The process involves:

- Model creation: Building a 3D model or scene using software like Blender, Maya, or AutoCAD.
- Rendering: The process of converting the 3D model into a 2D image by simulating light and shadow effects, textures, and camera perspectives.
- Post-processing: Enhancing the final output using techniques like compositing, color correction, and adding special effects.

Dynamics in Graphics* (1)

Graphics are not limited to static images but can be dynamically varied, allowing user control over aspects such as animation speed, scene portion, and detail level. This dynamic element is essential in interactive graphics, facilitated by both hardware and software for motion and updates:

1. Motion Dynamics: Objects can move relative to a stationary observer, or both objects and the view can move. For example, a flight simulator uses a moving platform to simulate motion and changes in the environment.
2. Update Dynamics: Changes to the shape, color, or other properties of objects are made. For instance, simulating an airplane's structure deformation in-flight based on user input. Smooth transitions enhance realism and communication of information, such as object properties and time variations.

Table of Contents

Image Analysis (1)

Image analysis focuses on extracting essential information from images to enable higher-level scene understanding. Simply knowing pixel positions and values is insufficient for tasks like object recognition or scene interpretation. Key tasks include:

- Computing perceived brightness and color
- Recovering 3D data from 2D images
- Detecting object boundaries (discontinuities)
- Characterizing uniform regions

Image analysis transforms raw pixel data into meaningful descriptions that aid in complex scene interpretation. It includes subareas like:

- Image enhancement (improving quality by reducing noise or enhancing contrast)
- Pattern detection and recognition (e.g., OCR for text input)

Image Analysis (2)

A key example is Optical Character Recognition (OCR), which allows efficient input of typed or handwritten text. The accuracy of handwriting recognition depends on the input device, such as a tablet stylus for online recognition, which records stroke order, direction, speed, and pressure to match these factors to stored templates.

Table of Contents

Image Recognition* (1)

Image recognition involves identifying objects in an image based on properties like color, texture, and edges. The process requires matching the observed spatial configuration of pixels with an expected projection.

Key conditions for recognition include:

- Deriving the position and orientation of an object from its spatial configuration.
- Verifying the correctness of this derivation.

Steps in recognizing an object's position, orientation, and category:

- Segmentation: Separating object pixels from the background.
- Feature Extraction: Identifying key characteristics (e.g., lines, curves) of the object.
- Derivation of Object Properties: Matching image features to object properties (e.g., shapes, orientation).

Image Recognition* (2)

- Recognition Difficulty: Task complexity varies with object type, background, and sensor properties.
- Edge Extraction: Simple methods like edge detection can identify object corners.
- Transformation Complexity: Recognizing complex objects can be challenging due to overlapping, shadows, and dynamic backgrounds.
- Task and Image Complexity: Transformation methods depend on the task and the image's complexity.

Image Recognition* (3)

0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	FF	FF	FF	FF	FF	0	0	0	0
0	0	0	0	FF	FF	FF	FF	FF	0	0	0	0
0	0	0	0	FF	FF	FF	FF	FF	0	0	0	0
0	0	0	0	FF	FF	FF	FF	FF	0	0	0	0
0	0	0	0	FF	FF	FF	FF	FF	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 4: Numeric digital intensity image with a white square (gray level FF) on black background (gray level 0) of a symbolic image

Image Recognition* (4)

N	N	N	N	N	N	N	N	N	N	N	N	N
N	N	N	N	N	N	N	N	N	N	N	N	N
N	N	N	N	C	N	N	N	C	N	N	N	N
N	N	N	N	N	N	N	N	N	N	N	N	N
N	N	N	N	N	N	N	N	N	N	N	N	N
N	N	N	N	N	N	N	N	N	N	N	N	N
N	N	N	N	C	N	N	N	C	N	N	N	N
N	N	N	N	N	N	N	N	N	N	N	N	N
N	N	N	N	N	N	N	N	N	N	N	N	N

Figure 5: Numeric digital intensity image of the corners of the image from Figure 4 (C = corner; N = no corner)

Steps in Image Recognition (1)

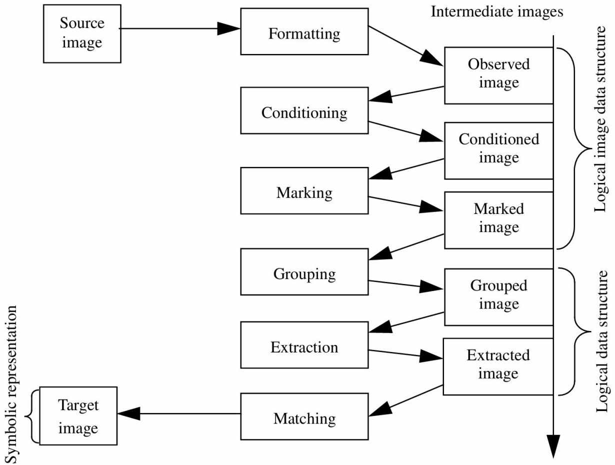

Figure 6: Steps involved in image recognition

Steps in Image Recognition (2)

1. **Formatting:** The formatting step shoots an image by use of a camera and transforms the image into digital form.
2. **Conditioning:** This step reduces noise and irrelevant variations in the image by estimating the underlying information pattern. It normalizes the background and suppresses disturbances, ensuring that only the relevant data is considered.
3. **Marking:** Here, the image is analyzed to identify spatial objects, which are groups of interconnected pixels. Edge recognition techniques detect boundaries or discontinuities (e.g., the rim of a cup), and marking operations filter and highlight the significant edges. Additional marking operations may identify specific points, like corners.

Steps in Image Recognition (3)

4. **Grouping:** In this step, the marked pixels are grouped into recognizable objects by identifying clusters of connected pixels. Techniques like line fitting (e.g., Hough transform) are used to connect edges and form objects. The grouping operation changes the image's logical structure, organizing the data into meaningful units.
5. **Extraction:** This operation calculates various properties for each pixel group, such as center, surface, orientation, or curvature. These properties give semantic meaning to the groups, allowing for further analysis. It can also identify spatial relationships between groups, such as proximity or overlap.

Steps in Image Recognition (4)

6. **Matching:** After extraction, the objects are identified but lack content meaning. The matching step assigns meaning to these objects by comparing them with known models or templates, recognizing them as specific items (e.g., a chair or the letter "A"). Measurements such as distances and angles can be made to verify the object's properties, especially in inspection scenarios.

Table of Contents

Image Transmission (1)

Image transmission involves transmitting digital images through computer networks, and there are several key requirements for the network:

- The network must accommodate burst data transport due to the large size of images.
- Reliable transport is necessary for successful transmission.
- Unlike audio/video transmission, time-dependence is not a significant factor for image transmission.

The image size depends on the representation format used for transmission. There are several possibilities for how image data can be transmitted:

1. Raw image data transmission: The image is generated through a video digitizer and transmitted in its digital form. The size is calculated as: size $=$ spatial resolution $\times$ pixel quantization. For example, transmitting an image with a resolution of 640x480 pixels and 8 bits per pixel requires 301,200 bytes.

Image Transmission (2)

2. Compressed image data transmission: The image is digitized and then compressed before transmission using methods like JPEG or MPEG. The reduction in image size depends on the compression method and rate.
3. Symbolic image data transmission: The image is represented using symbolic data like image primitives (e.g., 2D or 3D geometric representation) and attributes. The image size is equal to the structure size, which contains the transmitted symbolic information of the image. This method is commonly used in computer graphics.

Table of Contents

Image Enhancement

[2]Image enhancement is the process of modifying an image to make it better suited for a specific application. The word "specific" is important because it indicates that enhancement methods are designed for particular problems. For example, a technique that works well for improving X-ray images may not be suitable for enhancing satellite images in the infrared spectrum.

[2]See Section 3.2 and 3.3 of Gonzalez Book

Table of Contents

Enhancement by Point Processing (1)

Image enhancement using point processing refers to the manipulation of pixel values individually, based on their intensity. Point processing does not depend on neighboring pixels, meaning that the enhancement is applied to each pixel independently.

1. **Brightness Adjustment**
    - Purpose: To increase or decrease the overall brightness of an image.
    - Method: Add or subtract a constant value to all pixel intensities.
    - If $I(x, y)$ is the intensity of the pixel at position (x, y) and c is a constant, the new intensity becomes:

$$I'(x, y) = I(x, y) + c$$

    - This shifts the image brightness, but it may cause clipping of pixel values (if they exceed the maximum value).
2. **Contrast Adjustment**

Enhancement by Point Processing (2)

- Purpose: To improve the contrast by stretching or compressing the range of pixel values.
- Method: Stretch the pixel intensity range to cover the entire possible range.
- This is typically achieved using a linear transformation:

$$I'(x, y) = \frac{I(x, y) - \min(I)}{\max(I) - \min(I)} \times (L - 1)$$

where $\min(I)$ and $\max(I)$ are the minimum and maximum pixel values in the image, and L is the number of intensity levels (e.g., 256 for an 8-bit image).

3. **Histogram Equalization**
   - Purpose: To redistribute the image's intensity values across the entire range, improving global contrast.
   - Method: This technique modifies the image to make the histogram of pixel intensities as flat as possible.

Enhancement by Point Processing (3)

- The cumulative distribution function (CDF) is computed, and the pixel intensities are mapped to new values based on the CDF.

4. **Gamma Correction**
   - Purpose: To adjust the brightness of an image, commonly used in display devices.
   - Method: Gamma correction is a non-linear transformation:

   $$I'(x, y) = c \cdot I(x, y)^{\gamma}$$

   where c is a constant and γ is the gamma value (typically between 0 and 1 for darkening, and greater than 1 for brightening).

5. **Thresholding**
   - Purpose: To convert an image into a binary form, often used for segmentation or edge detection.

- Method: A threshold T is selected, and all pixel values are compared to it:

$$I'(x,y) = \begin{cases} 255 & \text{if } I(x,y) \geq T \\ 0 & \text{if } I(x,y) < T \end{cases}$$

- This results in a binary image with black and white regions.

6. **Identity Transformation**
   - Purpose: This transformation leaves the image unchanged. It is often used as a reference for comparing other transformations.
   - Method: The output pixel value is identical to the input pixel value: In this case, the output pixel value is the same as the input pixel value. It does not alter the image:

$$I(x,y) = I'(x,y)$$

This transformation is often used as a reference for comparing other transformations.

7. **Negative Transformation**
   - Purpose: To invert the colors of the image, often used in creative image processing or image analysis.
   - Method: This simply inverts the pixel intensities:

$$I'(x, y) = L - 1 - I(x, y)$$

where L is the number of intensity levels in the image (e.g., 256 for an 8-bit image).

Enhancement by Point Processing (6)

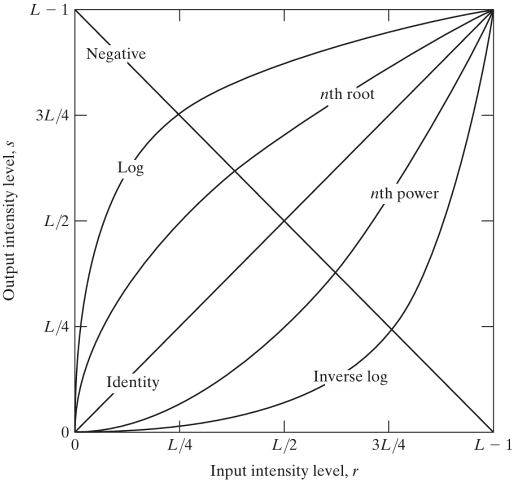

Figure 7: Some basic intensity transformation functions

Enhancement by Point Processing (7)

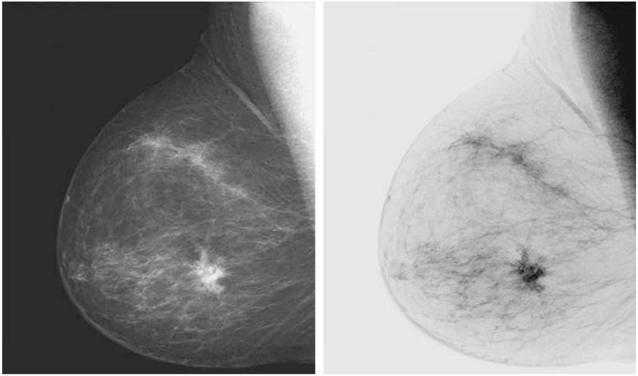

Figure 8: (LEFT) Original digital mammogram. (RIGHT) Negative image obtain
the negative transformation

- **Log Transformation**
  - Purpose: Enhances darker areas of the image by applying a logarithmic function to the pixel values:

    $$I'(x, y) = c \cdot \log(1 + I(x, y))$$

    where c is a constant, $I(x, y)$ is the input pixel value, and $I'(x, y)$ is the output pixel value.
  - This transformation is particularly useful for images with dark details that need enhancement.

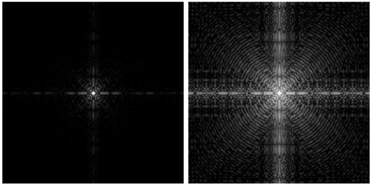

Figure 9: Result of applying the log transformation with c =

Enhancement by Point Processing (9)

- **Inverse-Log Transformation:**
  - Purpose: Enhances bright regions while compressing dark regions.
  - Method: Inverse of the log transformation:

$$I'(x, y) = c \cdot \exp(I(x, y))$$

where $\exp(I'(x, y))$ denotes the exponential of $I(x, y)$, and c is a scaling constant.

Enhancement by Point Processing (10)

Applications of Point Processing

- Image enhancement: Improving contrast, brightness, and clarity.
- Medical Imaging: Enhancing certain structures in images for better diagnosis.
- Satellite Imagery: Adjusting contrast and brightness to highlight specific features.
- Artistic Effects: Using transformations like gamma correction or negative transformation for creative effects.

Table of Contents

Spatial Filtering (1)

Spatial filtering is one of the most widely used methods for image enhancement. It operates directly on the image pixels based on their neighborhood (spatial domain), adjusting pixel values to achieve desired effects.

What is Spatial Filtering?

Spatial filtering involves modifying the intensity of each pixel in an image by applying a filter (kernel or mask) to the pixels in its neighborhood. The result is a new image where each pixel's value is replaced by a weighted average (or some other operation) of its neighboring pixels. The filter works by convolving the image with the filter kernel.

- Define a kernel (filter) matrix, which will modify the pixel's intensity.
- Convolve the image with the kernel to produce the enhanced image.

Spatial Filtering (2)

Spatial Filter Types

1. Linear Filters:
    - Averaging Filter: Blurs the image, reducing noise.
    - Gaussian Filter: Similar to averaging but uses a Gaussian function for weights, creating a smoother blur.
2. Non-Linear Filters:
    - Median Filter: Replaces the pixel value with the median of its neighborhood. Useful for removing salt-and-pepper noise.
    - Max/Min Filters: Retains the maximum or minimum pixel value in the neighborhood.

Spatial Filtering (3)

Convolution Operation: The core of spatial filtering is the convolution operation, which is mathematically expressed as:

$$g(x, y) = \sum_{i=-k}^{k} \sum_{j=-k}^{k} f(x + i, y + j) \cdot h(i, j)$$

where:

- $f(x, y)$ is the original image.
- $h(i, j)$ is the filter kernel.
- $g(x, y)$ is the filtered image.

Spatial Filtering (4)

The filter is applied to each pixel in the image and its neighbors, and the result is a new image $g(x, y)$.

Averaging Filter

This is a simple linear filter where each pixel's value is replaced by the average of its neighborhood. This is used for blurring the image to reduce noise.

The kernel for a 3x3 averaging filter is:

$$h = \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

The effect of the filter is that it smooths out sharp edges and reduces fine details.

Spatial Filtering (5)

Gaussian Filter

This is another linear filter that uses a Gaussian function to assign weights to pixels. The Gaussian filter produces a blur effect while preserving edges better than the averaging filter.

The 2D Gaussian kernel is given by:

$$h(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

where σ controls the amount of blur. Larger values of σ result in a stronger blur.

Median Filter

This is a non-linear filter that replaces the value of a pixel with the median value of its neighbors. It is particularly effective in removing salt-and-pepper noise while preserving edges.

Spatial Filtering (6)

Example of a 3x3 neighborhood:

$$\begin{bmatrix} 1 & 9 & 3 \\ 2 & 5 & 4 \\ 6 & 7 & 8 \end{bmatrix}$$

The median value of this matrix is 5, so the pixel is replaced with the value 5.

Spatial Filtering (7)

Applications of Spatial Filtering Spatial filtering is widely used in various fields for different image enhancement tasks:

- Noise Reduction: Filters like the median and Gaussian filters are effective for removing noise, especially in medical images or satellite imagery.
- Edge Detection: Filters can highlight edges by emphasizing areas of rapid intensity change, useful in object detection and image segmentation.
- Image Smoothing and Sharpening: Filters can enhance details and smooth out unwanted noise, used in photography, video processing, and computer vision.

Table of Contents

Color Image Processing (1)

- Color Models

  RGB (Red, Green, Blue): The most common model used in digital devices such as monitors and cameras. Each color is created by combining red, green, and blue intensities.

  HSV (Hue, Saturation, Value): More intuitive for users, allowing color adjustment by specifying the hue (color type), saturation (vibrancy), and value (brightness).

  YCbCr: Commonly used in video compression and broadcasting, separating brightness (Y) from chrominance (Cb, Cr) components.

  Lab Color Space: A perceptually uniform model based on human vision, independent of display or capture device. It contains Lightness (L) and two color-opponent channels (a and b) where a is representation of red-green channel and b is yellow-blue channel.

- Color Space Transformation
    - Conversion between color models (e.g., RGB to HSV, RGB to YCbCr) is essential for specific tasks like image enhancement, compression, or segmentation.
    - Transformation allows for more efficient color processing, such as isolating specific components for analysis.
- Color Image Enhancement
    - Contrast Adjustment: Adjusting color contrast to enhance details. Common in both RGB and HSV models.
    - Histogram Equalization: Improves contrast by redistributing intensities across the range. Applied to individual color channels for better image details.
    - Color Filtering: Emphasizing specific color ranges to highlight regions of interest (e.g., isolating red hues in medical images).

Color Image Processing (3)

- Color Image Segmentation:
  - Dividing an image into regions with similar color values to simplify analysis (e.g., separating objects from background).
  - Important in applications like object detection and facial recognition.

Color Image Processing (4)

- Applications:
    - Medical Imaging: Color processing helps detect anomalies, such as in skin cancer detection where color and texture variations are critical.
    - Computer Vision: Used in object recognition, face detection, and autonomous vehicles for identifying objects and navigating environments.
    - Digital Photography: Enhances color balance, saturation adjustments, and artistic filters for improving image quality.
    - Entertainment: Color grading in film and video editing to create mood and aesthetic through color manipulation.

Color Image Processing (5)

- Challenges:
  - Lighting Variations: Different lighting conditions affect how colors appear, making color consistency challenging.
  - Color Constancy: The ability to maintain consistent color perception despite changes in light.
  - Noise and Artifacts: Color channels may contain noise or artifacts, impacting the accuracy of color-based processing.
  - Device Dependent: Color reproduction can vary significantly across different devices (e.g., monitors, printers), complicating color management.

Rafael C Gonzalez.
Digital image processing.
Pearson education india, 2009.