Image Processing Fundamentals

Dr. Romik Chatterjee
Vice President of Engineering
Graftek Imaging, Inc.
Agenda

• Objectives and Motivations
• Enhancing Images
• Checking for Presence
• Locating Parts
• Measuring Features
• Identifying and Verifying Components
Class Goals

• Teach the fundamental image processing tools available in machine vision software
• Provide some background into how the algorithms work
• Accelerate your machine vision learning curve

• What not to expect
  – Learn how to develop a complete vision system
  – Learn specific, proprietary functions
  – 3D vision, color or other advanced topics
  – Discussion of the benefits of different application development environments
Image Processing for Machine Vision

• Objective
  – To extract useful information present in an image, in a limited time
• Secondary
  – To display an image for users
• Not
  – Improve appearance of image in general
• Used for
  – Image pre-processing
    • Minimize variations of information in the image
    • Prepare the image for processing and measurement
  – Application specific processing
    • Use image to count, locate, and measure attributes
Image Types

- **Grayscale**
  - 8 bit: pixel values range from 0 to 255
  - 12 bit: pixel values range from 0 to 4095
  - 16 bit: pixel values range from 0 to 65535

- **Color**
  - Composed of 3 grayscale images (RGB)

- **Other types**
  - Binary: pixel values: 0 and 1
    - Commonly used to identify objects of interest in an image
    - Usually the result of image processing step
  - Floating point: pixel values range from –X.xx to X.xx
    - Usually a result of a computation
What is an Ideal Image?

• Range of grayscale values
  – Spread out between 0 and 255
  – No pixels “saturated” at 255 (for most applications)
    • Impossible to distinguish between saturated pixels

• Good contrast
  – Between the “right” parts of the image

• Repeatable

In short, an ideal image requires the least number of image processing steps to obtain the result.
## Class Organization

<table>
<thead>
<tr>
<th>Enhance</th>
<th>Check</th>
<th>Locate</th>
<th>Measure</th>
<th>Identify</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Filter noise or unwanted features</td>
<td>• Create Particles</td>
<td>• Match patterns</td>
<td>• Detect edges</td>
<td>• Read text (OCR)</td>
</tr>
<tr>
<td>• Remove distortion</td>
<td>• Measure intensity</td>
<td>• Match geometry</td>
<td>• Measure distance</td>
<td>• Read 1D barcodes</td>
</tr>
<tr>
<td>• Calibrate images</td>
<td>• Analyze particles</td>
<td>• Set-up coordinate systems</td>
<td>• Calculate geometry</td>
<td>• Read 2D codes</td>
</tr>
</tbody>
</table>
Class Organization

Enhance
- Filter noise or unwanted features
- Remove distortion
- Calibrate images

Check
- Create Particles
- Measure intensity
- Analyze particles

Locate
- Match patterns
- Match geometry
- Set-up coordinate systems

Measure
- Detect edges
- Measure distance
- Calculate geometry

Identify
- Read text (OCR)
- Read 1D barcodes
- Read 2D codes
Motivation

Read characters on a textured surface
Motivation

Not possible to cleanly segment characters
Motivation

Results without preprocessing
Motivation

Read characters on a textured surface
Motivation

Image after periodic pattern is removed
Motivation

Results without preprocessing

Results with preprocessing
Objective of Image Preprocessing

- Process an image so that the resulting image is more suitable than the original for a specific application

- A preprocessing method that works well for one application may not be the best method for another application
Image Preprocessing

• Pre-processing occurs before the application specific processing

Acquisition → Preprocessing → Application Specific Processing

Preprocessing:
- Shading Correction
- De-blurring
- De-noising
- Contrast Enhancement
- Feature Enhancement

Application Specific Processing:
- Intensity Measurements
- OCR
- Pattern Matching
- Gauging
- Barcode
- Particle Analysis
Enhancement Techniques

- Spatial Domain: pixel-wise operations
  - Brightness, contrast and gamma
  - Lookup tables
  - Gray morphology
  - Filtering (smoothing, median, general convolution)

- Frequency Domain
  - Deblurring
  - Filtering
Lookup Tables

- A lookup table is a function that maps grayscale values in an image to new grayscale values, to create a new result image.

<table>
<thead>
<tr>
<th>Source</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>255</td>
</tr>
<tr>
<td>1</td>
<td>254</td>
</tr>
<tr>
<td>2</td>
<td>253</td>
</tr>
<tr>
<td>..</td>
<td>..</td>
</tr>
<tr>
<td>255</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Source</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>200</td>
</tr>
<tr>
<td>1</td>
<td>200</td>
</tr>
<tr>
<td>2</td>
<td>200</td>
</tr>
<tr>
<td>..</td>
<td>..</td>
</tr>
<tr>
<td>255</td>
<td>200</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Source</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>240</td>
</tr>
<tr>
<td>1</td>
<td>245</td>
</tr>
<tr>
<td>2</td>
<td>246</td>
</tr>
<tr>
<td>..</td>
<td>..</td>
</tr>
<tr>
<td>255</td>
<td>171</td>
</tr>
</tbody>
</table>
Brightness

- Adds a constant grayscale to all of the image
- Can improve appearance but not useful for most image processing steps
Contrast

- Used to increase or decrease contrast
- Normal = 45 (degree line)
  - High = Higher than 45
  - Low = Lower than 45
- Typical use is to improve edge detection
- Sacrifices one range of values to improve another
• Nonlinear contrast enhancement
• Higher gamma improves contrast for larger grayscale values
• Does not cause saturation
Histogram Equalization

- Alters grayscale values of pixels so that they become evenly distributed across the full grayscale range
- The function associates an equal number of pixels per constant grayscale interval
- Takes full advantage of the available shades of gray
- Enhances contrast of the image without modifying the structure
Histogram Equalization

Bright

Dark

Low Contrast

High Contrast

Improved contrast between characters and background
Other Lookup Tables

Original

Reverse

Square

Power 1/x
## Gray Morphology

<table>
<thead>
<tr>
<th>Morphology</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Erosion</td>
<td>Min(Neighbors)</td>
</tr>
<tr>
<td>Dilation</td>
<td>Max(Neighbors)</td>
</tr>
<tr>
<td>Open</td>
<td>Dilation(Erosion(I))</td>
</tr>
<tr>
<td>Close</td>
<td>Erosion(Dilation(I))</td>
</tr>
</tbody>
</table>

**Diagram:**
- Erosion (Min(Neighbors))
- Dilation (Max(Neighbors))
- Open (Dilation(Erosion(I)))
- Close (Erosion(Dilation(I)))
## Spatial Filtering

<table>
<thead>
<tr>
<th>Filter Type</th>
<th>Filters</th>
</tr>
</thead>
</table>
| Lowpass     | • Smoothing  
             • Gaussian  
             • Median  
             • Nth Order |
| Highpass    | • Gradient  
             • Laplacian  
             • Roberts  
             • Sobel  
             • Prewitt |

### Gaussian

![Gaussian Image](image1)

### Median

![Median Image](image2)

### Gradient

![Gradient Image](image3)

### Sobel

![Sobel Image](image4)
Frequency Domain Filtering

- Standard filtering can be done in frequency domain
  - Low Pass, High Pass, Band Pass, Band Stop, etc.
- Compute the Fourier transform of the image
- Multiply with the transfer function of the filter
- Take the inverse Fourier transform to get the filtered image

\[ I(x,y) \rightarrow \text{FFT} \rightarrow F(u,v) \rightarrow H(u,v) \rightarrow \text{IFFT} \rightarrow R(x,y) \]

Input Image

Periodic noise

Bandstop filtered
Low Pass Filter Examples

- Low Pass with Frequency Domain Filter

- Low Pass with Gaussian Filter
High Pass Filtering Examples

- Detect edges
  ![Image](image1.png) ➔ ![Image](image2.png)

- Sharpen image
  ![Image](image3.png) ➔ ![Image](image4.png)
ENHANCE IMAGES:
IMAGE CALIBRATION
Spatial Calibration

- Corrects for lens and perspective distortion
- Allows the user to take real-world measurements from image based on pixel locations.
Calibrating Your Image Setup

• Acquire image of a calibration grid with known real-world distances between the dots
• Learn the calibration (mapping information) from its perspective and distortion
• Apply this mapping information to subsequent images and measurements
Image Correction

- Use calibration to adjust image geometry so features are represented properly.
Types of Calibration

- 2D spatial calibration
  - Applied only to a plane
  - Corrects for lens and perspective distortion
  - Does not improve resolution of a measurement
  - Cannot compensate for poor lighting or unstable conditions

- 3D spatial calibration: x, y, z

- Flat field calibration
  - Corrects for lighting variations

- Color
<table>
<thead>
<tr>
<th>Enhance</th>
<th>Check</th>
<th>Locate</th>
<th>Measure</th>
<th>Identify</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Filter noise or unwanted features</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Remove distortion</td>
<td>• Create Particles</td>
<td>• Match patterns</td>
<td>• Detect edges</td>
<td>• Read text (OCR)</td>
</tr>
<tr>
<td>• Calibrate images</td>
<td>• Measure intensity</td>
<td>• Match geometry</td>
<td>• Measure distance</td>
<td>• Read 1D barcodes</td>
</tr>
<tr>
<td></td>
<td>• Analyze particles</td>
<td>• Set-up coordinate systems</td>
<td>• Calculate geometry</td>
<td>• Read 2D codes</td>
</tr>
</tbody>
</table>
Histograms and Thresholding

- **Histogram**
  - Provides a quantitative distribution of pixels in an image
  - Indicates the number of pixels at each gray level

- **Thresholding**
  - Converts each pixel value in an image to 0 or 1 according to the value of the original pixel
  - Helps extract significant structures in an image
Histogram and Thresholding

Original Image

Histogram of Image

Binary Image
Finding Gray Objects

- Can also set upper and lower limits for pixel values
- Pixels inside the bounds of the limit (blue region) are set to 1, and those outside the limit are set to 0
Automatic Thresholding

Manual Threshold

Auto Threshold
Local Thresholding

A

Global Threshold

B

Local Threshold

C
Particles and Connectivity

• Thresholding creates binary image:
  – Pixels are either ‘0’ or ‘1’
• A particle in a binary image is a group of connected ‘1’ pixels
• Defining connectivity
  – Connectivity-4: two pixels are considered part of the same particle if they are horizontally or vertically adjacent
  – Connectivity-8: two pixels are considered part of the same particle if they are horizontally, vertically, or diagonally adjacent.
How Connectivity Affects a Particle

- How many particles with connectivity-4?
How Connectivity Affects a Particle

- How many particles with connectivity-4?
How Connectivity Affects a Particle

- How many particles with connectivity-4?

- How many particles with connectivity-8?
How Connectivity Affects a Particle

- How many particles with connectivity-4?

- How many particles with connectivity-8?

![Diagram showing connectivity-4 and connectivity-8 particles]
Binary Morphology

- Binary morphological functions extract and alter the structure of particles in a binary image.
- Morphological functions remove unwanted information caused by the thresholding process:
  - Noise particles
  - Particles touching the border of an image
  - Particles touching each other
  - Particles with uneven borders
Erosion

- Decreases the size of objects in an image
  - Removes a layer of pixels along the boundary of the particle
  - Eliminates small isolated particles in the background and removes narrow peninsulas

- Use Erode to:
  - Separate particles for counting
Dilation

• Increases the size of objects in an image
  – Adds a layer of pixels around the boundary of an object (including the inside boundary for objects with holes)
  – Eliminates tiny holes in objects
  – Removes gaps or bays of insufficient width
• Use Dilate to:
  – Make particles touch
Erosion vs. Dilation

Erosion

Dilation
• An erosion followed by a dilation
  – Remove small particles and smooth boundaries
  – Does not significantly alter the size or shape of particles
  – Borders removed by the erosion process are replaced by the dilation process

• Use Open To:
  – Eliminate small particles that constitute noise
• A dilation followed by an erosion
  – Fills holes and creates smooth boundaries
  – Does not alter the size or shape of particles
  – Particles that do not connect after the dilation are not changed

• Use Close To:
  – Eliminate small holes that constitute noise
• Advanced morphological functions are combinations of operations, each of which is responsible for a single operation

• These functions execute the following tasks:
  – Separate particles
  – Remove small or large particles
  – Keep or remove particles identified by morphological parameters
  – Fill holes
  – Remove particles from an image border
  – Segmenting the image
Particle Filtering

- Keeps or removes particles based on geometric features
  - Area, length, aspect ratio and other features are commonly used to filter
- Typically used on binary images
- Cleans up noisy images
Application: Analyze Particles

Original

Threshold

Particle Filter
Class Organization

**Enhance**
- Filter noise or unwanted features
- Remove distortion
- Calibrate images

**Check**
- Create Particles
- Measure intensity
- Analyze particles

**Locate**
- Match patterns
- Match geometry
- Set-up coordinate systems

**Measure**
- Detect edges
- Measure distance
- Calculate geometry

**Identify**
- Read text (OCR)
- Read 1D barcodes
- Read 2D codes
LOCATING PARTS:
PATTERN MATCHING
Introduction to Matching

• Locates regions of a grayscale image that match a predefined template
  – Calculate a score for each matching region
  – Score indicates quality of match

• Returns the XY coordinates, rotation angle and scale for each match
Applications

• Presence Detection
• Counting
• Alignment
• Inspection
How It Works

• Two step process:
  – Step 1: Learn Template
    • Extract information useful for uniquely characterizing the pattern
    • Organize information to facilitate faster search of the pattern in the image
  – Step 2: Match
    • Use the information present in the template to locate regions in the target image
    • Emphasis is on search methods that quickly locate matched regions
Pattern Matching Methods

• Different ways to perform pattern matching based on the information extracted from the template

• Two common methods:
  – Correlation Pattern Matching
    • Relies on the grayscale information in the image for matching
  – Geometric Pattern Matching
    • Relies on edges and geometric features in the image for matching
Correlation Pattern Matching

- Grayscale information present in the image
- Directly uses the underlying grayscale distribution in the image for matching
- Grayscale values in the pattern are matched to regions in the image using normalized cross-correlation
- Score ranges from 0-1000
  - Used to allow imperfect match
Correlation Pattern Matching

- When to use:
  - Template primarily characterized by grayscale information
  - Matching under uniform light changes
  - Little occlusion and scale changes in image
  - Good for the general case

Good template

Bad template
Correlation Pattern Matching

• When NOT to use:
  – Non-uniform lighting
  – Occlusion more than 10%
  – Scale changes
Geometric Pattern Matching

- Matching tool you can use to locate parts that contain distinct edge information
- Not useful when template is predominantly defined by texture
GPM is Tolerant to...

- Occlusion
- Scale Changes
- Non-uniform Lighting
- Background Changes
GPM – Feature-based

Image → Extract Curves → Extract Features → Match Features

Template

circles

Target Image

parallel lines
## Feature Comparison

<table>
<thead>
<tr>
<th>Feature</th>
<th>CPM</th>
<th>GPM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Template contains texture-like information</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Template contains geometric information</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Find multiple match locations</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Rotation</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Scale</td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Occlusion</td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Matching under non-uniform lighting</td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Sub-pixel match locations</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
LOCATING PARTS:
COORDINATE SYSTEMS
Region of Interest

- Region of Interest
  - A portion of the image upon which an image processing step may be performed
  - Can be defined statically (fixed)
  - Or dynamically: based on features located in the image

- Used to
  - Process only pixels that are interesting
Coordinate Systems

• Defined by a reference point (origin) and angle within the image, or by the lines that make up its axes
• Allows you to define search areas that can move around the image with the object you are inspecting
• Usually based on a characteristic feature of the object under inspection
  – Use pattern match to locate features
  – Use features to establish coordinate system
Coordinate Systems – Set Up

- Define an origin
  - Locate an easy-to-find feature in your reference image
  - Set a coordinate system based on its location and orientation
- Set up measurement ROIs in reference to the new origin
- Acquire a new image
- Locate reference point
- Adjust measurement ROIs
Class Organization

Enhance
- Filter noise or unwanted features
- Remove distortion
- Calibrate images

Check
- Create Particles
- Measure intensity
- Analyze particles

Locate
- Match patterns
- Match geometry
- Set-up coordinate systems

Measure
- Detect edges
- Measure distance
- Calculate geometry

Identify
- Read text (OCR)
- Read 1D barcodes
- Read 2D codes
Edge Detection Overview

- Process of detecting transitions in an image
- One of the most commonly used machine vision tools
- Attractive because:
  - Simple to understand and use
  - Localized processing – fast
  - Applicable to many applications

Different Illuminations

Different Edges
1D Edge Detection

- Detect edge points along a line
- Basic operation:
  - Get pixel values along the line
  - Compute gradient information
  - Peaks and valleys represent edge locations
  - Get first, first & last, best, or all edges

![Graph showing pixel values and gradient values along a line](image)
Subpixel Accuracy

- Subpixel location of edge can be computed using parabolic interpolation.
Edge Detector Tools

- High level tools based on the edge detectors
- Rake:
  - Used to find multiple edges and fit a shape through them
- Configurable search directions, sub-sampling ratios, and display settings
Straight Edge (Line) Detection

- Detect straight lines in an image
  - Extension of 1D edge detection
- Straight edge detection options:
  - Rake-based
  - Projection-based
Edge Detection Applications

- Detect Features
- Alignment
- Gauging
- Inspection
Application: Locating Parts

Failed due to intensity
Application: Dimension Verification

- Dimensional measurements, such as lengths, distance, and diameter
  - Inline gauging inspections are used to verify assembly and packaging routines
  - Offline gauging is used to judge product quality according to a sample
Class Organization

Enhance
- Filter noise or unwanted features
- Remove distortion
- Calibrate images

Check
- Create Particles
- Measure intensity
- Analyze particles

Locate
- Match patterns
- Match geometry
- Set-up coordinate systems

Measure
- Detect edges
- Measure distance
- Calculate geometry

Identify
- Read text (OCR)
- Read 1D barcodes
- Read 2D codes
Identify

• 1D and 2D Codes
• Marking methods
• Reading
• Examples
1D Codes

- Applications using 1D bar codes have been around for over 35 years
- Barcode data is an index into a large central data storage
- Code is easily read by laser scanners
- Low data capacity in large footprint

Code 3 of 9
Code 128
EAN 13
2D Codes

• Usually not an index into a database
• Camera-based vision systems are preferred reading method
• High data capacity in small footprint

Data Matrix  PDF 417  QR Code
## 1D vs 2D

<table>
<thead>
<tr>
<th>1D Codes</th>
<th>2D Codes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low data capacity</td>
<td>High data capacity</td>
</tr>
<tr>
<td>Index into large database</td>
<td>Self contained data</td>
</tr>
<tr>
<td>Large footprint</td>
<td>Small footprint</td>
</tr>
<tr>
<td>Redundancy in Y dimension</td>
<td>Error correction capability</td>
</tr>
<tr>
<td>Readable by laser scanner</td>
<td>Requires camera based reader</td>
</tr>
<tr>
<td>Requires as much as 80% contrast</td>
<td>Can be read in low contrast</td>
</tr>
</tbody>
</table>

A 52 mm Data Matrix (approx. 3/10\textsuperscript{th} of an inch) can contain 400 characters of information.
Optical Character Recognition
Particle-based OCR

Typical steps:
• Region of interest around line of text
• Threshold
• Character segmentation
• Compare to library
• Character is learned or recognized
Class Organization

Enhance
- Filter noise or unwanted features
- Remove distortion
- Calibrate images

Check
- Create Particles
- Measure intensity
- Analyze particles

Locate
- Match patterns
- Match geometry
- Set-up coordinate systems

Measure
- Detect edges
- Measure distance
- Calculate geometry

Identify
- Read text (OCR)
- Read 1D barcodes
- Read 2D codes
Dr. Romik Chatterjee  
Vice President of Engineering

Graftek Imaging Inc.  
8900 Shoal Creek Boulevard  
Building 300, Suite B  
Austin, Texas 78757  
USA

Phone: +1 512-416-1099  
Email: romik@graftek.com

www.graftek.com