K. K. Wagh Institute of Engineering Education and Research, Nasik

Robot Vision -I

Dr. Padmakar J. Pawar

Professor and Head

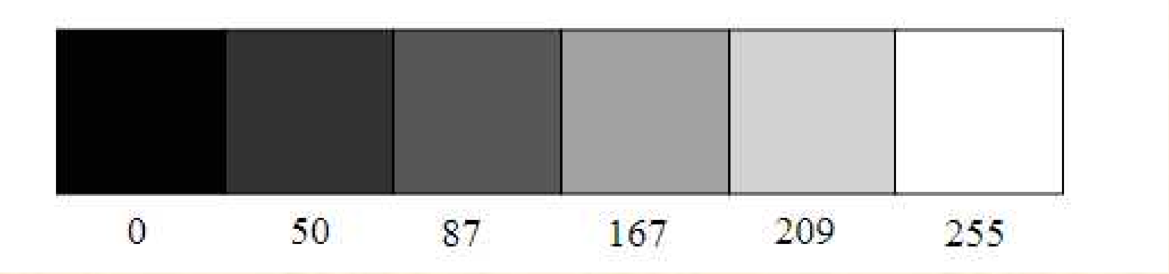
Department of Robotics and Automation

Introduction

- A robot vision system consists of one or more cameras, software and a robot. The camera takes picture of the working area or object to grip and software is used to analyze the image to identify its position, orientation, identity, and condition.
- This high level information then can be used to plan the robot motions such as how to grasp the part, how to avoid collisions etc.
- With effective use of robot vision system, solutions to automated assembly tasks can be made robust and more reliable

Image representation

- An image could be black and white, 8 bit colour format (Greyscale image with 256 shades), 16 bit colour format (RGB Format with 65536 shades of colour)



- A raw two dimensional image is regarded as analog function of reflected light intensity.
- If the image is to be processed with computer, the analog signals must be converted into equivalent digital representation.

Digitization of image

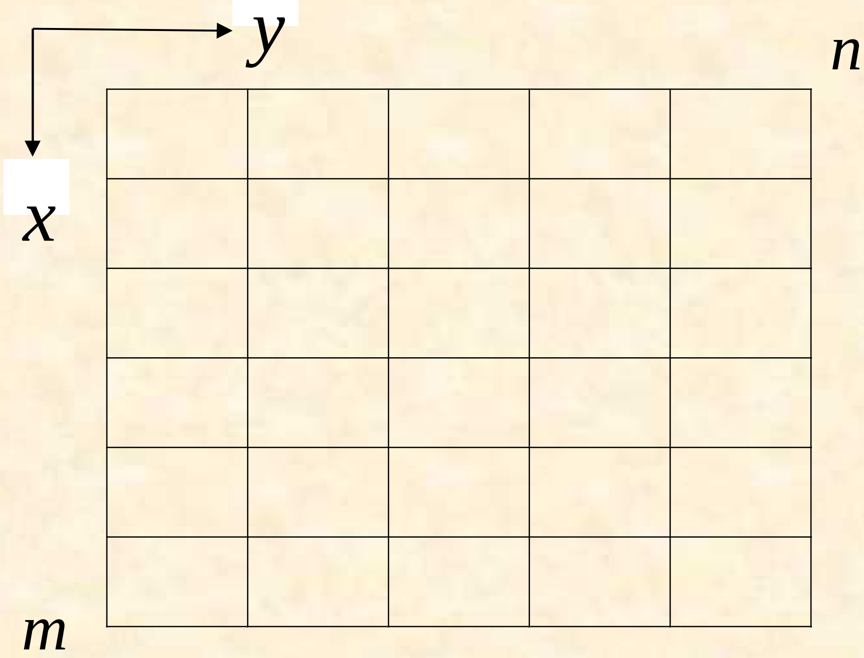
A digitization process is often used to convert analog data, such as sound, image, and text, into a numerical representation.

Steps of digitization of image:

1) Sampling

The data are sampled at regular intervals, such as the grid of pixels used to represent a digital image. The frequency of sampling is referred to as resolution of the image. Sampling turns continuous data (analog) into discrete (digital) data.

Digitized image (Sampling)



2) Quantification:

Each sample is then quantified, i.e. assigned a numerical value drawn from a defined range such as 0-255 in the case of a 8-bit gray scale image.

Techniques of image analysis

- Template matching
- Edge and corner point detection
- Shape analysis
- Segmentation
- Run length encoding

Template Matching

It is a technique to recognize whether or not a given part is a member of particular class of parts.

Main Image

Template Image

Result of template matching

Template $n0 \longrightarrow y$ Image

$m0$ x

0	5	1
8	9	3
8	6	3

3	7	7	7	7
1	5	2	7	9
4	0	0	4	0
0	0	9	9	3
7	3	8	6	3
7	3	9	8	5

m n

Perfomance Matrix

7	7	7
5	2	7
0	0	4

7	2	6
3	7	4
8	6	1

Performance index (ρ) **44**

S. N.	Translation	P. I. (ρ)
1	$\rho(0,0)$	36
2	$\rho(0,1)$	44
3	$\rho(0,2)$	42
4	$\rho(1,0)$	38
5	$\rho(1,1)$	49
6	$\rho(1,2)$	32
7	$\rho(2,0)$	42
8	$\rho(2,1)$	32
9	**$\rho(2,2)$**	**3**
10	$\rho(3,0)$	35
11	$\rho(3,1)$	34
12	$\rho(3,2)$	23

$$\rho\,(x,y) \triangleq \sum_{k=1}^{m0}\sum_{j=1}^{n0} |I(k+x, j+y) - T_i(k,j)|$$

Normalised cross correlation

Although template matching is appealing in view of its conceptual simplicity, this technique requires that the two images being compared should have same average intensities. However, the two images of same part often have average intensities that differ due to variation in the background or lighting conditions. Thus an important limitation of this technique is that it is sensitive to average light intensities.

To overcome this limitation an alternative approach known as normalised cross correlation can be used.

Image

3	7	7	7	7
1	5	7	7	9
4	0	0	4	0
0	0	9	9	3
7	3	8	6	3
7	3	9	8	5

0	4	0
9	9	3
8	6	3

0	20	0
72	81	9
64	36	9

$\Sigma T \cdot I =$ 291

$\sigma =$ 0.995

Avg. Intensity of image

0	16	0
81	81	9
64	36	9

17.205

Avg. Intensity of Template

0	25	1
64	81	9
64	36	9

17

$\|I\|.\|T\| =$ 292.4791

S. N.	Translatio	NCC
1	σ(0,0)	0.632
2	σ(0,1)	0.504
3	σ(0,2)	0.587
4	σ(1,0)	0.449
5	σ(1,1)	0.404
6	σ(1,2)	0.706
7	σ(2,0)	0.497
8	σ(2,1)	0.701
9	σ(2,2)	0.995
10	σ(3,0)	0.69

$$\|I_{x,y}\| \triangleq \left[\sum_{k=1}^{m0}\sum_{j=1}^{n0} I^2(k+x, j+y)\right]^{\frac{1}{2}}$$

$$\|T_i\| \triangleq \left[\sum_{k=1}^{m0}\sum_{j=1}^{n0} T_i^2(k,j)\right]^{\frac{1}{2}}$$

$$\sigma_i(x,y) = \frac{\sum_{k=1}^{m0}\sum_{j=1}^{n0}(k+x, j+y).T_i(k,j)}{\|I_{x,y}\|.\|T_i\|}$$

The criteria for addition of pixels in region growing

It can be based on any characteristic of the regions in the image such as

- Average intensity
- Variance
- Colour
- Texture
- Motion
- Shape
- Size

Region Growing

It a important technique for image segmentation In which image is partitioned into multiple segments to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze.

Start with a single pixel (seed) and add new pixels slowly

Seed Point								
8	3	4	0	0	2	4	4	2
	3	1	0	2	1	3	4	1
Threshold	2	0	4	4	1	2	1	5
4	4	3	2	0	3	2	4	8
	2	4	3	2	1	1	1	6
	7	6	8	8	8	8	0	5
	6	8	6	7	7	5	4	5
	8	7	6	7	6	8	2	8

1	2	3	4	5	6	7
0	1	1	1	0	1	0
1	0	4	4	3	1	1
4	0	1	1	1	0	5
4	4	2	2	2	3	5
4	0	4	2	2	2	5
8	6	6	6	8	6	5
7	7	7	8	6	7	5
6	7	6	7	8	7	1

5	4	8	8	6	4	4	6
5	7	8	6	7	5	4	7
6	8	4	4	7	6	7	3
4	5	6	8	5	6	4	0
6	4	5	6	7	7	7	2
1	2	0	0	0	0	8	3
2	0	2	1	1	2	4	3
0	1	2	1	2	0	6	0

Robot Vision -II

Dr. Padmakar J. Pawar

Professor and Head
Department of Production Engineering

Digitized image

y n

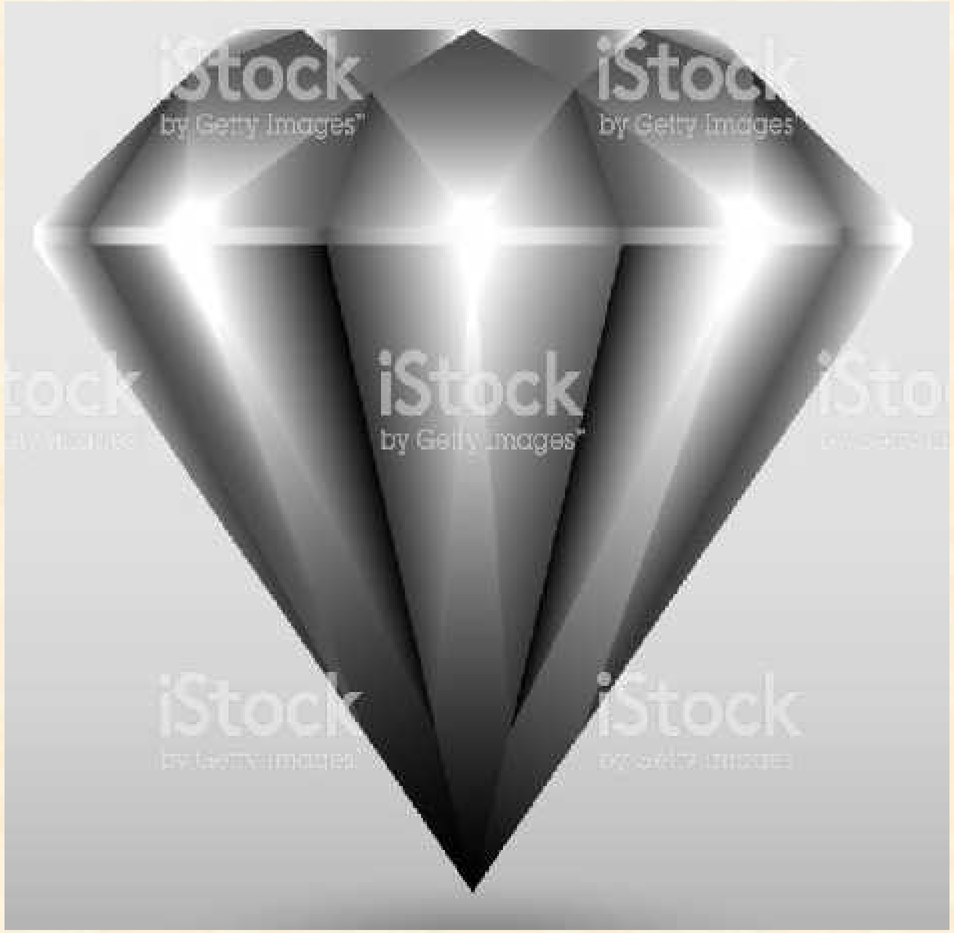
x

3	7	7	7	7
1	5	2	7	9
4	0	0	4	0
0	0	9	9	3
7	3	8	6	3
7	3	9	8	5

m

Edge detection:

If the faces of the object are smooth and homogeneous, then the intensity of the reflected light will be constant or at least will vary uniformly. However, at the boundary of the adjacent faces, the reflected light intensity will have a jump discontinuity. Therefore at the edge the gradient of light intensity will be infinite. Hence the gradient operator can be used to locate the edge.


iStock
by Getty Images

Edge detection using gradients

x

y

P1	P2	P3
P4	P5	P6
P7	P8	P8

$$\nabla = \begin{Bmatrix} \dfrac{\partial f}{dx} \\ \dfrac{\partial f}{\partial y} \end{Bmatrix}$$

$$\nabla f(x) = (p3 - p1) + (C.P6 - C.P4) + (P8 - P7)$$

$$\nabla f(y) = (p7 - p1) + (C.P8 - C.P2) + (P8 - P3)$$

C = 1, Prewitt operator:

$$Mx = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix};\ My = \begin{bmatrix} -1 & -1 & 1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

C = 2, Sobel Operator

$$Mx = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix};\ My = \begin{bmatrix} -1 & -2 & 1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

M_x

-1	0	1
-1	0	1
-1	0	1

M_y

-1	-1	-1
0	0	0
1	1	1

Image

7	5	3
8	9	1
8	6	2

M1

-7	0	3
-8	0	1
-8	0	2

M2

-7	-5	-3
0	0	0
8	6	2

Avg. I:	5.444

Fx	5.667

Fy	1

∇F	5.754

Moments:

Image moments are useful to describe objects after segmentation. Some important properties of the image which are found *via* image moments include area (or total intensity), its centroid, and information about its orientation.

0,0	1	2	3	4	5	6	7	8	9	10
1	1	0	1	0	0	0	0	0	0	0
2	1	0	1	1	1	1	0	1	0	0
3	0	0	1	0	1	0	0	0	0	1
4	1	1	1	0	0	1	0	1	0	1
5	0	1	1	0	1	1	0	0	1	1
6	1	0	1	0	1	0	1	0	0	0
7	0	1	1	0	0	0	1	1	1	0
8	0	0	0	0	0	0	0	0	0	0

y, x

$$m_{kl} = \sum M_x^k . M_y^l$$

$$A = m_{00} = 32$$

$$x_c = \frac{m_{10}}{m_{00}} = 4.25$$

$$y_c = \frac{m_{01}}{m_{00}} = 4.9688$$

0,0

Mx

1	0	1	0	0	0	0	0	0	0
2	0	2	2	2	2	0	2	0	0
0	0	3	0	3	0	0	0	0	3
4	4	4	0	0	4	0	4	0	4
0	5	5	0	5	5	0	0	5	5
6	0	6	0	6	0	6	0	0	0
0	7	7	0	0	0	7	7	7	0
0	0	0	0	0	0	0	0	0	0

SET	*k*	*l*	*mkl*	m00	m10	m01	m11	m20	m02
	0	0		32	136	159	703	638	1055

0,0

My

1	0	3	0	0	0	0	0	0	0
1	0	3	4	5	6	0	8	0	0
0	0	3	0	5	0	0	0	0	10
1	2	3	0	0	6	0	8	0	10
0	2	3	0	5	6	0	0	9	10
1	0	3	0	5	0	7	0	0	0
0	2	3	0	0	0	7	8	9	0
0	0	0	0	0	0	0	0	0	0

m

1	2	3	4	5	6	7	8	9	10
1	0	3	0	0	0	0	0	0	0
1	0	3	4	5	6	0	8	0	0
0	0	3	0	5	0	0	0	0	10
1	2	3	0	0	6	0	8	0	10
0	2	3	0	5	6	0	0	9	10
1	0	3	0	5	0	7	0	0	0
0	2	3	0	0	0	7	8	9	0
0	0	0	0	0	0	0	0	0	0

x \ y	1	2	3	4	5	6	7	8	9	10
1	1	0	1	0	0	0	0	0	0	0
2	1	0	1	1	1	1	0	1	0	0
3	0	0	1	0	1	0	0	0	0	1
4	1	1	1	0	0	1	0	1	0	1
5	0	1	1	0	1	1	0	0	1	1
6	1	0	1	0	1	0	1	0	0	0
7	0	1	1	0	0	0	1	1	1	0
8	0	0	0	0	0	0	0	0	0	0

$$\mu_{kj} = \sum M_{xc}^{k} . M_{yc}^{j}$$

0,0

Mx

1	0	1	0	0	0	0	0	0	0
2	0	2	2	2	2	0	2	0	0
0	0	3	0	3	0	0	0	0	3
4	4	4	0	0	4	0	4	0	4
0	5	5	0	5	5	0	0	5	5
6	0	6	0	6	0	6	0	0	0
0	7	7	0	0	0	7	7	7	0
0	0	0	0	0	0	0	0	0	0

µ02

Moment of inertia corresponding to x axis

µ20

Moment of inertia corresponding to y axis

SET	*k*	*j*	*µ kj*	µ02	µ20
	2	0		265	110

0,0

My

1	0	3	0	0	0	0	0	0	0
1	0	3	4	5	6	0	8	0	0
0	0	3	0	5	0	0	0	0	10
1	2	3	0	0	6	0	8	0	10
0	2	3	0	5	6	0	0	9	10
1	0	3	0	5	0	7	0	0	0
0	2	3	0	0	0	7	8	9	0
0	0	0	0	0	0	0	0	0	0

µ

1	2	3	4	5	6	7	8	9	10
10.563	0	10.563	0	0	0	0	0	0	0
5.0625	0	5.0625	5.0625	5.0625	5.0625	0	5.0625	0	0
0	0	1.5625	0	1.5625	0	0	0	0	1.5625
0.0625	0.0625	0.0625	0	0	0.0625	0	0.0625	0	0.0625
0	0.5625	0.5625	0	0.5625	0.5625	0	0	0.5625	0.5625
3.0625	0	3.0625	0	3.0625	0	3.0625	0	0	0
0	7.5625	7.5625	0	0	0	7.5625	7.5625	7.5625	0
0	0	0	0	0	0	0	0	0	0

Mxc

-3.25	0	-3.25	0	0	0	0	0	0	0
-2.25	0	-2.25	-2.25	-2.25	-2.25	0	-2.25	0	0
0	0	-1.25	0	-1.25	0	0	0	0	-1.25
-0.25	-0.25	-0.25	0	0	-0.25	0	-0.25	0	-0.25
0	0.75	0.75	0	0.75	0.75	0	0	0.75	0.75
1.75	0	1.75	0	1.75	0	1.75	0	0	0
0	2.75	2.75	0	0	0	2.75	2.75	2.75	0
0	0	0	0	0	0	0	0	0	0

Myc

-3.97	0	-1.97	0	0	0	0	0	0	0
-3.97	0	-1.97	-0.97	0.03	1.03	0	3.03	0	0
0	0	-1.97	0	0.03	0	0	0	0	5.03
-3.97	-2.97	-1.97	0	0	1.03	0	3.03	0	5.03
0	-2.97	-1.97	0	0.03	1.03	0	0	4.03	5.03
-3.97	0	-1.97	0	0.03	0	2.03	0	0	0
0	-2.97	-1.97	0	0	0	2.03	3.03	4.03	0
0	0	0	0	0	0	0	0	0	0

Run Length Encoding

Since high resolution images contain many pixels, they require considerable amount of storage space and take long time to transmit over communication channel. A technique of compression of a binary image based on encoding is called run length encoding.

1	1	1	1	1	1	1	1
1	1	1	1	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1
1	1	1	1	1	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1
1	1	1	1	1	1	1	1

Run	Bit Value	Length	Stringlength
1	1	12	4
2	0	18	5
3	1	7	3
4	0	17	5
5	1	10	4
		64	**5**

Length	Bit	Total
01100	1	6
10010	0	6
00111	1	6
10001	0	6
01010	1	6
		30

Practice problems 1:

Determine the area, centroid, and moment of inertia of the following image.

y →

x ↓ 0,0	1	2	3	4	5	6	7	8	9	10	11	12
1	0	1	0	1	1	0	0	1	0	0	1	1
2	0	1	1	1	0	1	1	1	1	1	1	1
3	0	0	0	1	0	0	1	1	0	0	0	1
4	1	1	1	0	0	0	0	0	1	1	0	0
5	1	0	1	1	1	0	1	0	1	0	0	0
6	0	1	1	1	0	0	0	0	1	1	1	0
7	0	1	0	0	0	0	0	1	0	0	0	0
8	0	1	0	1	1	0	1	1	1	0	0	0
9	1	0	0	1	1	0	1	1	1	0	1	0
10	0	1	0	1	1	0	0	0	0	1	0	0

Practice problems 2:

Determine the compression ratio for the following image using run length encoding

	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	1
	1	1	1	1	1	1	1	1	1	1
	1	1	1	1	1	1	1	1	1	1
	1	1	1	1	0	0	0	0	0	0
	0	1	1	1	1	1	1	1	1	1
	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	1	1	1	1	1	1
	1	1	1	1	1	1	1	1	1	1

Further readings:

1. Saha S. K. (2014) Introduction to Robotics, McGraw Hill Education Pvt. Ltd, New Delhi, ISBN: 978-93-3290-280-0

2. Schilling R. J. (2009) Fundamentals of Robotics: Analysis and Control, PHI Learning Pvt. Ltd, New Delhi, ISBN: 978-81-203-1047-6

Thank You!