

# Vision

Some material is from  
<http://aima.eecs.berkeley.edu/slides-tex/>

<b>Introduction</b>	<b>2</b>
Introduction	2
More Vision Processing Stages	3
Pinhole Camera Model	4
Pinhole Camera Properties	5
Stapler Image	6
Stapler Square	7
<b>Edge Detection</b>	<b>8</b>
Introduction to Edge Detection	8
Smoothing and Differentiation Example	9
Convolution	10
Sobel Operators	11
Arctangent Result	12
<b>Shape Recovery</b>	<b>13</b>
Vision from Multiple Images	13
Shape from Stereo	14
Shape from Motion	15
Optical Flow	16
Optimal Flow Example	17
Example Continued	18
Shape from Texture	19
Texture Example	20
<b>Object Recognition</b>	<b>21</b>
Introduction to Object Recognition	21
Shape-context Matching	22

## Introduction

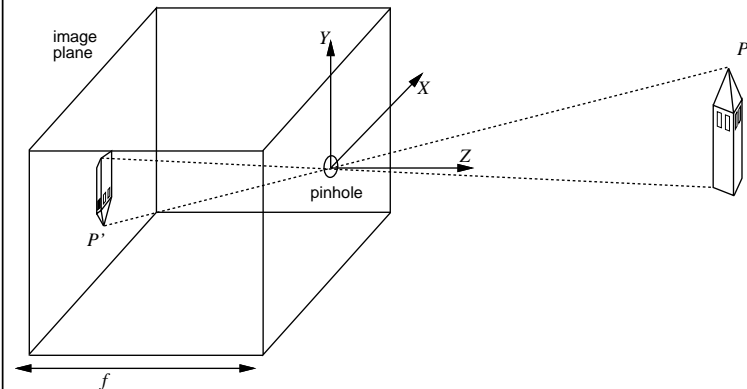
### Introduction

- Vision is the problem of recovering information from *images*.
- Vision involves geometry (perspective, motion), physics (optics), neuroscience (eyes, optic nerve, visual cortex), and psychology (what people perceive).
- Digital images: Light projected on a plane results in a 2D array of *pixels*.
- Edge detection: *Discontinuities* in surfaces result in discontinuities in an image.

### More Vision Processing Stages

- Segmentation: Continuous surfaces result in continuous regions in an image.
- Shape from stereo and motion: Detecting the same surface in multiple images, recovering location and movement.
- Recognition: Matching the shapes recovered from image(s) with the shapes of known objects.

### Pinhole Camera Model



## Pinhole Camera Properties

- Light reflects off objects into a pinhole and is projected onto an image plane, a distance  $f$  from the pinhole.
- A point  $P = (X, Y, Z)$  projects on a line through  $(0, 0, 0)$  and onto  $(x, y, -f)$  on the image plane. This line also intersects:

$$\left(\frac{X}{Z}, \frac{Y}{Z}, \frac{Z}{Z}\right) = \left(\frac{x}{-f}, \frac{y}{-f}, \frac{-f}{-f}\right)$$

- Solving for  $x$  and  $y$  results in:

$$x = -\frac{fX}{Z} \quad y = -\frac{fY}{Z}$$

CS 5233 Artificial Intelligence

Vision - 5

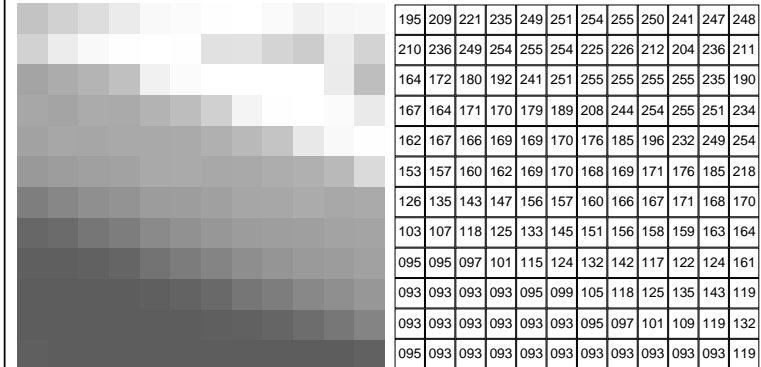
## Stapler Image



CS 5233 Artificial Intelligence

Vision - 6

## Stapler Square



CS 5233 Artificial Intelligence

Vision - 7

## Edge Detection

8

### Introduction to Edge Detection

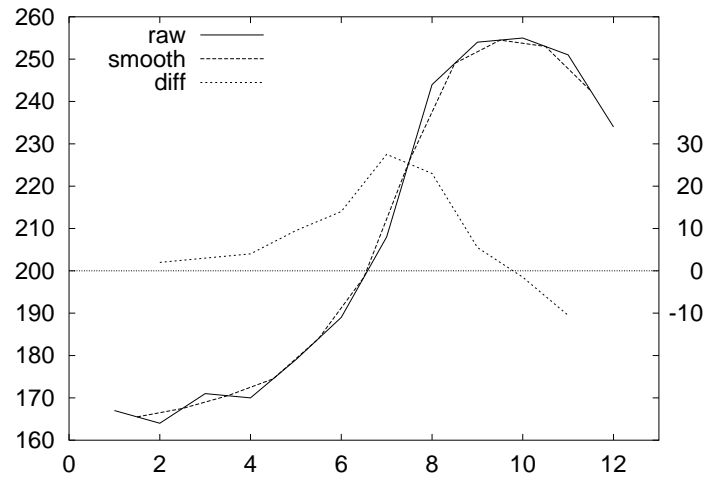
- *Edge detection* tries to find discontinuities in the scene (depth, orientation, reflectance, illumination).
- Edge detection usually involves *smoothing* (or *filtering*) to reduce noise.
- *Gaussian* smoothing of a point is a weighted average of a neighborhood of points with higher weights towards the center.
- Smoothing is followed by *differentiation* to find differences in intensity.
- *Discrete differentiation* is the difference between two neighboring points.

CS 5233 Artificial Intelligence

Vision - 8

## Smoothing and Differentiation Example

The "raw" curve is one row from Figure 24.4.  
 The "smooth" curve averages two pixels.  
 The "diff" curve is the difference of two points.



CS 5233 Artificial Intelligence

Vision - 9

## Convolution

- $G(\cdot)$  is a Gaussian function.  
 $G'(\cdot)$  is its derivative.  
 $I(x, y)$  is the image.
- Convolution for vertical and horizontal edges:
 
$$V(x, y) = \sum_{i,j} G'(i)G(j)I(x - i, y - j)$$

$$H(x, y) = \sum_{i,j} G'(j)G(i)I(x - i, y - j)$$
- Obtain edge pixels and orientation by:
 
$$E(x, y) = (V(x, y)^2 + H(x, y)^2 > \tau)$$

$$O(x, y) = \arctan(-H(x, y), V(x, y))$$
- Infer "clean" line segments by combining edge pixels with same orientation.

CS 5233 Artificial Intelligence

Vision - 10

## Sobel Operators

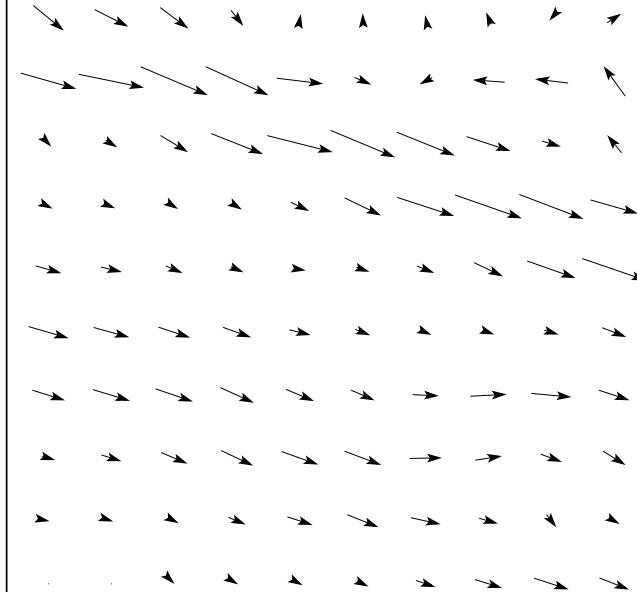
The *Sobel operators* are *masks* that approximate convolution.

Vertical	Horizontal																		
$g_1$	$g_2$																		
<table border="1" style="border-collapse: collapse;"> <tr><td>-1</td><td>0</td><td>1</td></tr> <tr><td>-2</td><td>0</td><td>2</td></tr> <tr><td>-1</td><td>0</td><td>1</td></tr> </table>	-1	0	1	-2	0	2	-1	0	1	<table border="1" style="border-collapse: collapse;"> <tr><td>1</td><td>2</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>-1</td><td>-2</td><td>-1</td></tr> </table>	1	2	1	0	0	0	-1	-2	-1
-1	0	1																	
-2	0	2																	
-1	0	1																	
1	2	1																	
0	0	0																	
-1	-2	-1																	

CS 5233 Artificial Intelligence

Vision - 11

## Arctangent Result



CS 5233 Artificial Intelligence

Vision - 12

# Shape Recovery

13

## Vision from Multiple Images

- If we can match features in multiple images from different locations and times, then location and movement can be recovered.
- *Intensity-based matching* tries to match regions of one image to another.
- The sum of square differences and the cross-correlation are common measures of similarity:

$$SSD = \sum_{(i,j) \in R} (L(i, j) - R(i - \Delta x, j - \Delta y))^2$$

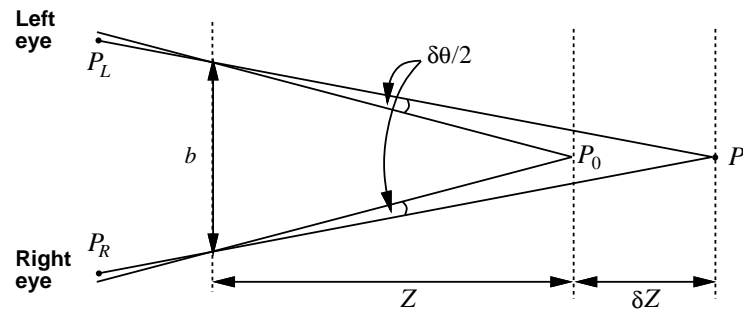
$$CC = \sum_{(i,j) \in R} L(i, j)R(i - \Delta x, j - \Delta y)$$

CS 5233 Artificial Intelligence

Vision - 13

## Shape from Stereo

- In stereo vision, we get another image at the same time at a different location.
- The *epipolar constraint* states that a given feature on one image plane must lie on a line in the other image plane.



CS 5233 Artificial Intelligence

Vision - 14

## Shape from Motion

- The problem of *shape from motion* is to recover the motion of the observer and the shape of the scene from a series of images.
- *Optical flow* describes apparent motion on the image plane.
- If an observer moves straight toward (away from) a point, that point is the *focus of expansion* (or *contraction*).
- The *aperture problem* is that optical flow can be detected perpendicular to an edge, but not parallel to it.

CS 5233 Artificial Intelligence

Vision - 15

## Optical Flow

- If a point on the image plane moves with velocity  $(u, v)$  from times  $t$  to  $t + \Delta t$ , then:

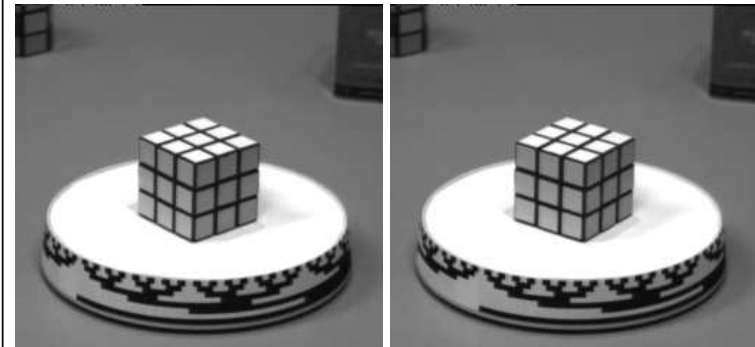
$$I(x, y, t) \approx I(x + u\Delta t, y + v\Delta t, t + \delta t)$$

- Further, we expect  $(u, v) \approx (u', v')$  at the next time point.
- Optical flow is determined by finding the  $(u, v)$  that leads to the best approximations.
- From this information, we can determine the observer's motion (assuming most things stay put), and then distance and motion in the scene.

CS 5233 Artificial Intelligence

Vision - 16

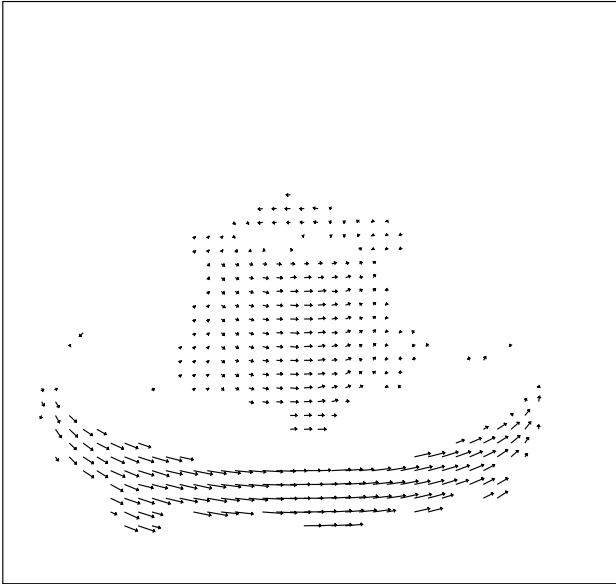
## Optical Flow Example



CS 5233 Artificial Intelligence

Vision - 17

## Example Continued



CS 5233 Artificial Intelligence

Vision – 18

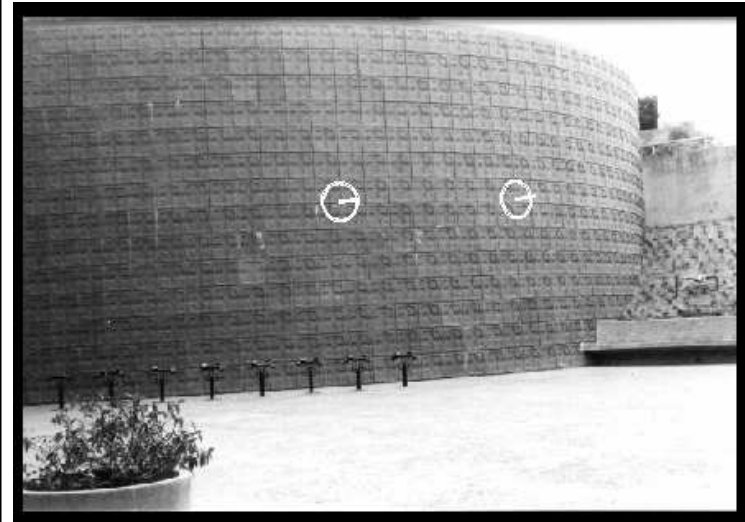
## Shape from Texture

- Idea: assume actual texture is uniform.
- Compute surface shape that would produce this distortion.
- A similar idea works for shading by assuming uniform reflectance.

CS 5233 Artificial Intelligence

Vision – 19

## Texture Example



CS 5233 Artificial Intelligence

Vision – 20

## Object Recognition

21

### Introduction to Object Recognition

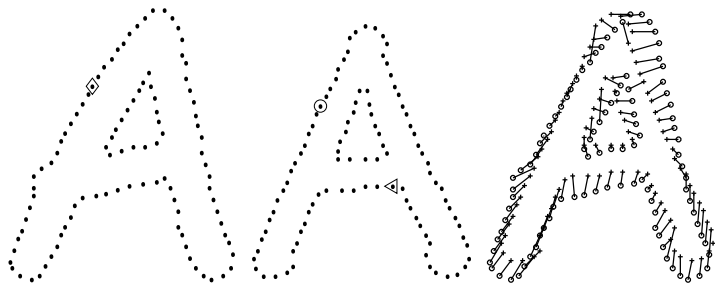
- Simple idea:
  - extract 3-D shapes from image
  - match against “shape library”
- Problems:
  - extracting and representing shape of object
  - representing shape and variability in library
  - occlusion, shadows, markings, noise, ...
- Approaches:
  - index by invariant properties of objects
  - alignment of features with library features
  - match image against multiple stored views
  - machine learning methods

CS 5233 Artificial Intelligence

Vision – 21

## Shape-context Matching

Determine total distance between shapes by sum of distances for corresponding points under best matching.



Attributes  
from Image

Attributes  
from Library

Attribute  
Matching