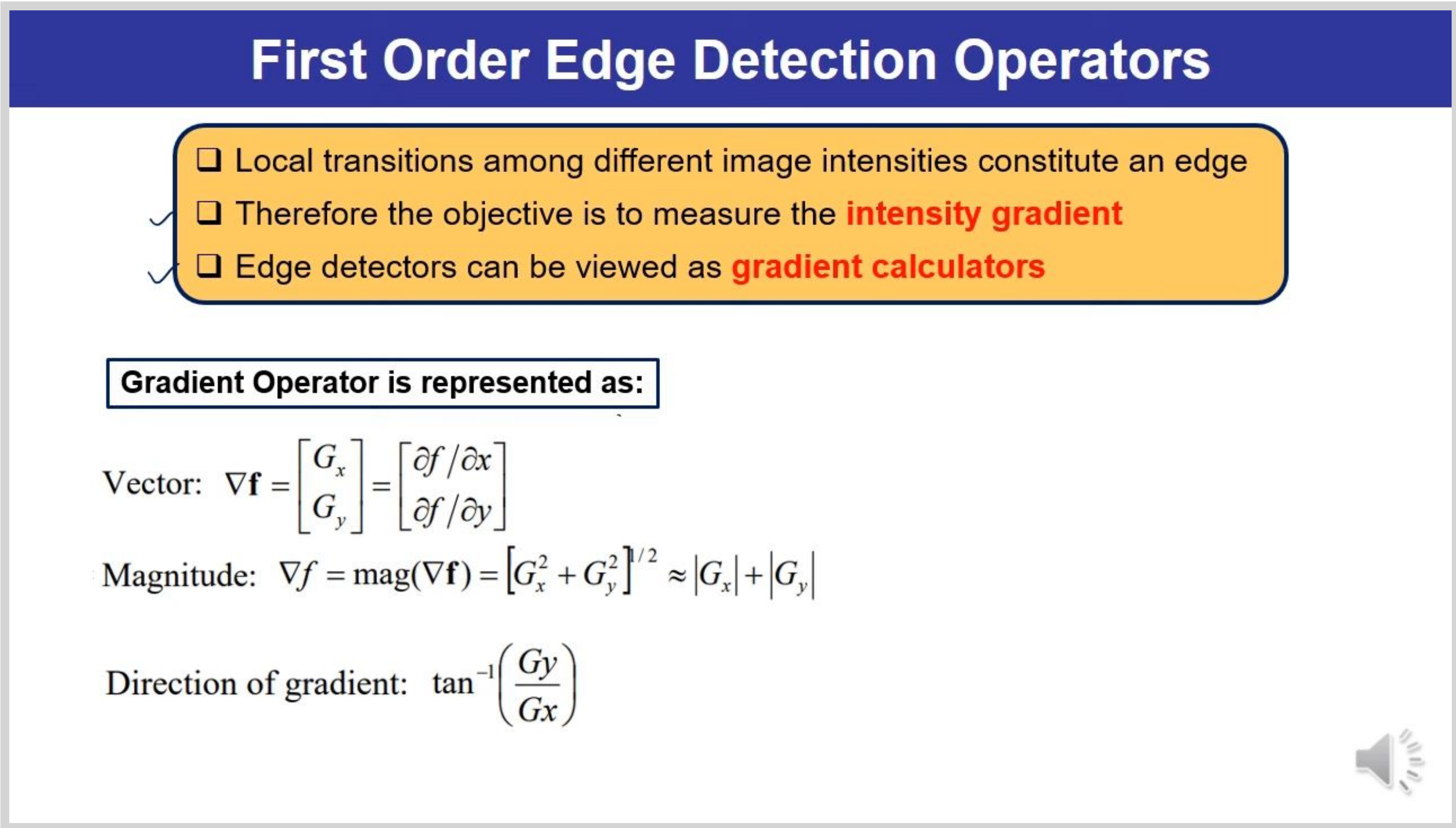
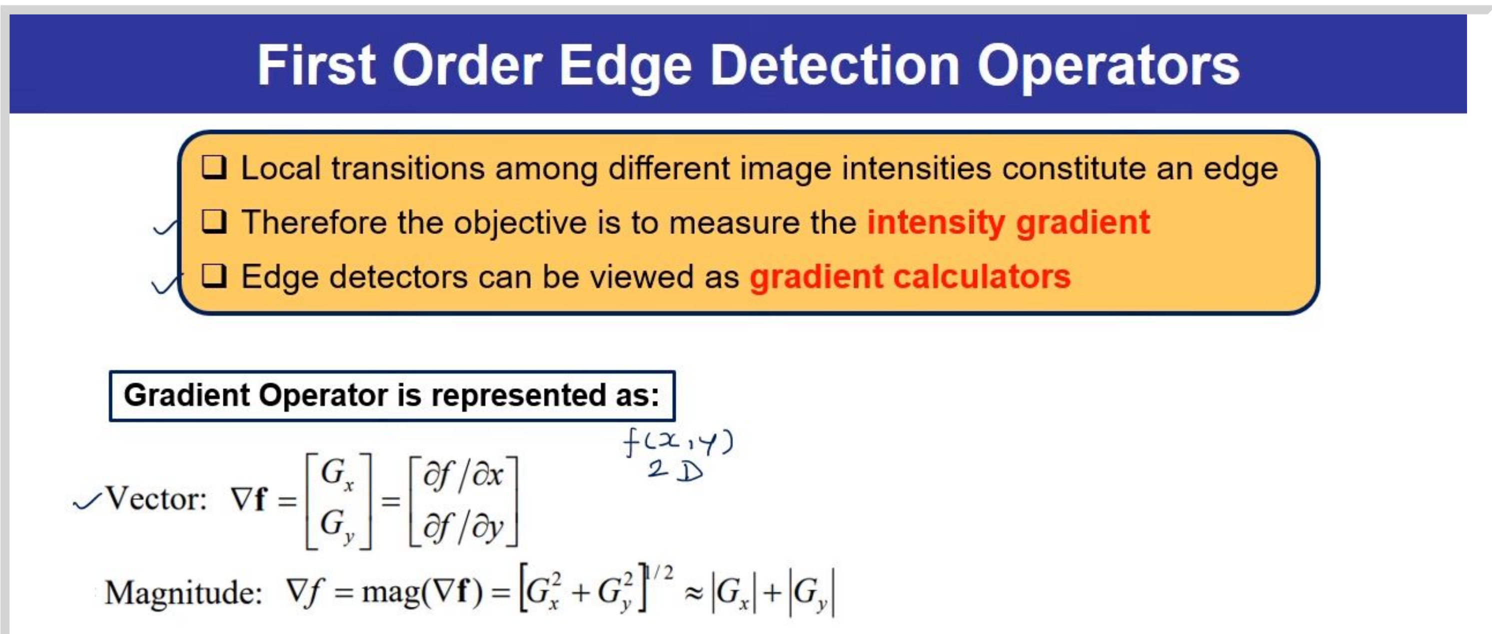
02:30


First Order Edge Detection Operators
❑ Local transitions among different image intensities constitute an edge
❑ Therefore the objective is to measure the intensity gradient
❑ Edge detectors can be viewed as gradient calculators
Gradient Operator is represented as:
Vector: $\nabla \mathbf{f} = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \partial f / \partial x \\ \partial f / \partial y \end{bmatrix}$
Magnitude: $\nabla f = \text{mag}(\nabla \mathbf{f}) = \left[G_x^2 + G_y^2\right]^{1/2} \approx |G_x| + |G_y|$
Direction of gradient: $\tan^{-1}\left(\frac{Gy}{Gx}\right)$

02:49


First Order Edge Detection Operators
❑ Local transitions among different image intensities constitute an edge
❑ Therefore the objective is to measure the intensity gradient
❑ Edge detectors can be viewed as gradient calculators
Gradient Operator is represented as:
f(x,y)
2D
Vector: $\nabla \mathbf{f} = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \partial f / \partial x \\ \partial f / \partial y \end{bmatrix}$
Magnitude: $\nabla f = \text{mag}(\nabla \mathbf{f}) = \left[G_x^2 + G_y^2\right]^{1/2} \approx |G_x| + |G_y|$

Direction of gradient: $\tan^{-1}\left(\dfrac{Gy}{Gx}\right)$

03:53

First Order Edge Detection Operators

- ❑ Local transitions among different image intensities constitute an edge
- ❑ Therefore the objective is to measure the **intensity gradient**
- ❑ Edge detectors can be viewed as **gradient calculators**

Gradient Operator is represented as:

f(x,y)
2D

Vector: $\nabla \mathbf{f} = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \partial f / \partial x \\ \partial f / \partial y \end{bmatrix}$

Magnitude: $\nabla f = \text{mag}(\nabla \mathbf{f}) = \left[G_x^2 + G_y^2\right]^{1/2} \approx |G_x| + |G_y|$

Direction of gradient: $\tan^{-1}\left(\dfrac{Gy}{Gx}\right)$

❖ An edge can be extracted by computing the derivative of the image function

- ✓ **Magnitude of the derivative**, indicates the strength or contrast of edge
- ✓ **Direction of the derivative vector**, indicates the edge orientation

05:08

First Order Edge Detection Operators

Backward Difference:

2 1
x − Δx x

= [f(x) – f(x-Δx)] / Δx

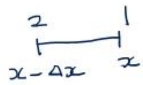
Forward Difference:

= [f(x+ Δx) – f(x)] / Δx

05:25

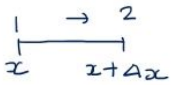
First Order Edge Detection Operators

Backward Difference:



= [f(x) – f(x-Δx)] / Δx

Forward Difference:

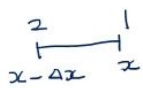


= [f(x+ Δx) – f(x)] / Δx

05:57

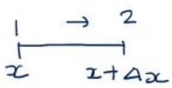
First Order Edge Detection Operators

Backward Difference:



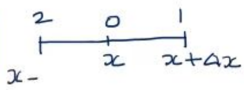
= [f(x) – f(x-Δx)] / Δx

Forward Difference:



= [f(x+ Δx) – f(x)] / Δx

Central Difference:

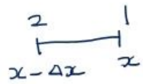


= [f(x+ Δx) – f(x-Δx)] / 2Δx

06:39

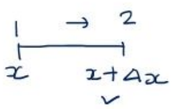
First Order Edge Detection Operators

Backward Difference:



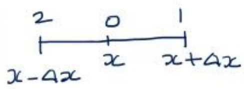
= [f(x) – f(x-Δx)] / Δx

Forward Difference:



= [f(x+ Δx) – f(x)] / Δx

Central Difference:



= [f(x+ Δx) – f(x-Δx)] / 2Δx

These differences can be obtained by applying the following masks, assuming Δx=1:

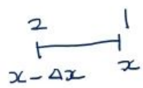
Backward Difference = f(x) – f(x-1)
= [1 -1]

Forward Difference: = f(x+1) – f(x)
= [-1 +1]

06:51

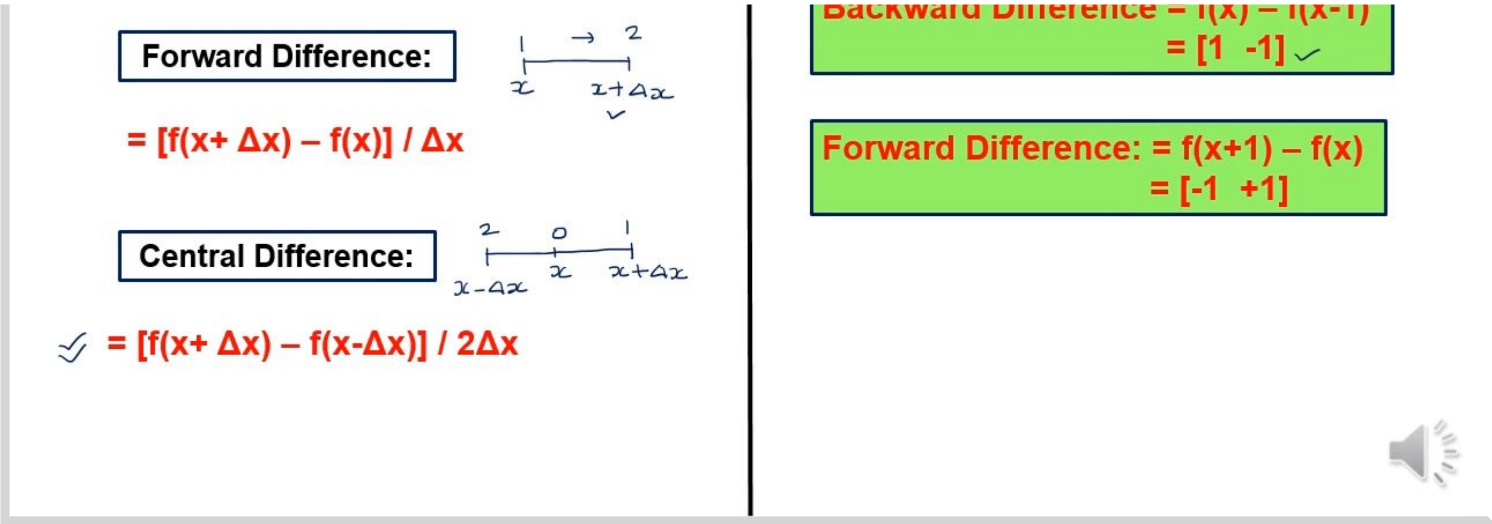
First Order Edge Detection Operators

Backward Difference:

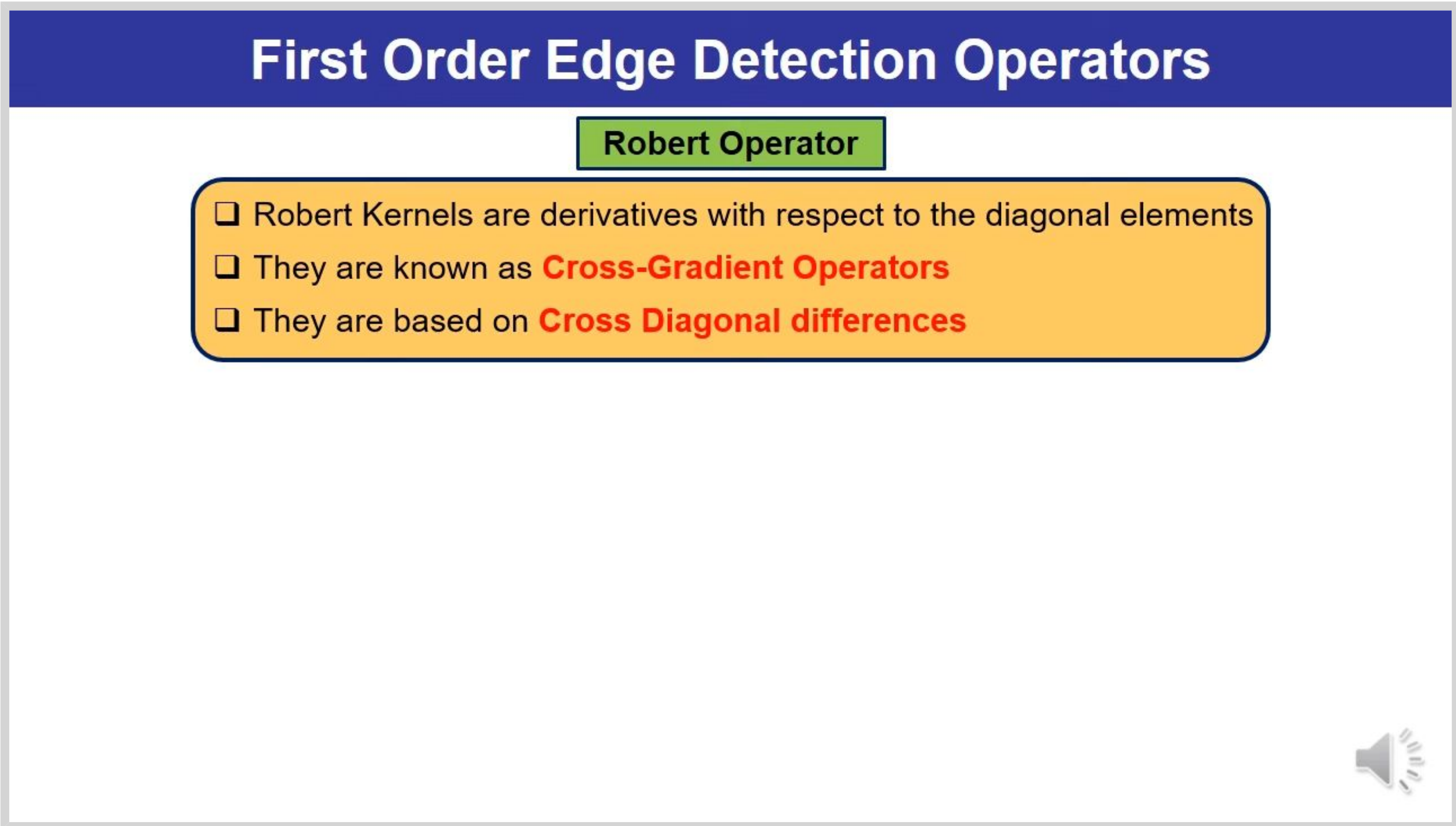


= [f(x) – f(x-Δx)] / Δx

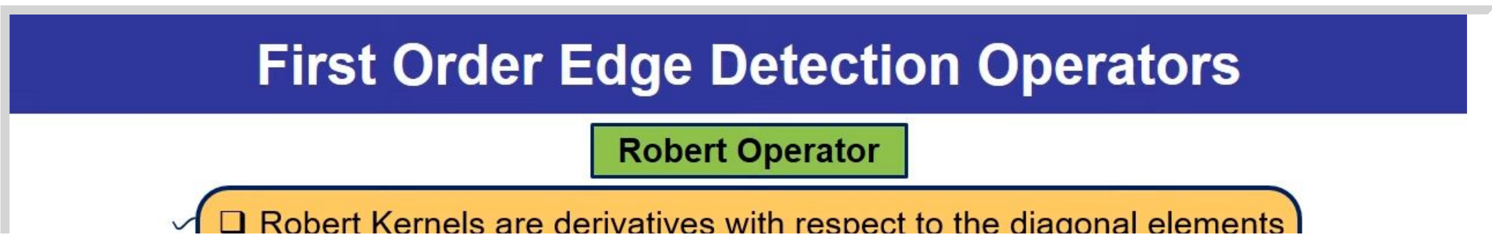
These differences can be obtained by applying the following masks, assuming Δx=1:


Forward Difference:
= [f(x+ Δx) – f(x)] / Δx
Central Difference:
= [f(x+ Δx) – f(x-Δx)] / 2Δx
= [1 -1]
Forward Difference: = f(x+1) – f(x)
= [-1 +1]

07:20


First Order Edge Detection Operators
Robert Operator
Robert Kernels are derivatives with respect to the diagonal elements
They are known as Cross-Gradient Operators
They are based on Cross Diagonal differences

07:49


First Order Edge Detection Operators
Robert Operator
Robert Kernels are derivatives with respect to the diagonal elements

❑ Robert Kernels are derivatives with respect to the diagonal elements
❑ They are known as **Cross-Gradient Operators**
❑ They are based on **Cross Diagonal differences**

Let $f(x,y)$ & $f(x+1,y)$ be neighbouring pixels, then
$$\frac{\partial f}{\partial x} = f(x+1,y) - f(x,y)$$

09:00

First Order Edge Detection Operators

Robert Operator

❑ Robert Kernels are derivatives with respect to the diagonal elements
❑ They are known as **Cross-Gradient Operators**
❑ They are based on **Cross Diagonal differences**

Let $f(x,y)$ & $f(x+1,y)$ be neighbouring pixels, then
$$\frac{\partial f}{\partial x} = f(x+1,y) - f(x,y)$$

Robert masks may be of:

G_x

1	0
0	–1

G_y

0	1
–1	0

09:13

First Order Edge Detection Operators

Robert Operator

- Robert Kernels are derivatives with respect to the diagonal elements
- They are known as **Cross-Gradient Operators**
- They are based on **Cross Diagonal differences**

Generic gradient based algorithm can be:

- Read the image and smooth it
- Convolve the image f with g_x
- Convolve the image f with g_y
- Compute the edge magnitude and edge orientation
- Compare the edge magnitude with a threshold value
  - If edge magnitude is higher, assign it as a possible edge point

Let $f(x,y)$ & $f(x+1,y)$ be neighbouring pixels, then

$$\frac{\partial f}{\partial x} = f(x+1,y) - f(x,y)$$

Robert masks may be of:

G_x

1	0
0	–1

G_y

0	1
–1	0

This algorithm can be applied to other masks also

09:18

First Order Edge Detection Operators

Robert Operator

- Robert Kernels are derivatives with respect to the diagonal elements
- They are known as **Cross-Gradient Operators**
- They are based on **Cross Diagonal differences**

Generic gradient based algorithm can be:

- Read the image and smooth it
- Convolve the image f with g_x
- Convolve the image f with g_y
- Compute the edge magnitude and edge orientation
- Compare the edge magnitude with a threshold value
  - If edge magnitude is higher, assign it as a possible edge point

Let $f(x,y)$ & $f(x+1,y)$ be neighbouring pixels, then

$$\frac{\partial f}{\partial x} = f(x+1,y) - f(x,y)$$

Robert masks may be of:

G_x

1	0
0	–1

G_y

0	1
–1	0

This algorithm can be applied to other masks also

10:36

First Order Edge Detection Operators

Prewitt Operator

The Prewitt Method takes the central difference of the neighbouring pixels; This difference can be represented mathematically as:

11:01

First Order Edge Detection Operators

Prewitt Operator

$$\begin{array}{ccc} 1 & 2 & 3 \\ x-\Delta x & x & x+\Delta x \end{array}$$

The Prewitt Method takes the central difference of the neighbouring pixels; This difference can be represented mathematically as:

$$\frac{\partial f}{\partial x} = [f(x+1) - f(x-1)]/2$$

$\Rightarrow$ For 2D $\rightarrow$ $[f(x+1, y) - f(x-1, y)]/2$

central difference is obtained by mask $\rightarrow$ $[-1 \quad 0 \quad +1]$

11:53

First Order Edge Detection Operators

Prewitt Operator

1 2 3
$x-\Delta x$ x $x+\Delta x$

The Prewitt Method takes the central difference of the neighbouring pixels; This difference can be represented mathematically as:

$$\frac{\partial f}{\partial x} = [f(x+1) - f(x-1)]/2$$

$\Rightarrow$ For 2D $\rightarrow$ (x,y)

$$[f(x+1,y) - f(x-1,y)]/2$$

central difference is obtained by mask $\rightarrow$ $[-1 \quad 0 \quad +1]$

G_x

–1	–1	–1
0	0	0
1	1	1

G_y

–1	0	1
–1	0	1
–1	0	1

12:56

First Order Edge Detection Operators

Prewitt Operator

1 2 3
$x-\Delta x$ x $x+\Delta x$

The Prewitt Method takes the central difference of the neighbouring pixels; This difference can be represented mathematically as:

$$\frac{\partial f}{\partial x} = [f(x+1) - f(x-1)]/2$$

$\Rightarrow$ For 2D $\rightarrow$ (x,y)

$$[f(x+1,y) - f(x-1,y)]/2$$

central difference is obtained by mask $\rightarrow$ $[-1 \quad 0 \quad +1]$

G_x

–1	–1	–1
0	0	0
1	1	1

G_y

–1	0	1
–1	0	1
–1	0	1

3 × 3 digital approximation of Prewitt Operator is given as:

$$G_x = \frac{\partial f}{\partial x} = (z_7 + z_8 + z_9) - (z_1 + z_2 + z_3)$$

$$G_y = \frac{\partial f}{\partial y} = (z_3 + z_6 + z_9) - (z_1 + z_4 + z_7)$$

$$\nabla f \cong |G_x| + |G_y|$$

13:43

First Order Edge Detection Operators

Sobel Operator

13:50

First Order Edge Detection Operators

Sobel Operator

- ❑ It provides both a differentiating and a smoothing effect
- ❑ Sobel Operator relies on the central differences
- ❑ It can be viewed as an approximation of first Gaussian Derivative
- ❑ Here convolution is both commutative an associative

14:40

First Order Edge Detection Operators

Sobel Operator

- It provides both a differentiating and a smoothing effect
- Sobel Operator relies on the central differences
- It can be viewed as an approximation of first Gaussian Derivative
- Here convolution is both commutative an associative

$$\frac{\partial}{\partial x}(f * G) = f * \frac{\partial}{\partial x}(G)$$

14:59

First Order Edge Detection Operators

Sobel Operator

- It provides both a differentiating and a smoothing effect
- Sobel Operator relies on the central differences
- It can be viewed as an approximation of first Gaussian Derivative
- Here convolution is both commutative an associative

$$\frac{\partial}{\partial x}(f * G) = f * \frac{\partial}{\partial x}(G)$$

G_x

–1	–2	–1
0	0	0
1	2	1

G_y

–1	0	1
–2	0	2
–1	0	1

15:49

First Order Edge Detection Operators

Sobel Operator

- It provides both a differentiating and a smoothing effect
- Sobel Operator relies on the central differences
- It can be viewed as an approximation of first Gaussian Derivative
- Here convolution is both commutative an associative

$$\frac{\partial}{\partial x}(f * G) = f * \frac{\partial}{\partial x}(G)$$

G_x

–1	–2	–1
0	0	0
1	2	1

G_y

–1	0	1
–2	0	2
–1	0	1

3 × 3 digital approximation of Sobel Operator is given as:

$$G_x = \frac{\partial f}{\partial x} = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)$$

$$G_y = \frac{\partial f}{\partial y} = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)$$

$$\nabla f \cong |G_x| + |G_y|$$

Additional mask can be used to detect edges in diagonal as:

0	1	2
–1	0	1
–2	–1	0

–2	–1	0
–1	0	1
0	1	2

16:10

First Order Edge Detection Operators

Sobel Operator

- It provides both a differentiating and a smoothing effect
- Sobel Operator relies on the central differences
- It can be viewed as an approximation of first Gaussian Derivative
- Here convolution is both commutative an associative

$$\frac{\partial}{\partial x}(f * G) = f * \frac{\partial}{\partial x}(G)$$

G_x

–1	–2	–1
0	0	0
1	2	1

G_y

–1	0	1
–2	0	2
–1	0	1

3x3

Additional mask can be used to detect edges in diagonal as:

3 × 3 digital approximation of Sobel Operator is given as:

0	1	2

–2	–1	0

$G_x = \frac{\partial f}{\partial x} = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)$

$G_y = \frac{\partial f}{\partial y} = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)$

$\nabla f \cong |G_x| + |G_y|$

–1	0	1
–2	–1	0

–1	0	1
0	1	2

16:22

First Order Edge Detection Operators

Template Matching Masks

16:40

First Order Edge Detection Operators

Template Matching Masks

- ❑ Gradient masks are isotropic and insensitive to direction
- ❑ Sometimes it is necessary to design direction sensitive filters, such type of filters are known as Template Matching Filters

16:48

First Order Edge Detection Operators

Template Matching Masks

- Gradient masks are isotropic and insensitive to direction
- Sometimes it is necessary to design direction sensitive filters, such type of filters are known as Template Matching Filters

Digital Image Processing

Lecture No – 52

First Order Edge Detection Operators
(Image Segmentation)

Dr. Sapna Katiyar
Professor, ECE Department