Computational Photography

Dr. Irfan Essa
Professor
School of Interactive Computing

Study the basics of computation and its impact on the entire workflow of photography, from capturing, manipulating and collaborating on, and sharing photographs.

Image Processing and Filtering, via Convolution and Cross-Correlation

Dr. Irfan Essa
Professor
School of Interactive Computing

Towards Edge Detection: Computing Image Gradients

Lesson Objectives

- ★ Describe in your own words two (2) reasons for the need to detect Features in an image.
- ★ Define what is an Edge of an image from the perspective of Information Theory.
- ★ Describe in your own words the concept of an Image Gradient as needed to compute Edges.
- ★ Explain the concept of an Image Gradient with respect to how it is computed:
    - in continuous form for a function; and
    - in a discrete form for an image.

Review: Convolution and Cross-Correlation Methods

- Can be applied to process images, *e.g.*, smoothing images with kernels, removing noise, *etc.*
- When *h* is symmetric, no difference, use either one?

Review: Convolution and Cross-Correlation Methods

Cross-Correlation: $G = h \otimes F$ $$G[i,j] = \sum_{u=-k}^{k} \sum_{v=-k}^{k} h[u,v] F[i+u, j+v]$$

- ★ Can be applied to process images, *e.g.*, smoothing images with kernels, removing noise, *etc.*
- ★ When *h* is symmetric, no difference, use either one?

4

Review: Convolution and Cross-Correlation Methods

Cross-Correlation: $G = h \otimes F$ $$G[i,j] = \sum_{u=-k}^{k} \sum_{v=-k}^{k} h[u,v]F[i+u, j+v]$$

Convolution: $G = h * F$ $$G[i,j] = \sum_{u=-k}^{k} \sum_{v=-k}^{k} h[u,v]F[i-u, j-v]$$

- ★ Can be applied to process images, *e.g.*, smoothing images with kernels, removing noise, *etc.*
- ★ When *h* is symmetric, no difference, use either one?

Using Filters to Find Features in Image

Slide adapted from Aaron Bobick

Using Filters to Find Features in Image

- Filtering as a way to remove or reduce noise

Slide adapted from Aaron Bobick

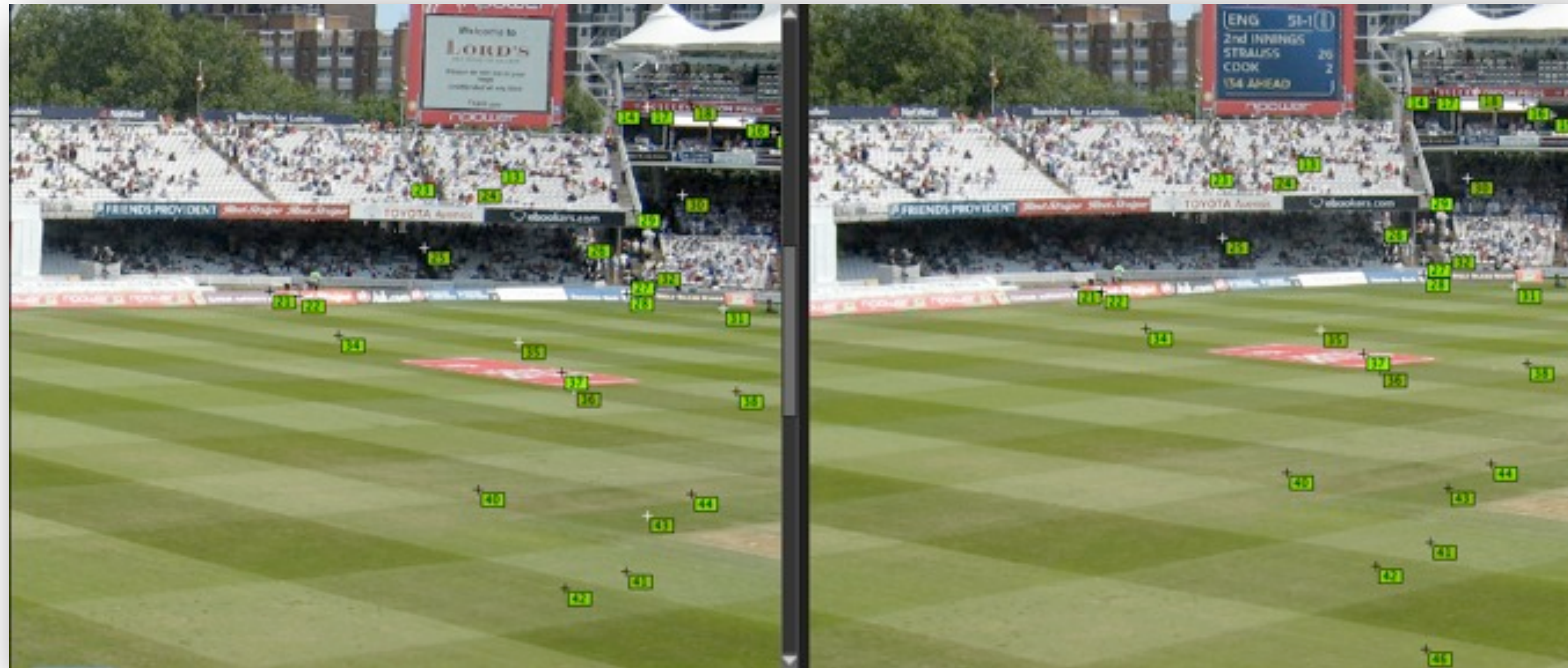
Using Filters to Find Features in Image

- Filtering as a way to remove or reduce noise
- Consider how filters will allow us to abstract higher-level "features".

Slide adapted from Aaron Bobick

Using Filters to Find Features in Image

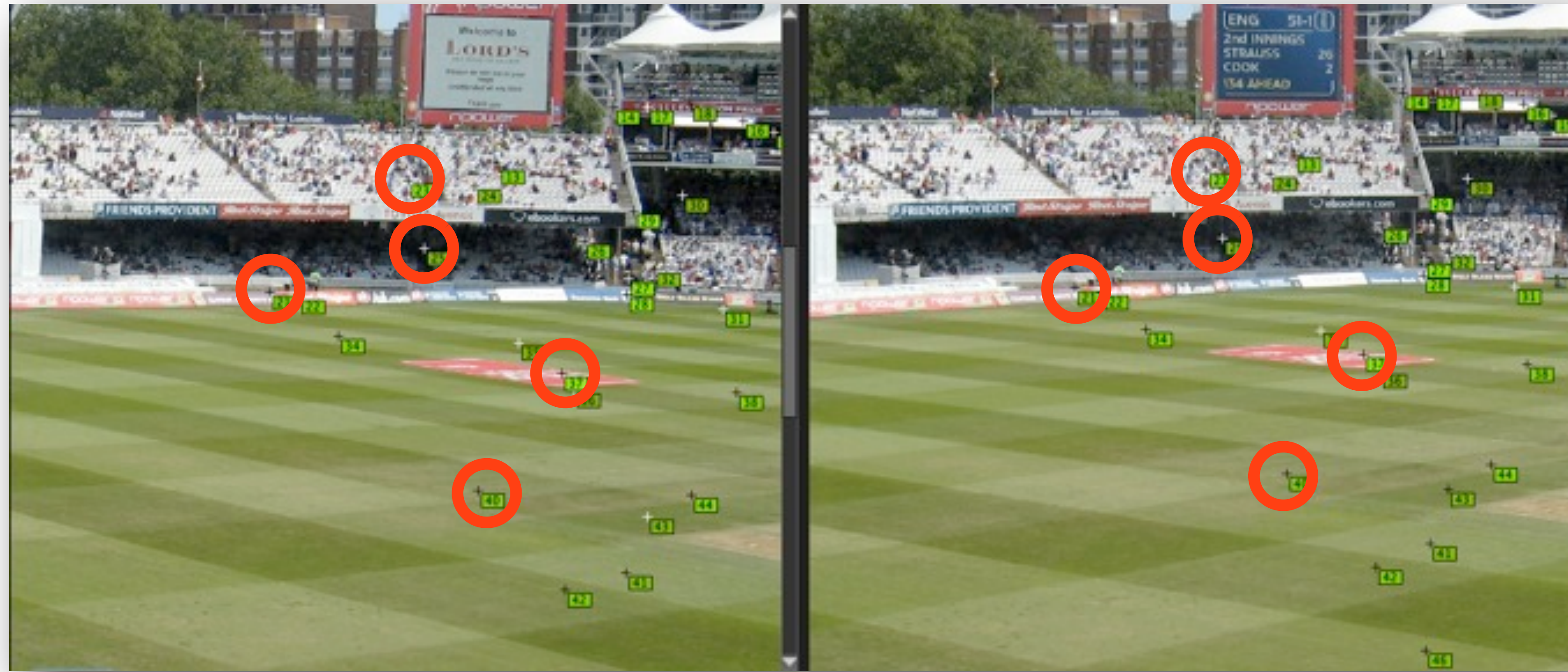
- ★ Filtering as a way to remove or reduce noise
- ★ Consider how filters will allow us to abstract higher-level "features".



Slide adapted from Aaron Bobick

Using Filters to Find Features in Image

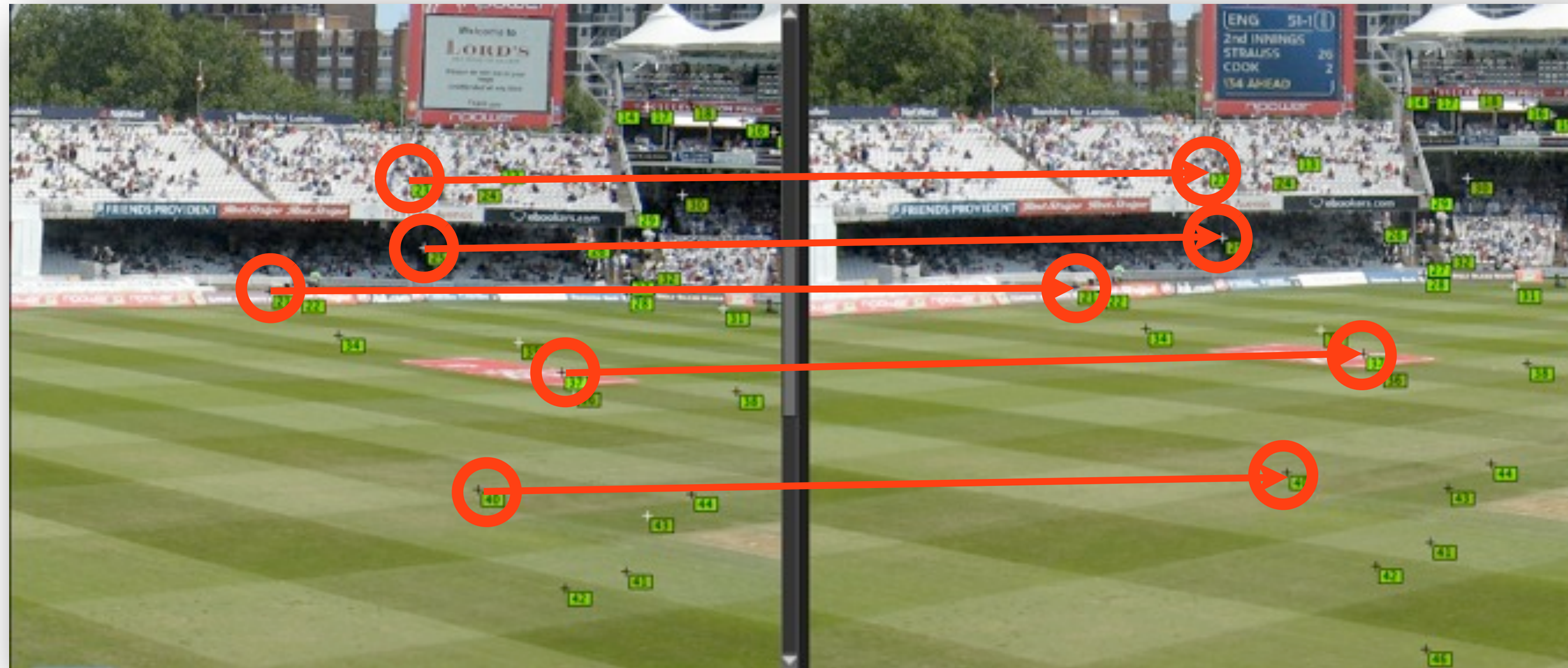
- ★ Filtering as a way to remove or reduce noise
- ★ Consider how filters will allow us to abstract higher-level "features".



Slide adapted from Aaron Bobick

Using Filters to Find Features in Image

- ★ Filtering as a way to remove or reduce noise
- ★ Consider how filters will allow us to abstract higher-level "features".

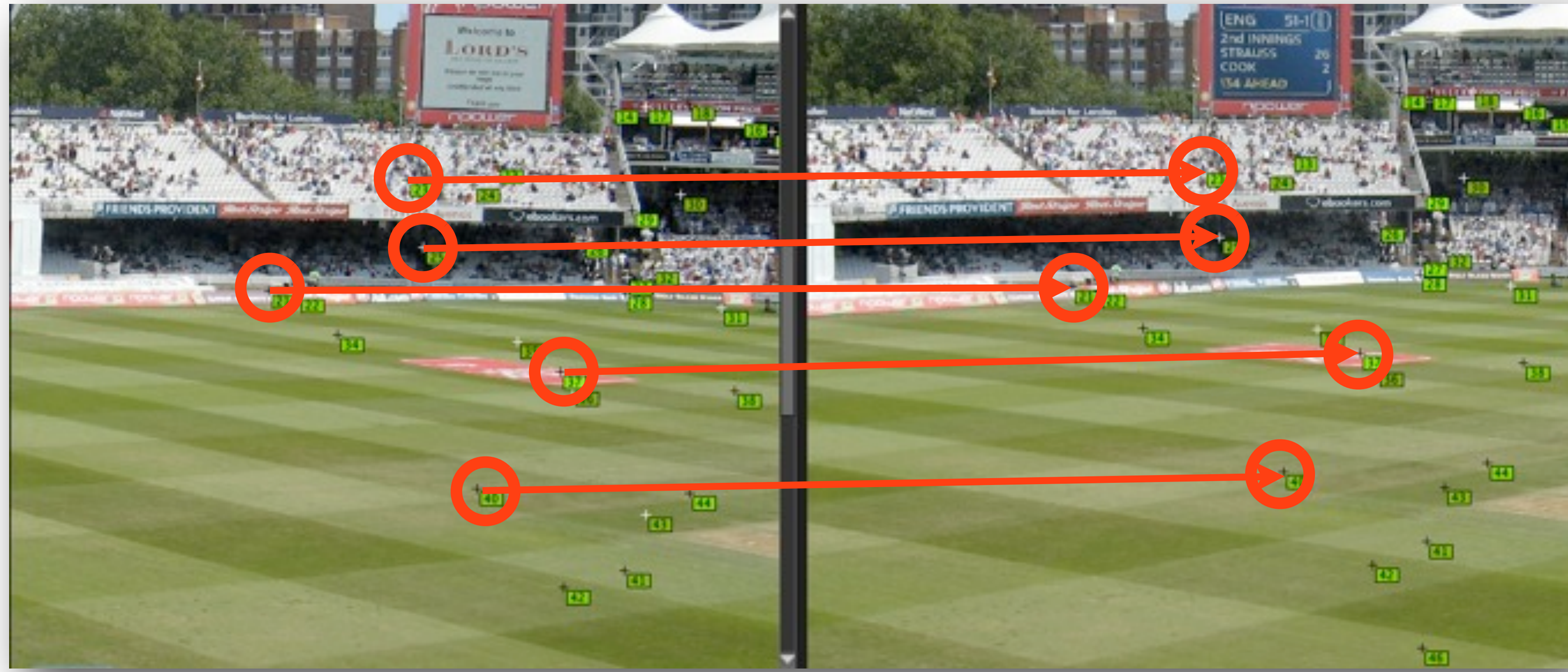


Slide adapted from Aaron Bobick

5

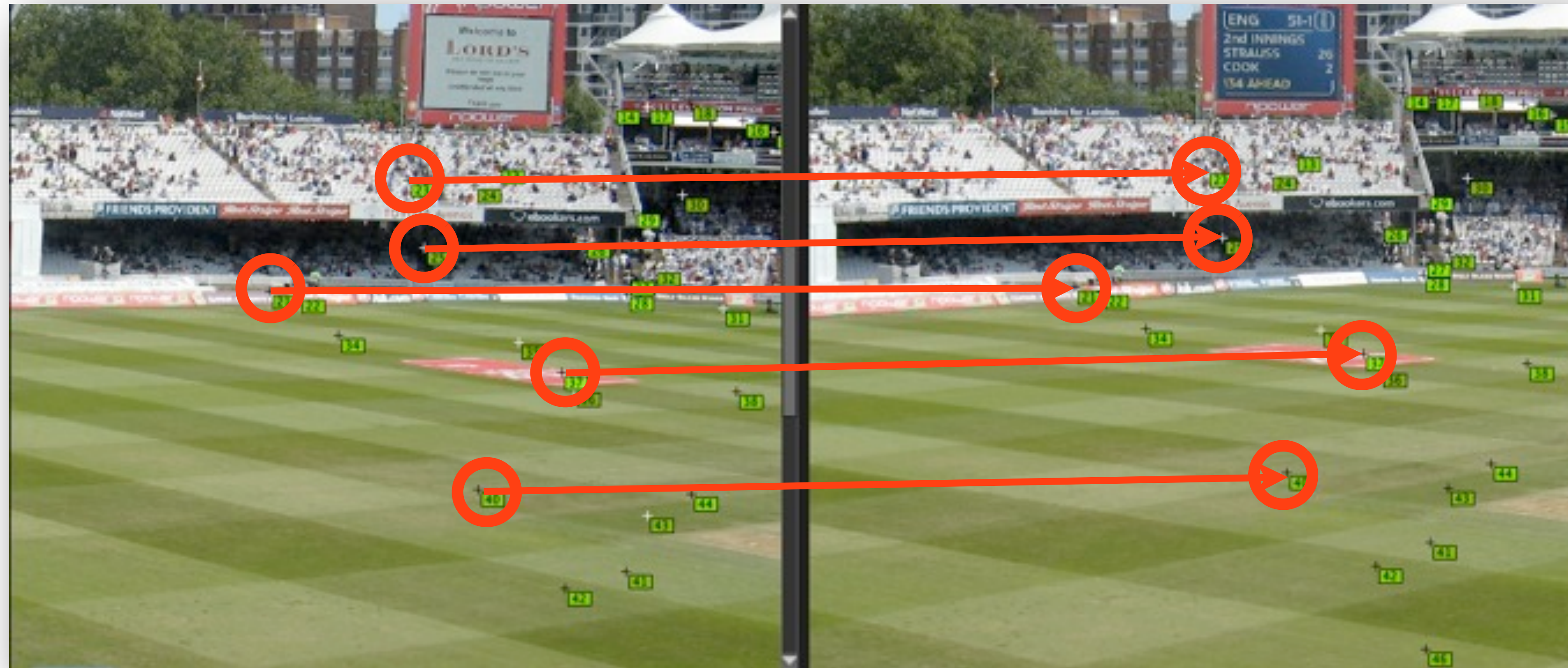
Using Filters to Find Features in Image

- ★ Filtering as a way to remove or reduce noise
- ★ Consider how filters will allow us to abstract higher-level "features".
    - Map raw pixels to an intermediate representation that will be used for subsequent processing



Slide adapted from Aaron Bobick

Using Filters to Find Features in Image



- ★ Filtering as a way to remove or reduce noise
- ★ Consider how filters will allow us to abstract higher-level "features".
  - Map raw pixels to an intermediate representation that will be used for subsequent processing
  - Reduce amount of data, discard redundancy, preserve what's useful

Slide adapted from Aaron Bobick

What are Good Features to Match between Images

Slide adapted from Aaron Bobick

What are Good Features to Match between Images

- ★ **Good Features**

Slide adapted from Aaron Bobick

What are Good Features to Match between Images

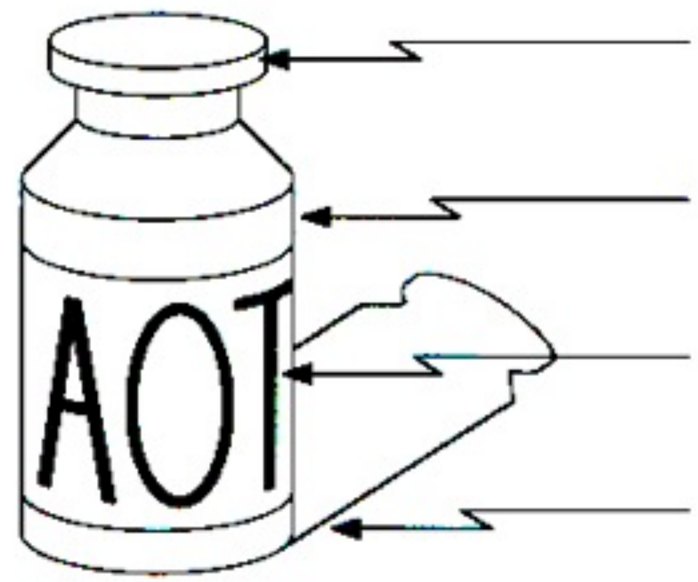
★ Good Features

- Parts or properties of the image that encode it in a compact form

Slide adapted from Aaron Bobick

What are Good Features to Match between Images

- ★ **Good Features**
  - Parts or properties of the image that encode it in a compact form
- ★ **Edges**



Slide adapted from Aaron Bobick

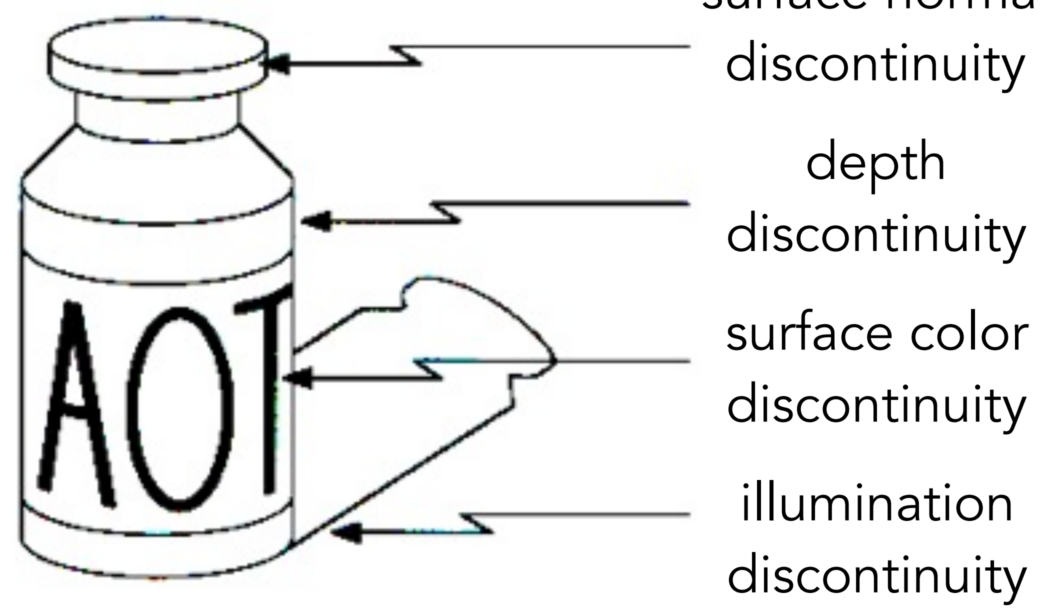
What are Good Features to Match between Images

★ **Good Features**

- Parts or properties of the image that encode it in a compact form

★ **Edges**

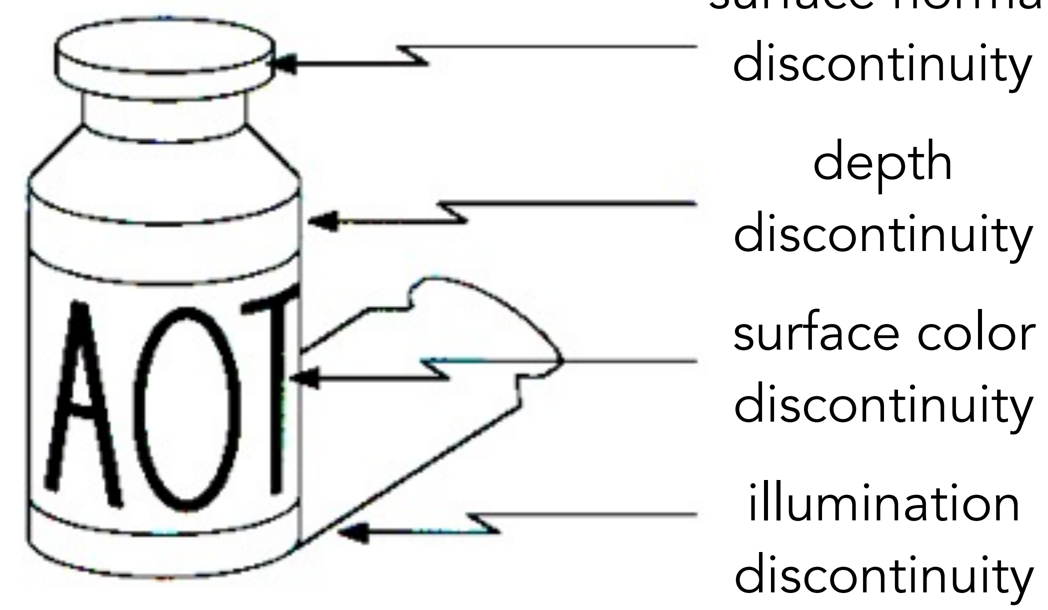
- Edges are caused by a variety of factors



Slide adapted from Aaron Bobick

What are Good Features to Match between Images

- **Good Features**
  - Parts or properties of the image that encode it in a compact form
- **Edges**
  - Edges are caused by a variety of factors
  - Information theory view: Edges encode change, therefore edges efficiently encode an image



Slide adapted from Aaron Bobick

Image Courtesy Professor Henrik Christensen

Cause of Edges in Real Images

Cause of Edges in Real Images

Image Courtesy Professor Henrik Christensen

Images as Functions

- Edges appear as cliffs in the 3d height-filed view of an image
- Image: *F(x,y)*

Images as Functions

- Edges appear as cliffs in the 3d height-filed view of an image
- Image: *F(x,y)*

Edge Detection: Look for Changes

12	90	89	86	87	82
10	12	88	85	83	84
9	15	12	84	84	88
12	14	10	82	88	89
11	17	16	12	88	90
10	16	15	17	89	88

Edge Detection: Look for Changes

★ Basic Idea:

- Look for a neighborhood with strong signs of change

12	90	89	86	87	82
10	12	88	85	83	84
9	15	12	84	84	88
12	14	10	82	88	89
11	17	16	12	88	90
10	16	15	17	89	88

Edge Detection: Look for Changes

★ Basic Idea:

- Look for a neighborhood with strong signs of change

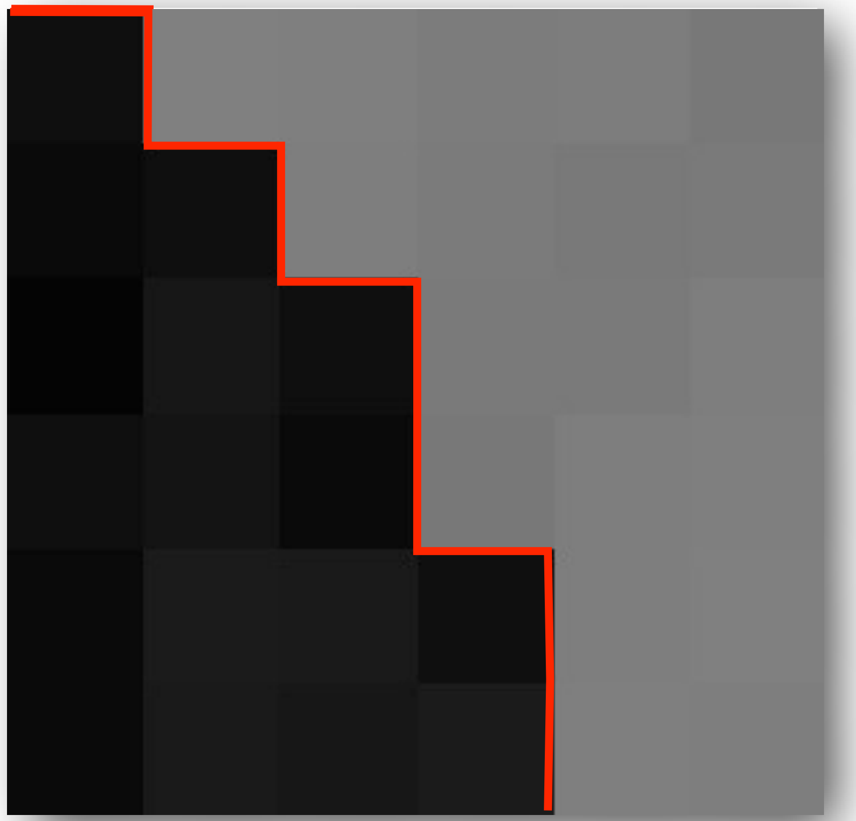
12	90	89	86	87	82
10	12	88	85	83	84
9	15	12	84	84	88
12	14	10	82	88	89
11	17	16	12	88	90
10	16	15	17	89	88

Edge Detection: Look for Changes

★ Basic Idea:

- Look for a neighborhood with strong signs of change

Edge Detection: Look for Changes



★ Basic Idea:

- Look for a neighborhood with strong signs of change

Edge Detection: Look for Changes

★ Basic Idea:

- Look for a neighborhood with strong signs of change

12	90	89	86	87	82
10	12	88	85	83	84
9	15	12	84	84	88
12	14	10	82	88	89
11	17	16	12	88	90
10	16	15	17	89	88

Edge Detection: Look for Changes

12	90	89	86	87	82
10	12	88	85	83	84
9	15	12	84	84	88
12	14	10	82	88	89
11	17	16	12	88	90
10	16	15	17	89	88

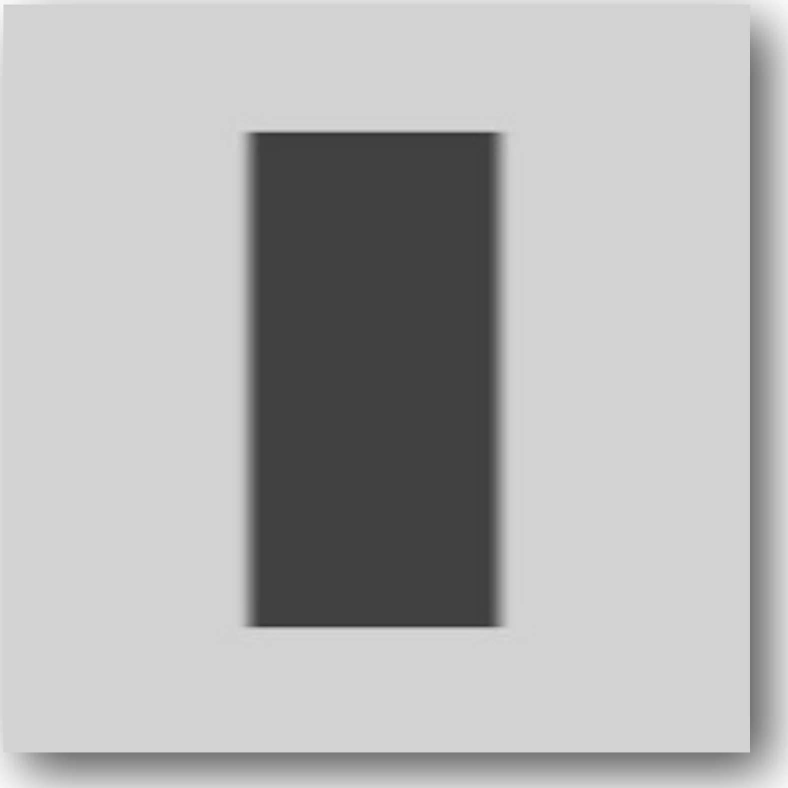
- ★ Basic Idea:
  - Look for a neighborhood with strong signs of change
- ★ Issues to consider
  - Size of the neighborhood?
  - What metrics to consider to detect a “change”?

Derivatives of *F(x,y)* to get Edges

Test Image

Slide adapted from Aaron Bobick and Svetlana Lazebnik

Derivatives of *F(x,y)* to get Edges

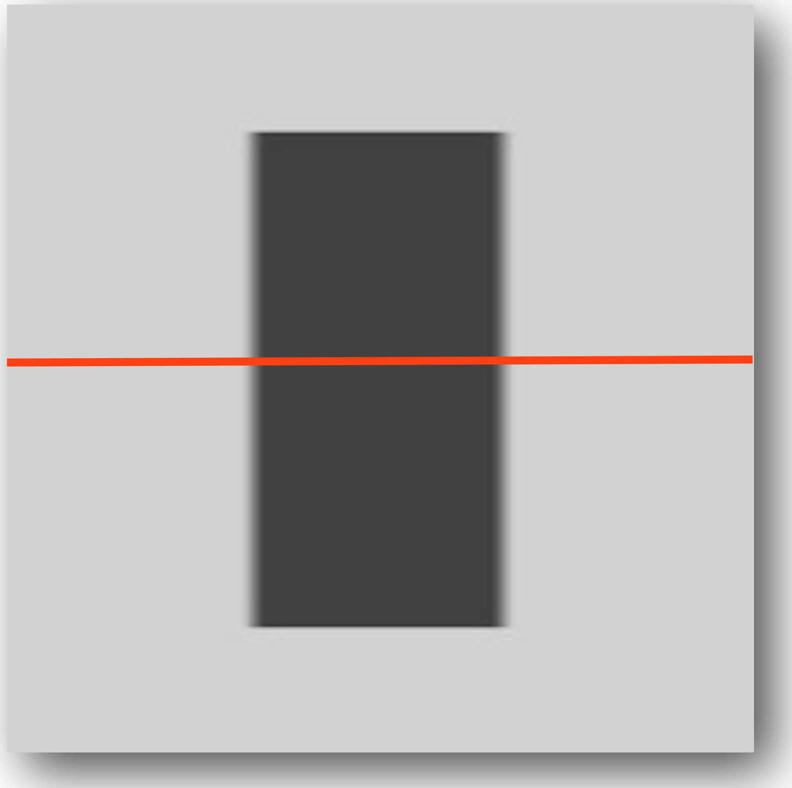


Test Image

- An edge is where there is rapid change in the image intensity function

Slide adapted from Aaron Bobick and Svetlana Lazebnik

Derivatives of *F(x,y)* to get Edges

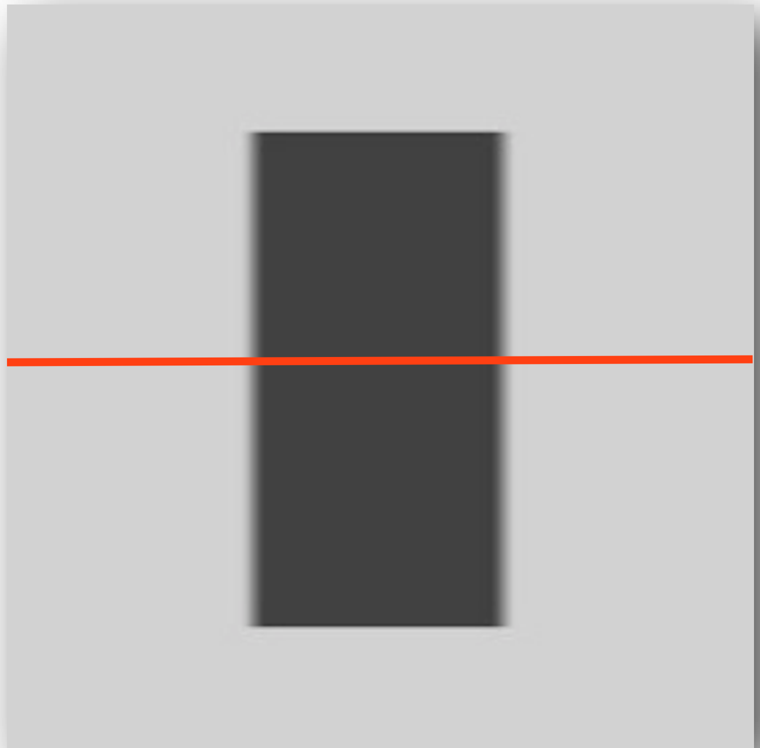


Test Image

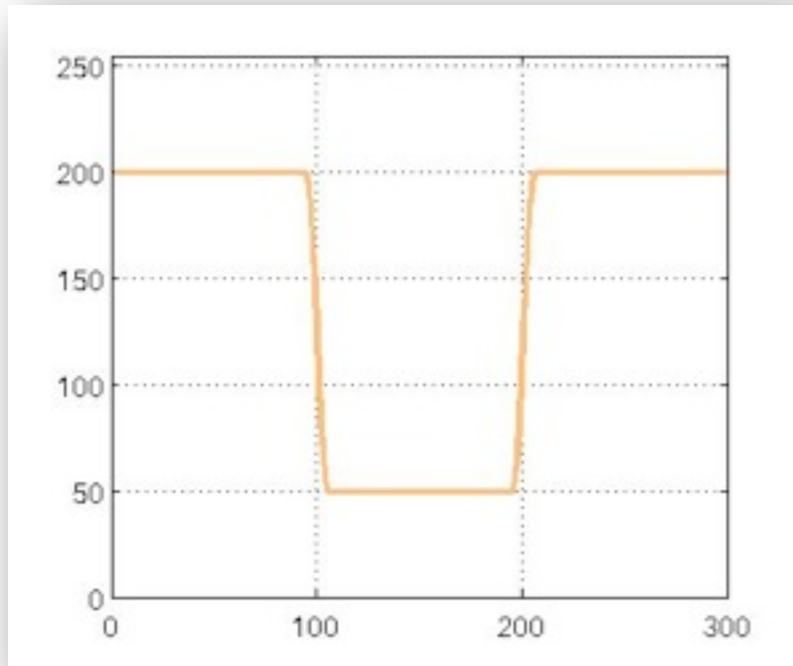
★ An edge is where there is rapid change in the image intensity function

Slide adapted from Aaron Bobick and Svetlana Lazebnik

Derivatives of *F(x,y)* to get Edges



Test Image

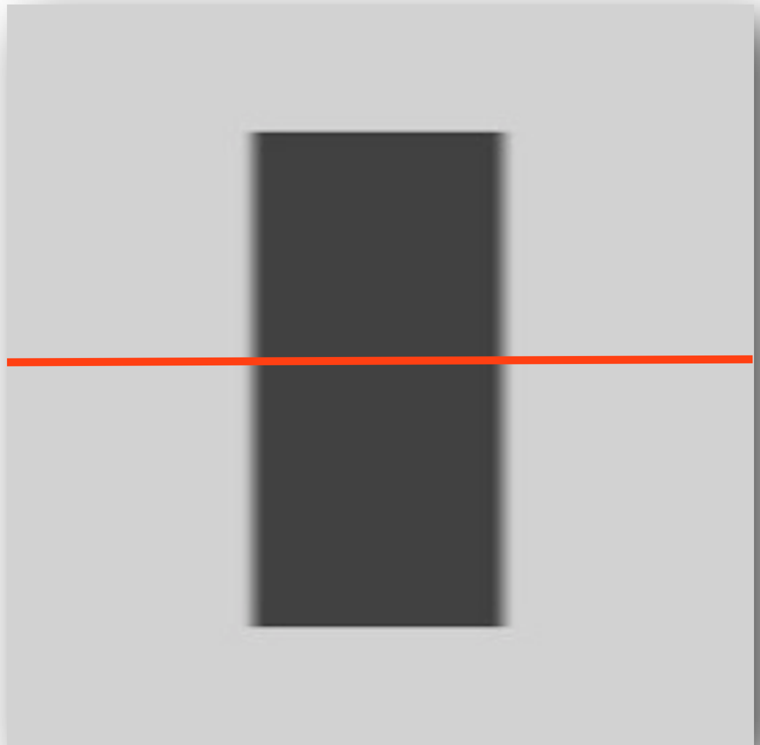


Intensity along
horizontal scan line (in red)

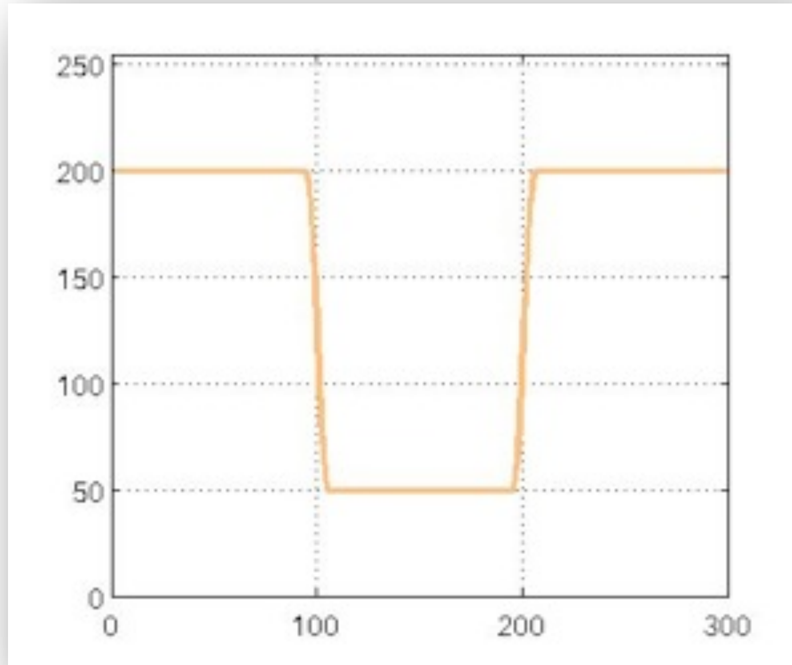
★ An edge is where there is rapid change in the image intensity function

Slide adapted from Aaron Bobick and Svetlana Lazebnik

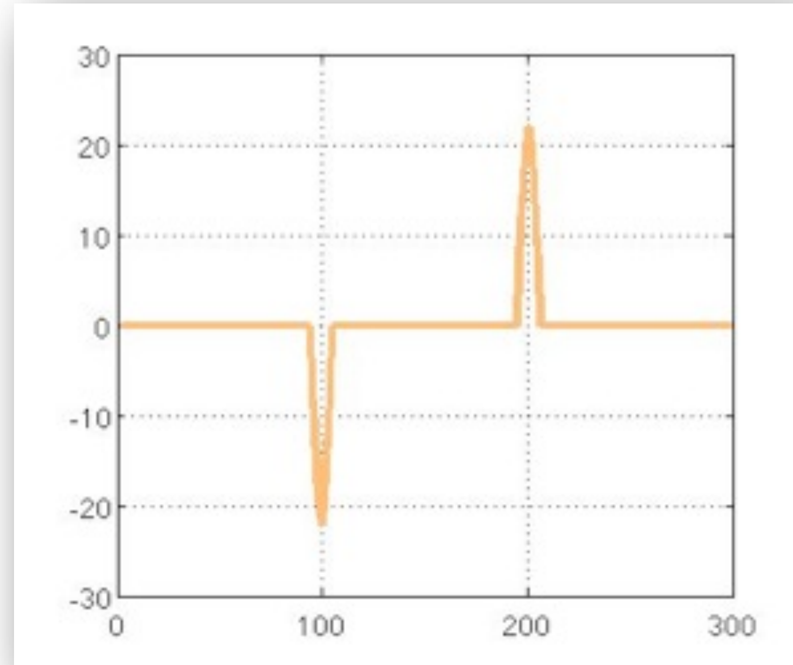
Derivatives of *F(x,y)* to get Edges



Test Image



Intensity along horizontal scan line (in red)

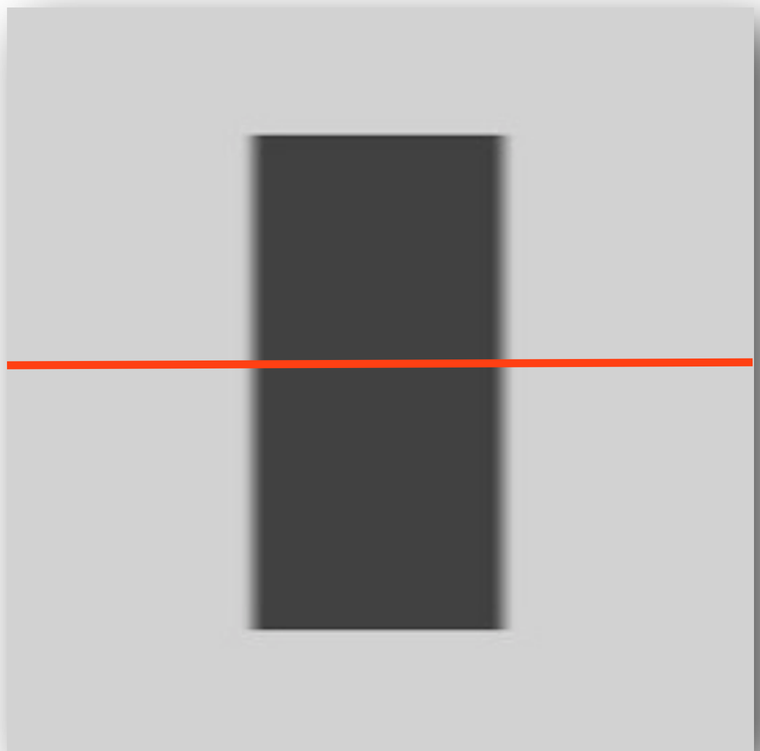


First Derivative

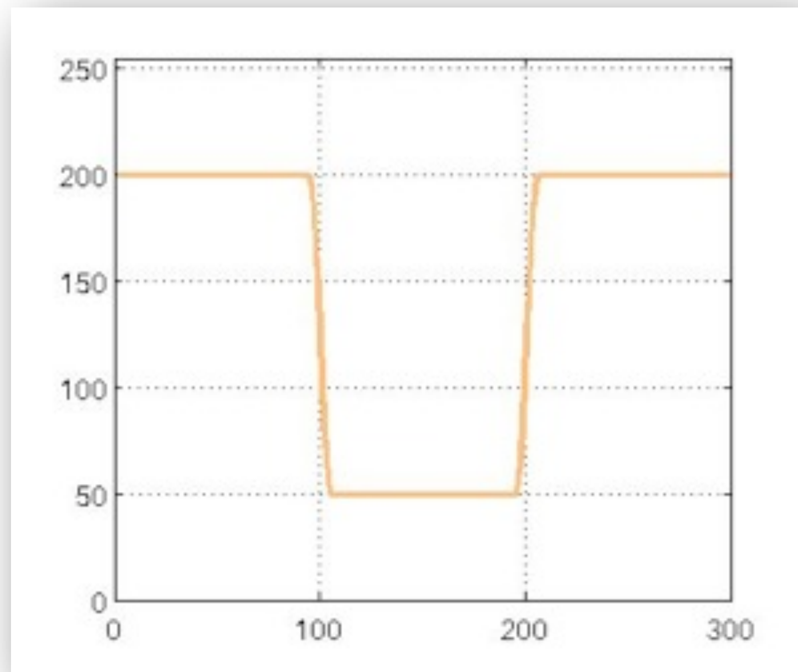
- ★ An edge is where there is rapid change in the image intensity function

Slide adapted from Aaron Bobick and Svetlana Lazebnik

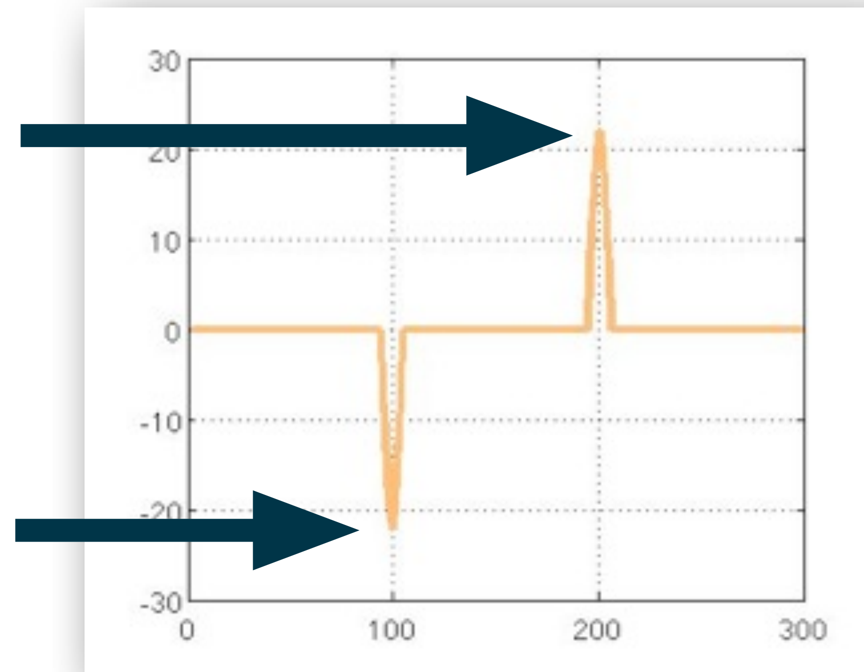
Derivatives of *F(x,y)* to get Edges



Test Image



Intensity along horizontal scan line (in red)



First Derivative

- ★ An edge is where there is rapid change in the image intensity function
- ★ Extrema of the derivative is where the vertical edges are (as we just looked for derivative in *x*)

Slide adapted from Aaron Bobick and Svetlana Lazebnik

Differential Operators for Images

Differential Operators for Images

★ Need an operation that when applied to an image returns its derivatives: Differential Operators

11

Differential Operators for Images

- ★ Need an operation that when applied to an image returns its derivatives: Differential Operators
- ★ Model these "operators" as masks/kernels: When applied to an image yield a new function that is the image *gradient function*.

Differential Operators for Images

- Need an operation that when applied to an image returns its derivatives: Differential Operators
- Model these "operators" as masks/kernels: When applied to an image yield a new function that is the image *gradient function*.
- Then "threshold" this *gradient function* to select the edge pixels.

Differential Operators for Images

- ★ Need an operation that when applied to an image returns its derivatives: Differential Operators
- ★ Model these "operators" as masks/kernels: When applied to an image yield a new function that is the image *gradient function*.
- ★ Then "threshold" this *gradient function* to select the edge pixels.
- ★ Need to define a *gradient*!

Gradient of an image = Measure of change in an Image (function, *F*) in *x* (rows) and *y* (columns)

Definition: Image Gradient (Mathematically)

Gradient of an Image is: $\nabla F = \left[\frac{\delta F}{\delta x}, \frac{\delta F}{\delta y} \right]$

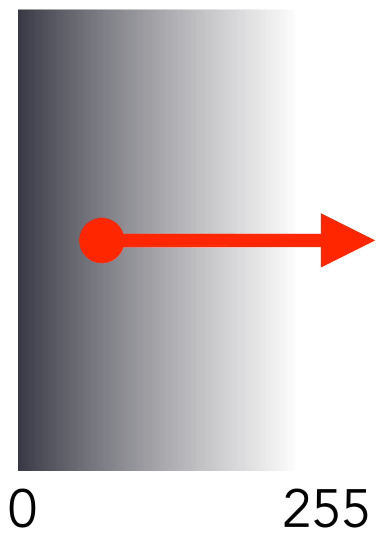
Definition: Image Gradient (Mathematically)

Gradient of an Image is: $\nabla F = \left[\frac{\delta F}{\delta x}, \frac{\delta F}{\delta y} \right]$

Gradient of an image = Measure of change in an Image (function, *F*) in *x* (rows) and *y* (columns)

Definition: Image Gradient (Mathematically)

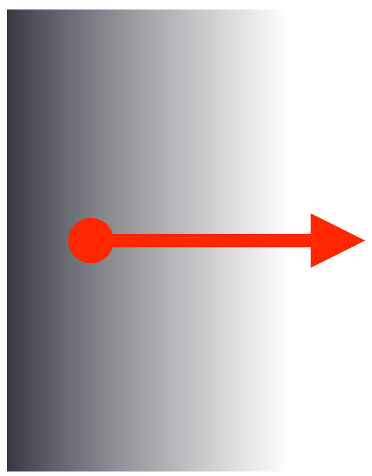
Gradient of an Image is: $\nabla F = \left[\frac{\delta F}{\delta x}, \frac{\delta F}{\delta y} \right]$



Gradient of an image = Measure of change in an Image (function, *F*) in *x* (rows) and *y* (columns)

Definition: Image Gradient (Mathematically)

Gradient of an Image is: $\nabla F = \left[\frac{\delta F}{\delta x}, \frac{\delta F}{\delta y}\right]$



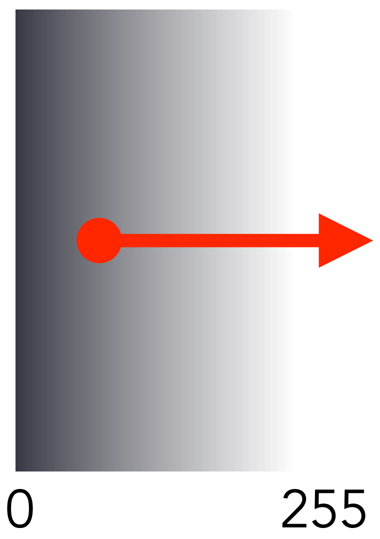
0 255

$$\nabla F = \left[\frac{\delta F}{\delta x}, 0\right]$$

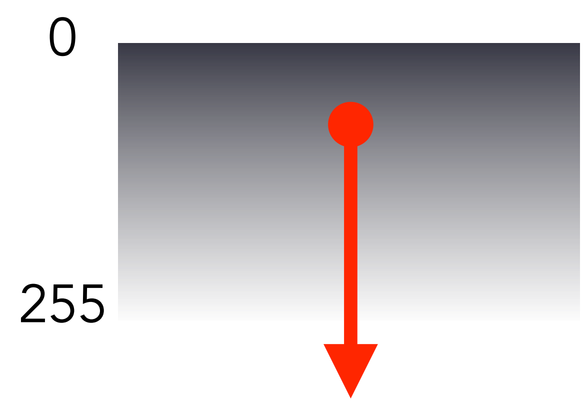
Gradient of an image = Measure of change in an Image (function, *F*) in *x* (rows) and *y* (columns)

Definition: Image Gradient (Mathematically)

Gradient of an Image is: $\nabla F = \left[\frac{\delta F}{\delta x}, \frac{\delta F}{\delta y} \right]$



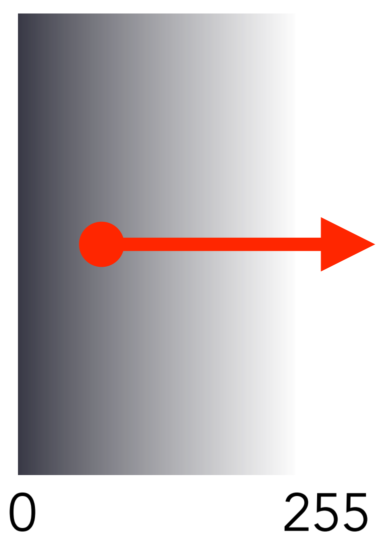
$$\nabla F = \left[\frac{\delta F}{\delta x}, 0 \right]$$



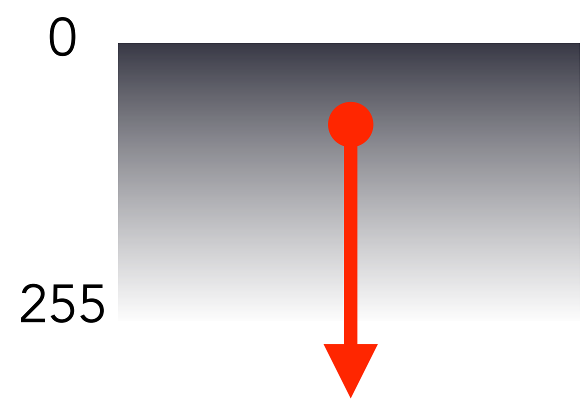
Gradient of an image = Measure of change in an Image (function, *F*) in *x* (rows) and *y* (columns)

Definition: Image Gradient (Mathematically)

Gradient of an Image is: $\nabla F = \left[\frac{\delta F}{\delta x}, \frac{\delta F}{\delta y} \right]$



$$\nabla F = \left[\frac{\delta F}{\delta x}, 0 \right]$$



$$\nabla F = \left[0, \frac{\delta F}{\delta y} \right]$$

Gradient of an image = Measure of change in an Image (function, *F*) in *x* (rows) and *y* (columns)

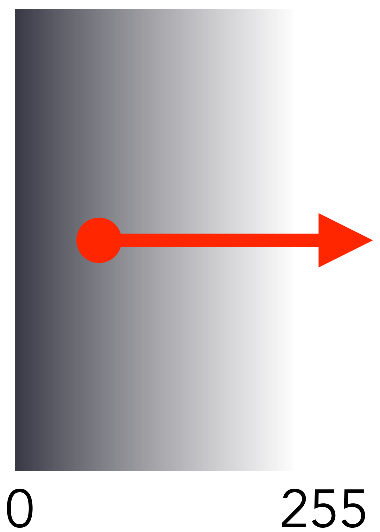
12

Georgia Institute of Technology

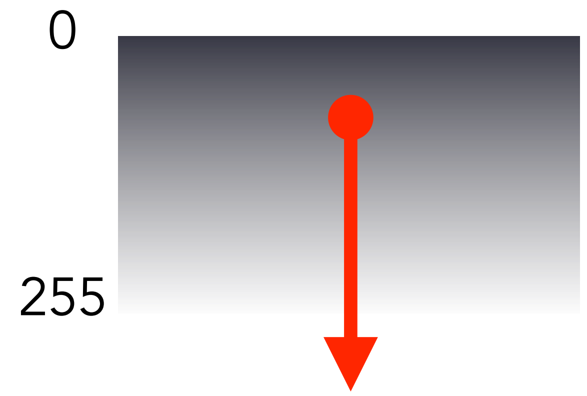
Definition: Image Gradient (Mathematically)

Gradient of an Image is: $\nabla F = \left[\frac{\delta F}{\delta x}, \frac{\delta F}{\delta y} \right]$

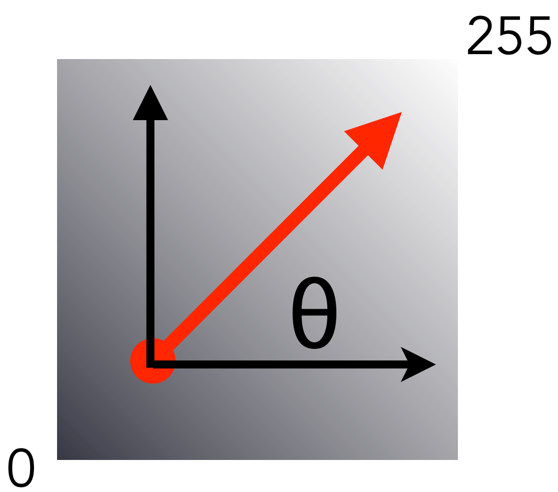
Gradient of an image = Measure of change in an Image (function, *F*) in *x* (rows) and *y* (columns)



$$\nabla F = \left[\frac{\delta F}{\delta x}, 0 \right]$$

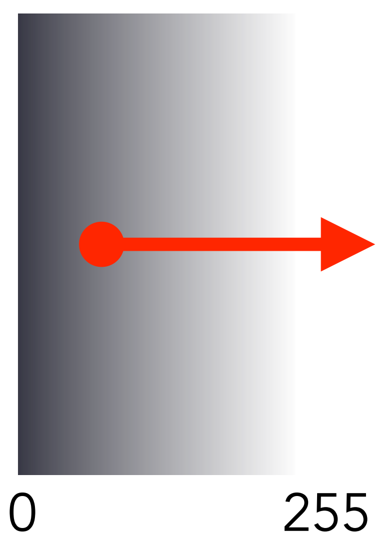


$$\nabla F = \left[0, \frac{\delta F}{\delta y} \right]$$

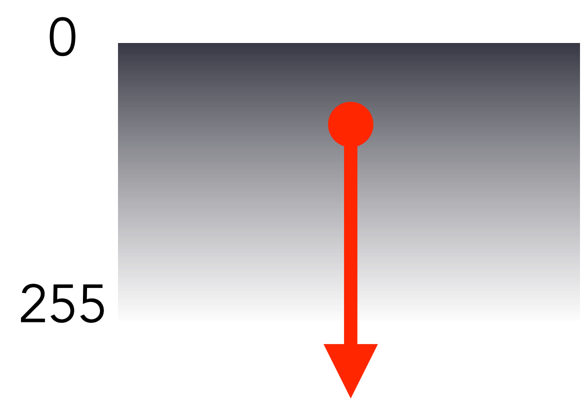


Definition: Image Gradient (Mathematically)

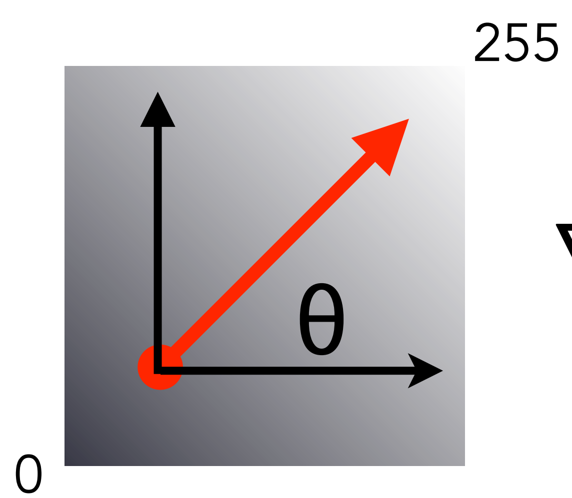
Gradient of an Image is: $\nabla F = \left[\frac{\delta F}{\delta x}, \frac{\delta F}{\delta y}\right]$



$$\nabla F = \left[\frac{\delta F}{\delta x}, 0\right]$$



$$\nabla F = \left[0, \frac{\delta F}{\delta y}\right]$$

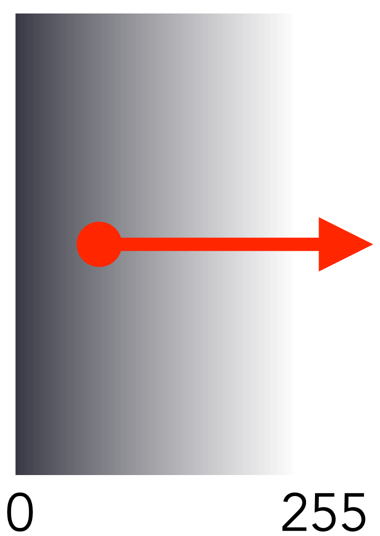


$$\nabla F = \left[\frac{\delta F}{\delta x}, \frac{\delta F}{\delta y}\right]$$

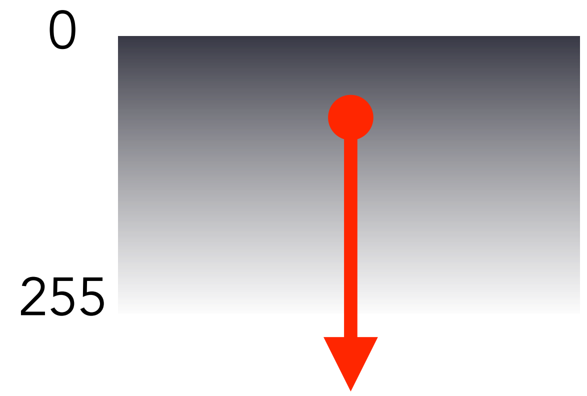
Gradient of an image = Measure of change in an Image (function, *F*) in *x* (rows) and *y* (columns)

Definition: Image Gradient (Mathematically)

Gradient of an Image is: $\nabla F = \left[\frac{\delta F}{\delta x}, \frac{\delta F}{\delta y} \right]$

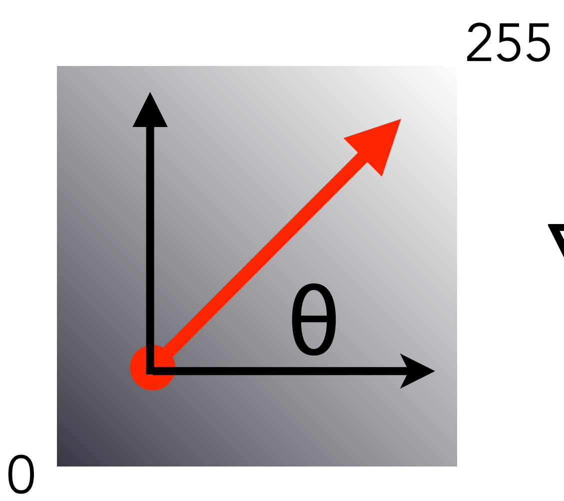


$$\nabla F = \left[\frac{\delta F}{\delta x}, 0 \right]$$



$$\nabla F = \left[0, \frac{\delta F}{\delta y} \right]$$

Gradient of an image = Measure of change in an Image (function, *F*) in *x* (rows) and *y* (columns)

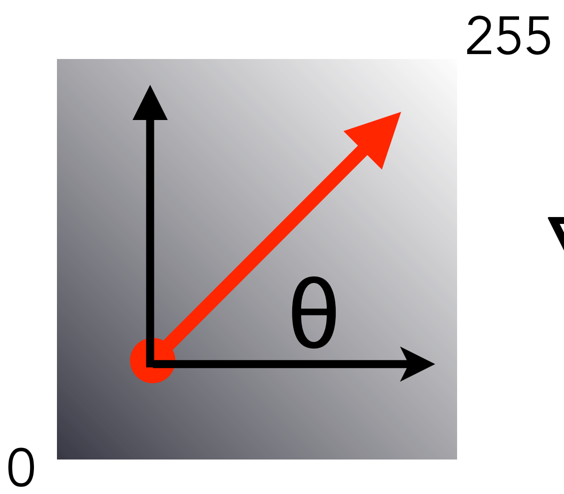


$$\nabla F = \left[\frac{\delta F}{\delta x}, \frac{\delta F}{\delta y} \right]$$

The gradient points in the direction of most rapid increase in intensity

Definition: Image Gradient (Mathematically)

Gradient of an image = Measure of change in an Image (function, *F*) in *x* (rows) and *y* (columns)

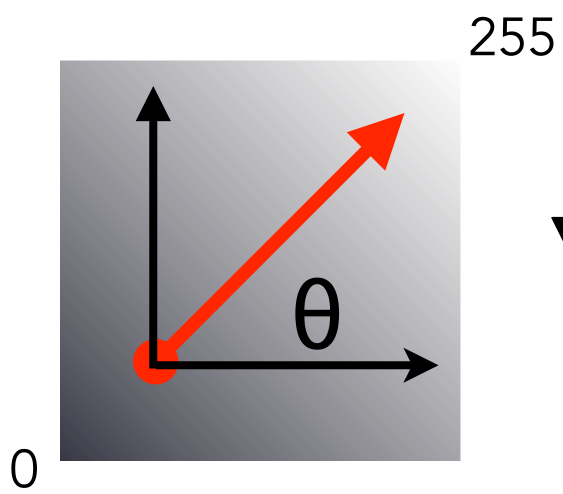


$$\nabla F = \left[\frac{\delta F}{\delta x}, \frac{\delta F}{\delta y} \right]$$

The gradient points in the direction of most rapid increase in intensity

Slide adapted from Aaron Bobick

Definition: Image Gradient (Mathematically)



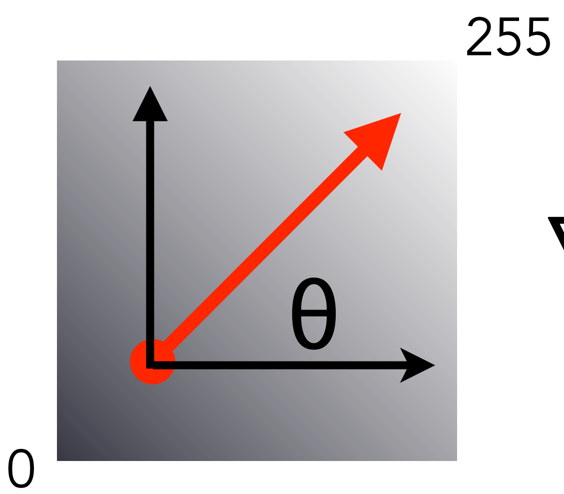
$$\nabla F = \left[\frac{\delta F}{\delta x}, \frac{\delta F}{\delta y}\right]$$

The gradient points in the direction of most rapid increase in intensity

Gradient of an image = Measure of change in an Image (function, *F*) in *x* (rows) and *y* (columns)

Slide adapted from Aaron Bobick

Definition: Image Gradient (Mathematically)



$$\nabla F = \left[\frac{\delta F}{\delta x}, \frac{\delta F}{\delta y} \right]$$

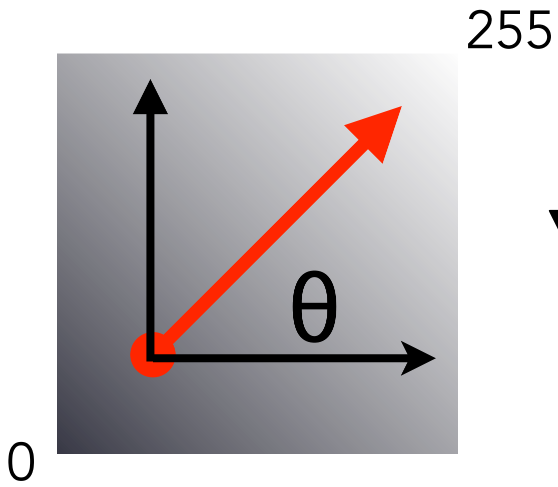
The gradient points in the direction of most rapid increase in intensity

Gradient of an image = Measure of change in an Image (function, *F*) in *x* (rows) and *y* (columns)

Gradient Direction is: $\theta = \tan^{-1}\left[\frac{\delta F}{\delta y} / \frac{\delta F}{\delta x} \right]$

Slide adapted from Aaron Bobick

Definition: Image Gradient (Mathematically)



$$\nabla F = \left[\frac{\delta F}{\delta x}, \frac{\delta F}{\delta y} \right]$$

The gradient points in the direction of most rapid increase in intensity

Gradient of an image = Measure of change in an Image (function, *F*) in *x* (rows) and *y* (columns)

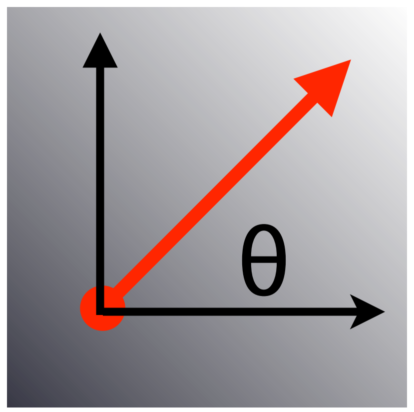
Gradient Direction is: $\theta = \tan^{-1} \left[\frac{\delta F}{\delta y} / \frac{\delta F}{\delta x} \right]$

How does this direction relate to edge direction?

Slide adapted from Aaron Bobick

Definition: Image Gradient (Mathematically)

255



0

$$\nabla F = \left[\frac{\delta F}{\delta x}, \frac{\delta F}{\delta y} \right]$$

The gradient points in the direction of most rapid increase in intensity

Gradient of an image = Measure of change in an Image (function, *F*) in *x* (rows) and *y* (columns)

Gradient Direction is: $\theta = \tan^{-1} \left[\frac{\delta F}{\delta y} / \frac{\delta F}{\delta x} \right]$

How does this direction relate to edge direction?

Gradient Magnitude provides edge strength:

$$\| \nabla F \| = \sqrt{ \left(\frac{\delta F}{\delta x} \right)^2 + \left(\frac{\delta F}{\delta y} \right)^2 }$$

Slide adapted from Aaron Bobick

Definition: Image Gradient (Discrete)

For 2D function, *F(x,y)*, the partial derivative is:

$$\frac{\delta F(x,y)}{\delta x} = \lim_{\epsilon \to 0} \frac{F(x+\epsilon, y) - F(x,y)}{\epsilon}$$

For discrete data, we can approximate using finite differences:

Slide adapted from Aaron Bobick

Definition: Image Gradient (Discrete)

For 2D function, *F(x,y)*, the partial derivative is:

$$\frac{\delta F(x,y)}{\delta x} = \lim_{\epsilon \to 0} \frac{F(x+\epsilon, y) - F(x,y)}{\epsilon}$$

For discrete data, we can approximate using finite differences:

$$\frac{\delta F(x,y)}{\delta x} \approx \frac{F(x+1, y) - F(x,y)}{1}$$

Slide adapted from Aaron Bobick

Definition: Image Gradient (Discrete)

For 2D function, *F(x,y)*, the partial derivative is:

$$\frac{\delta F(x,y)}{\delta x} = \lim_{\epsilon \to 0} \frac{F(x+\epsilon,y) - F(x,y)}{\epsilon}$$

For discrete data, we can approximate using finite differences:

X : $$\frac{\delta F(x,y)}{\delta x} \approx \frac{F(x+1,y) - F(x,y)}{1}$$

Slide adapted from Aaron Bobick

Definition: Image Gradient (Discrete)

For 2D function, *F(x,y)*, the partial derivative is:

$$\frac{\delta F(x,y)}{\delta x} = \lim_{\epsilon \to 0} \frac{F(x+\epsilon,y) - F(x,y)}{\epsilon}$$

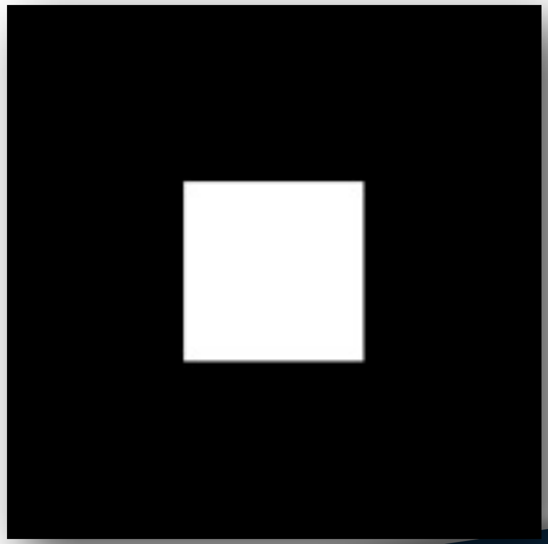
For discrete data, we can approximate using finite differences:

X : $$\frac{\delta F(x,y)}{\delta x} \approx \frac{F(x+1,y) - F(x,y)}{1}$$

Y : $$\frac{\delta F(x,y)}{\delta y} \approx F(x,y+1) - F(x,y)$$

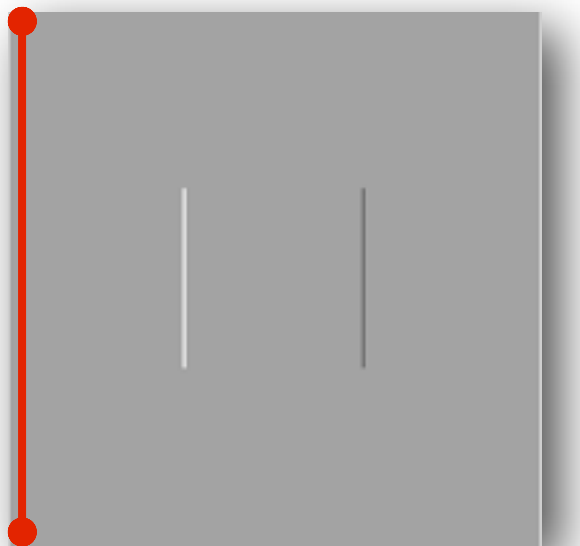
Slide adapted from Aaron Bobick

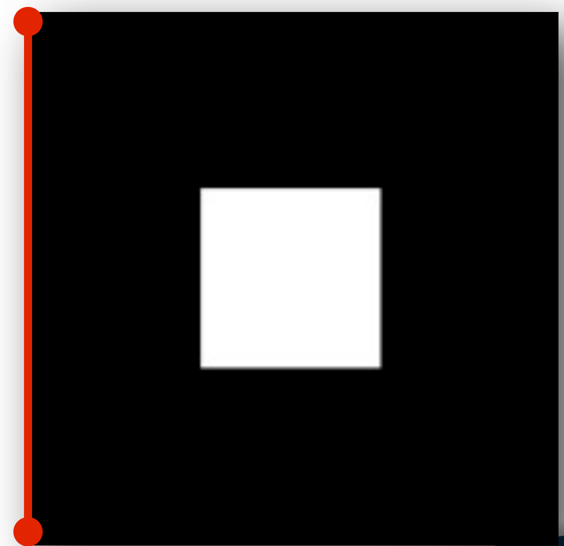
Differentiating an Image in X and Y



Differentiating an Image in X and Y

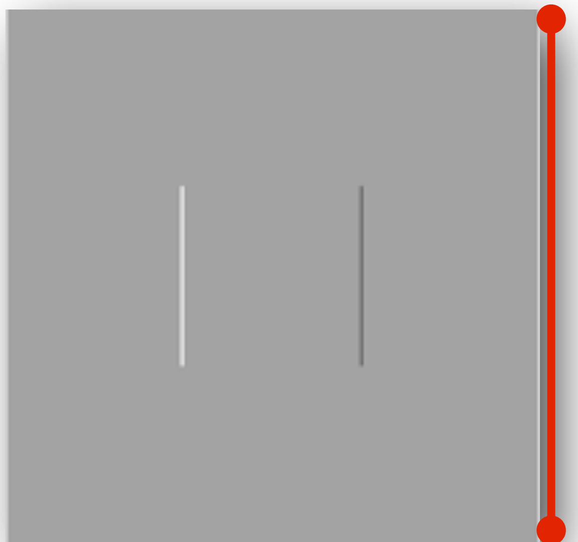
$$\frac{\delta F(x,y)}{\delta x} \approx F(x+1,y) - F(x,y)$$

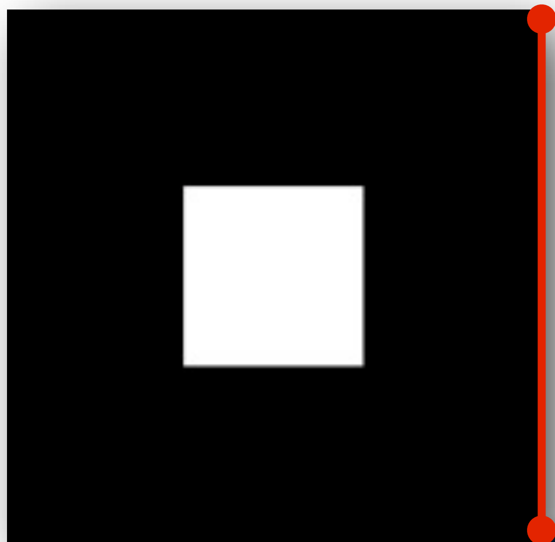




Differentiating an Image in X and Y

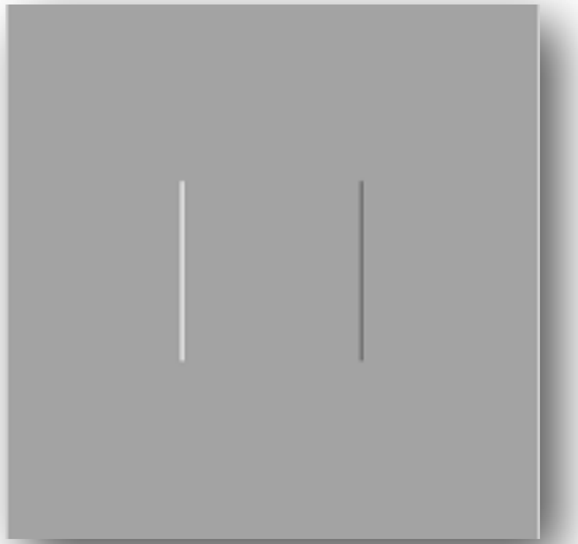
$$\frac{\delta F(x,y)}{\delta x} \approx F(x+1,y) - F(x,y)$$

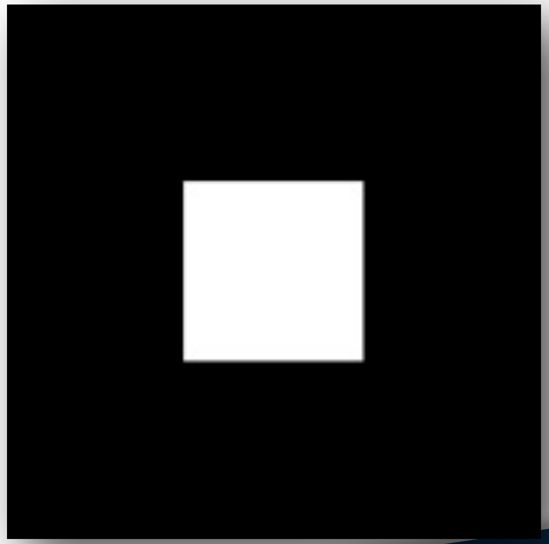




Differentiating an Image in X and Y

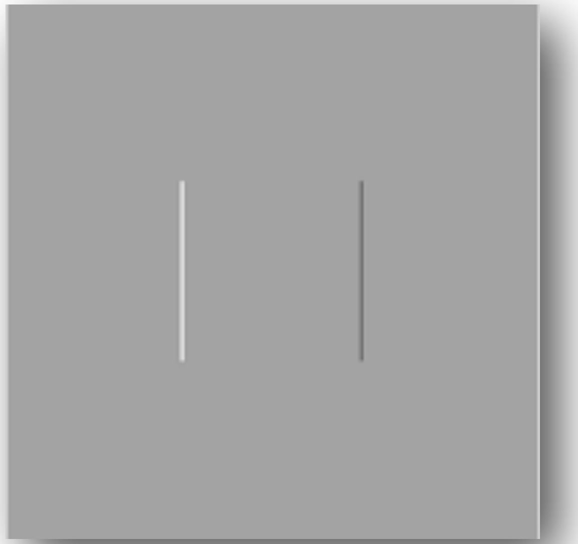
$$\frac{\delta F(x,y)}{\delta x} \approx F(x+1,y) - F(x,y)$$

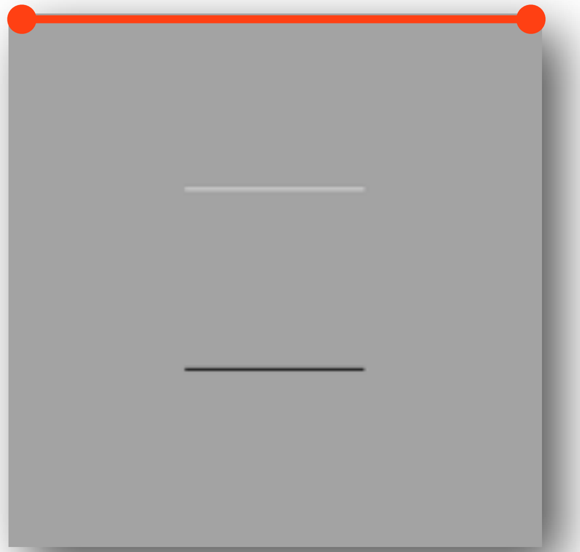


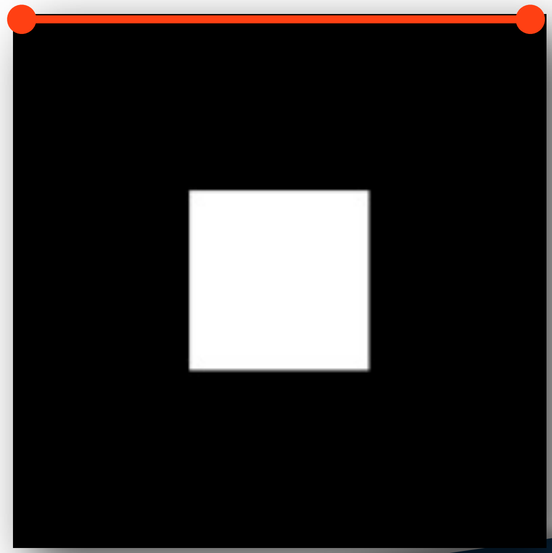


Differentiating an Image in X and Y

$$\frac{\delta F(x,y)}{\delta x} \approx F(x+1,y) - F(x,y) \qquad \frac{\delta F(x,y)}{\delta y} \approx F(x,y+1) - F(x,y)$$



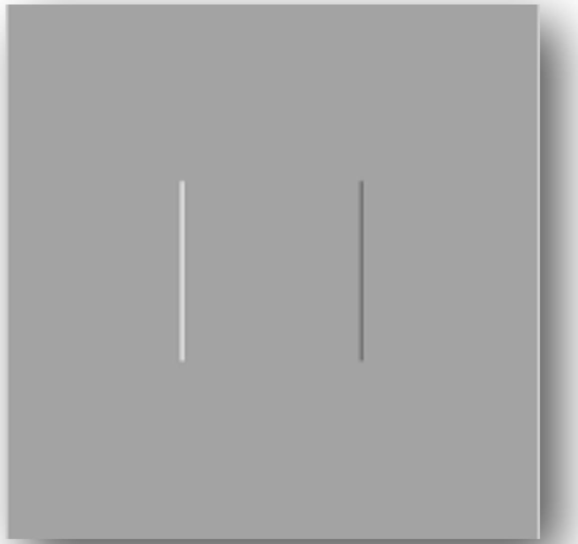


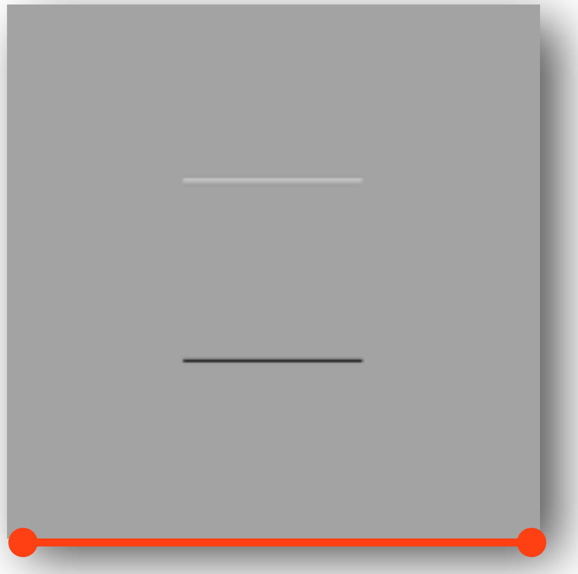


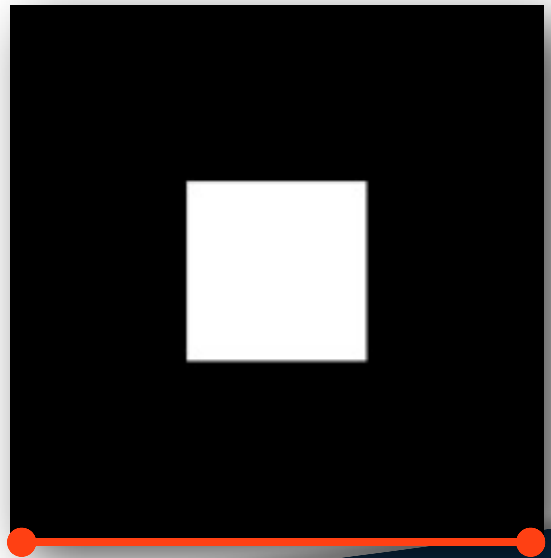
Differentiating an Image in X and Y

$$\frac{\delta F(x,y)}{\delta x} \approx F(x+1,y) - F(x,y)$$

$$\frac{\delta F(x,y)}{\delta y} \approx F(x,y+1) - F(x,y)$$



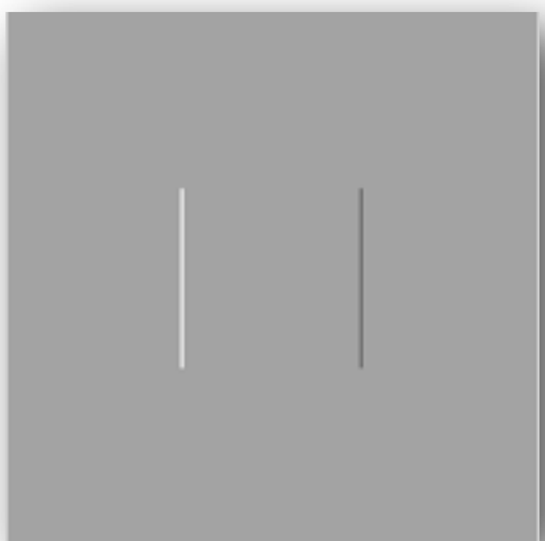


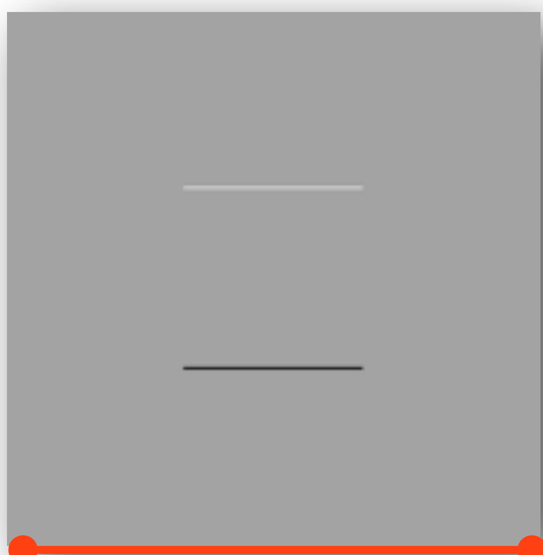


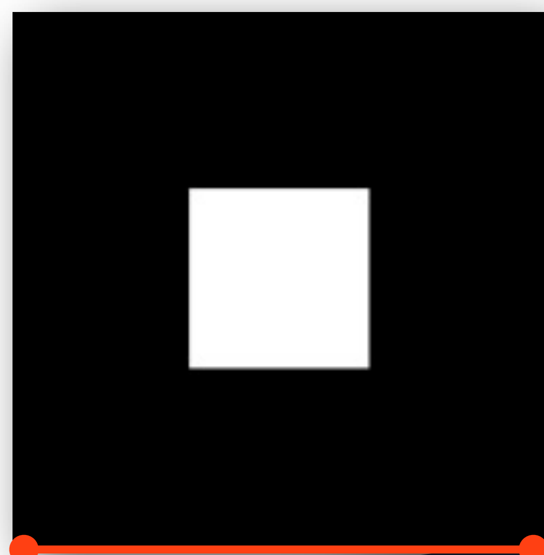
Differentiating an Image in X and Y

$$\frac{\delta F(x,y)}{\delta x} \approx F(x+1,y) - F(x,y)$$

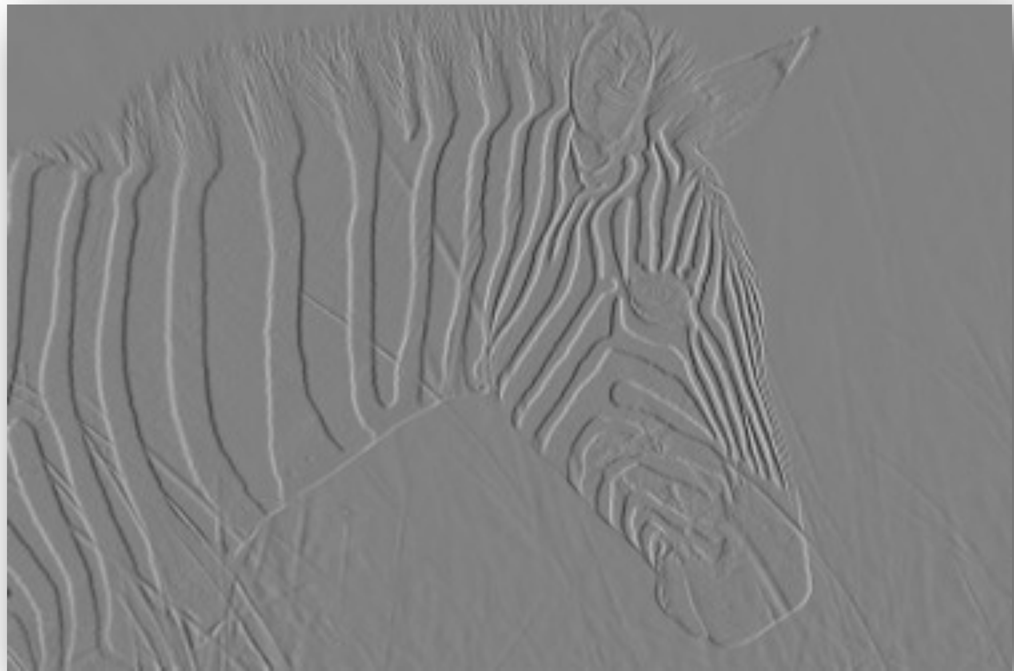
$$\frac{\delta F(x,y)}{\delta y} \approx F(x,y+1) - F(x,y)$$



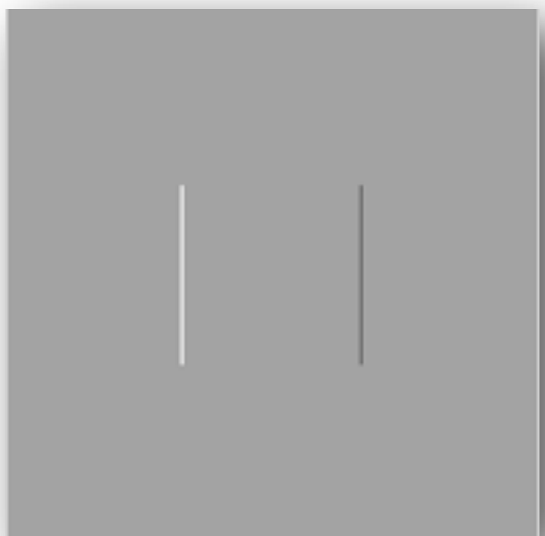


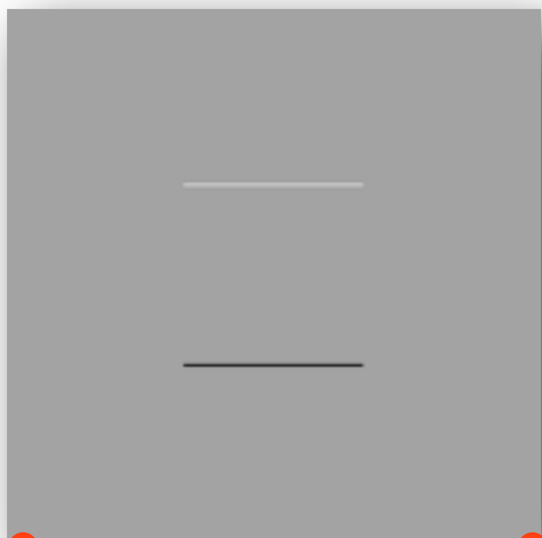


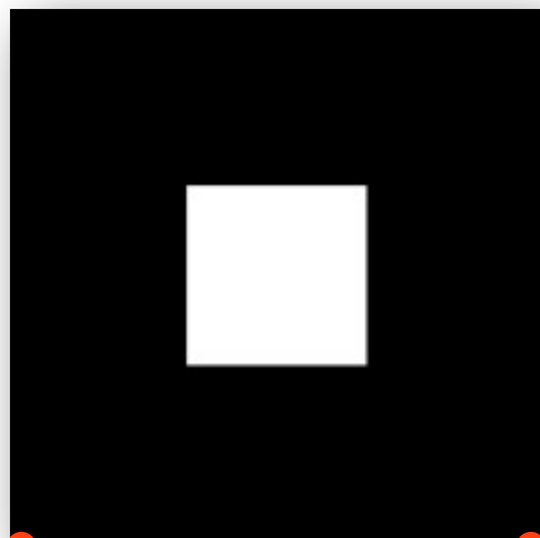
Differentiating an Image in X and Y



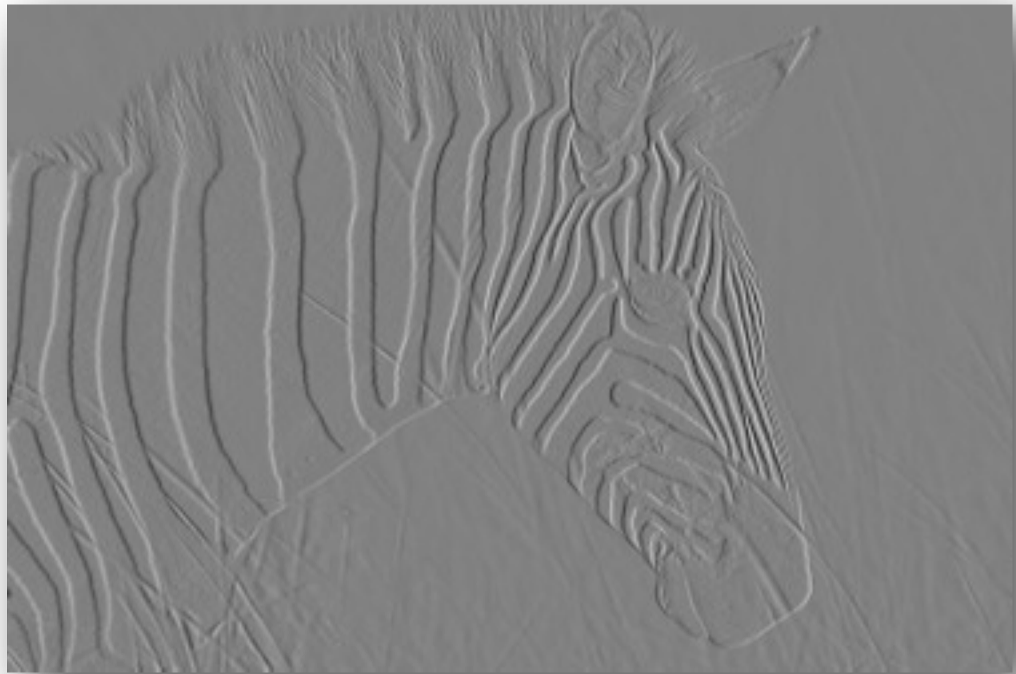
$$\frac{\delta F(x,y)}{\delta x} \approx F(x+1,y) - F(x,y) \qquad \frac{\delta F(x,y)}{\delta y} \approx F(x,y+1) - F(x,y)$$

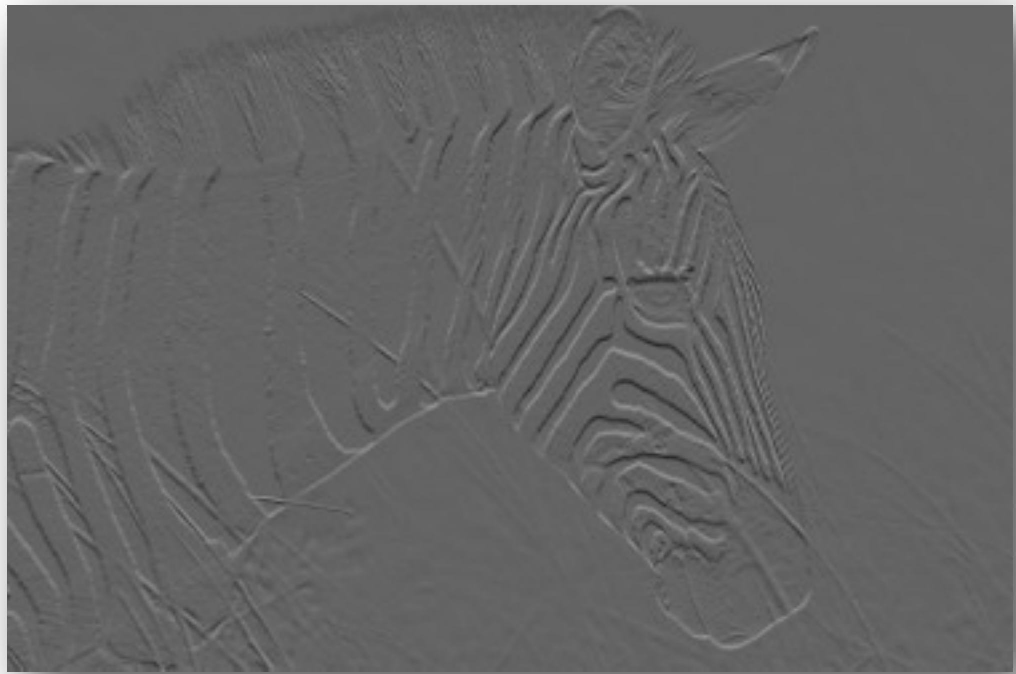






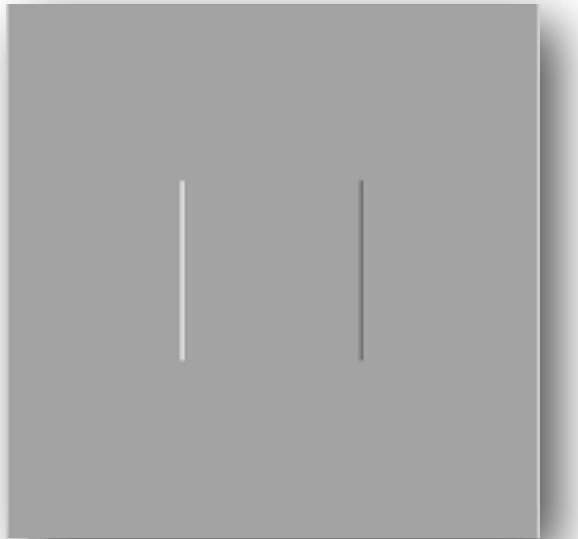
Differentiating an Image in X and Y

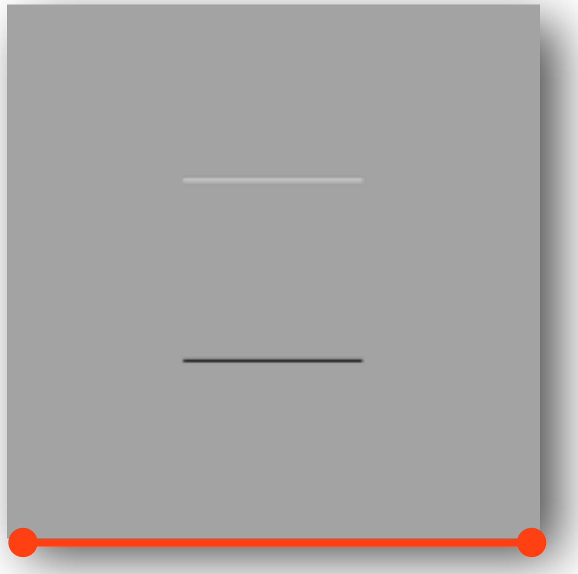




$$\frac{\delta F(x,y)}{\delta x} \approx F(x+1,y) - F(x,y)$$

$$\frac{\delta F(x,y)}{\delta y} \approx F(x,y+1) - F(x,y)$$





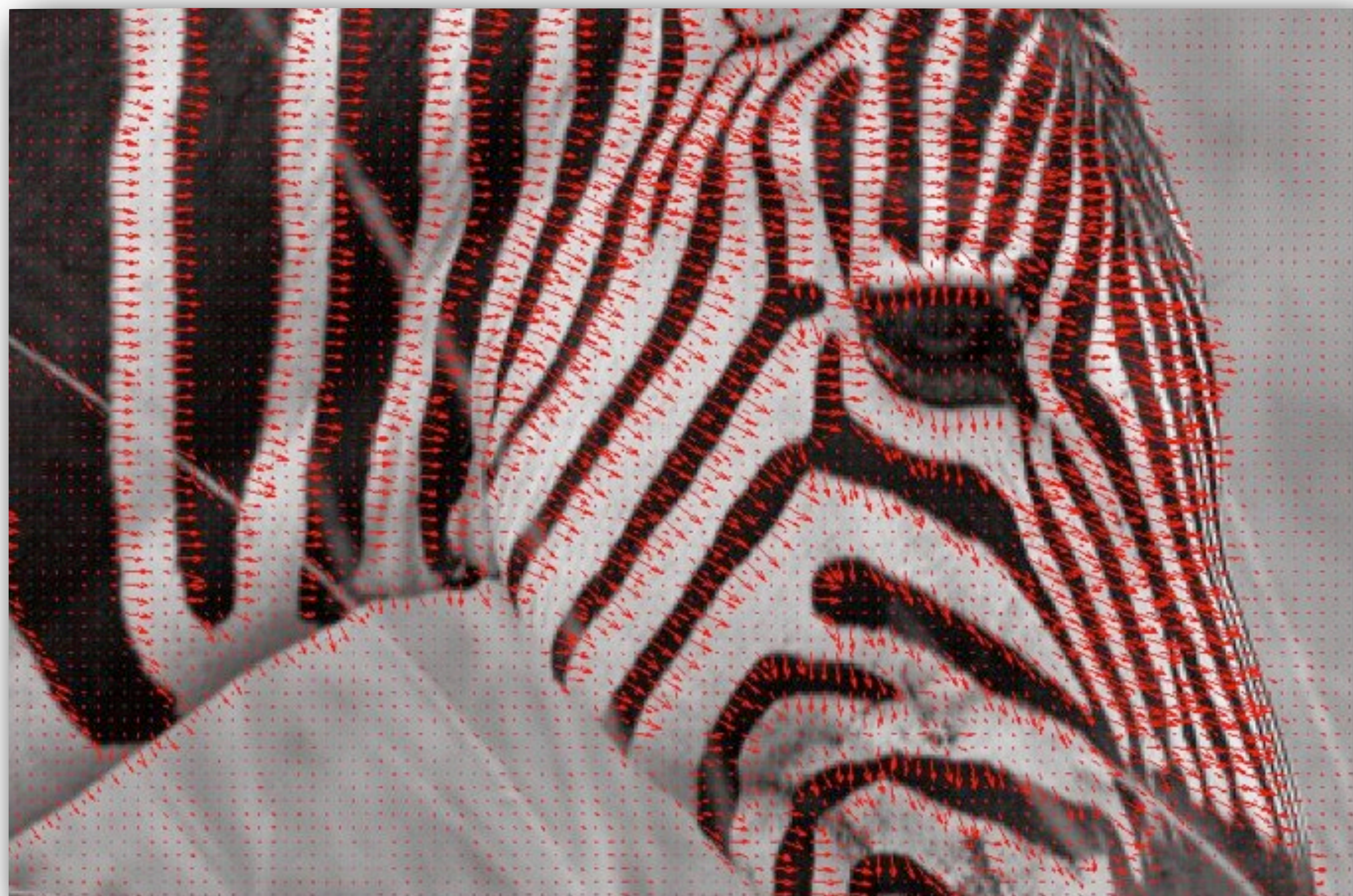
Gradient Images

$$\| \nabla F \| = \sqrt{\left(\frac{\delta F}{\delta x}\right)^2 + \left(\frac{\delta F}{\delta y}\right)^2} \qquad \theta = \tan^{-1}\left[\frac{\delta F}{\delta y} / \frac{\delta F}{\delta x}\right]$$

Gradient Images

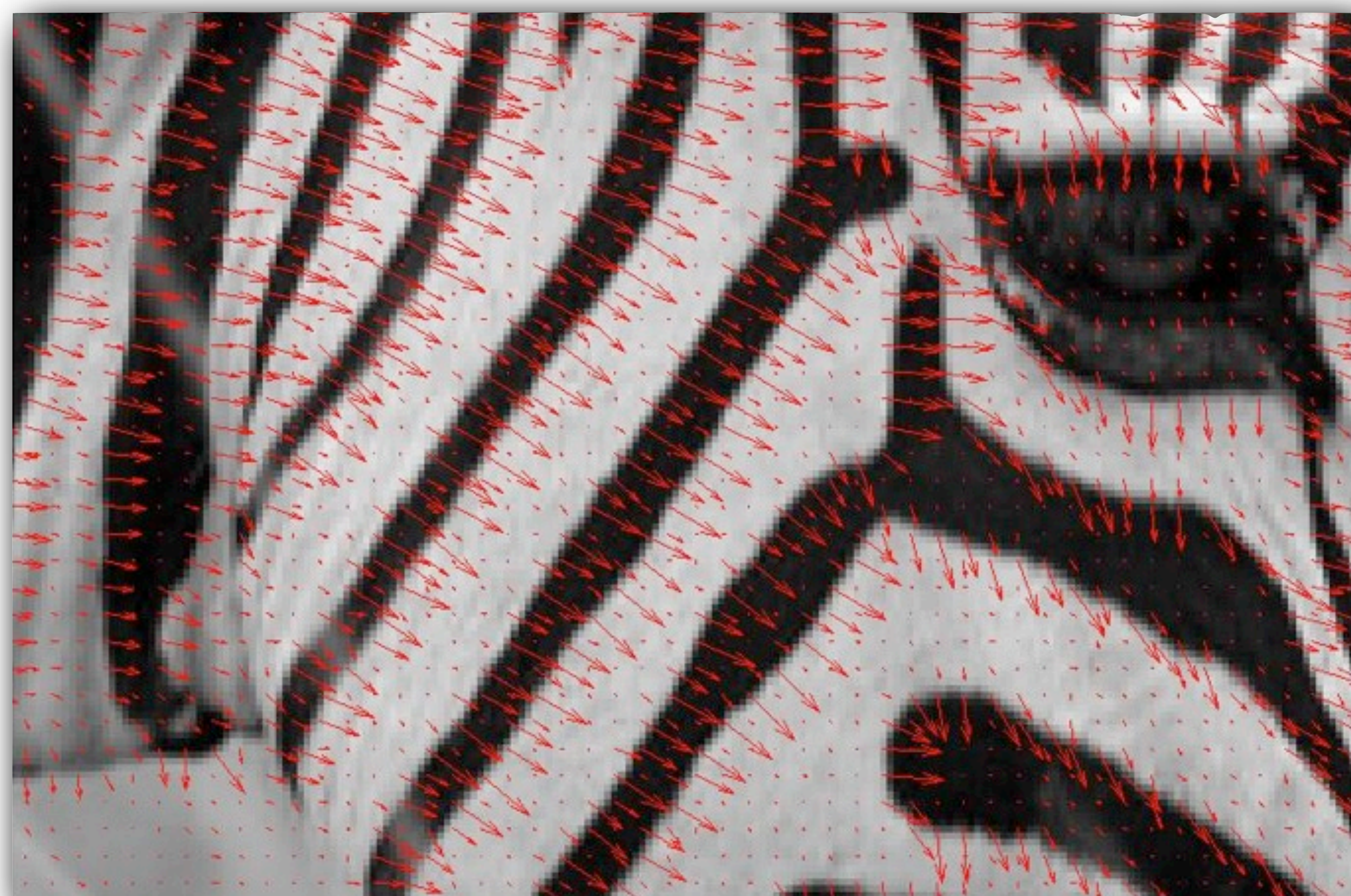
$$\| \nabla F \| = \sqrt{\left(\frac{\delta F}{\delta x}\right)^2 + \left(\frac{\delta F}{\delta y}\right)^2} \qquad \theta = \tan^{-1}\left[\frac{\delta F}{\delta y} / \frac{\delta F}{\delta x}\right]$$

Gradient Images



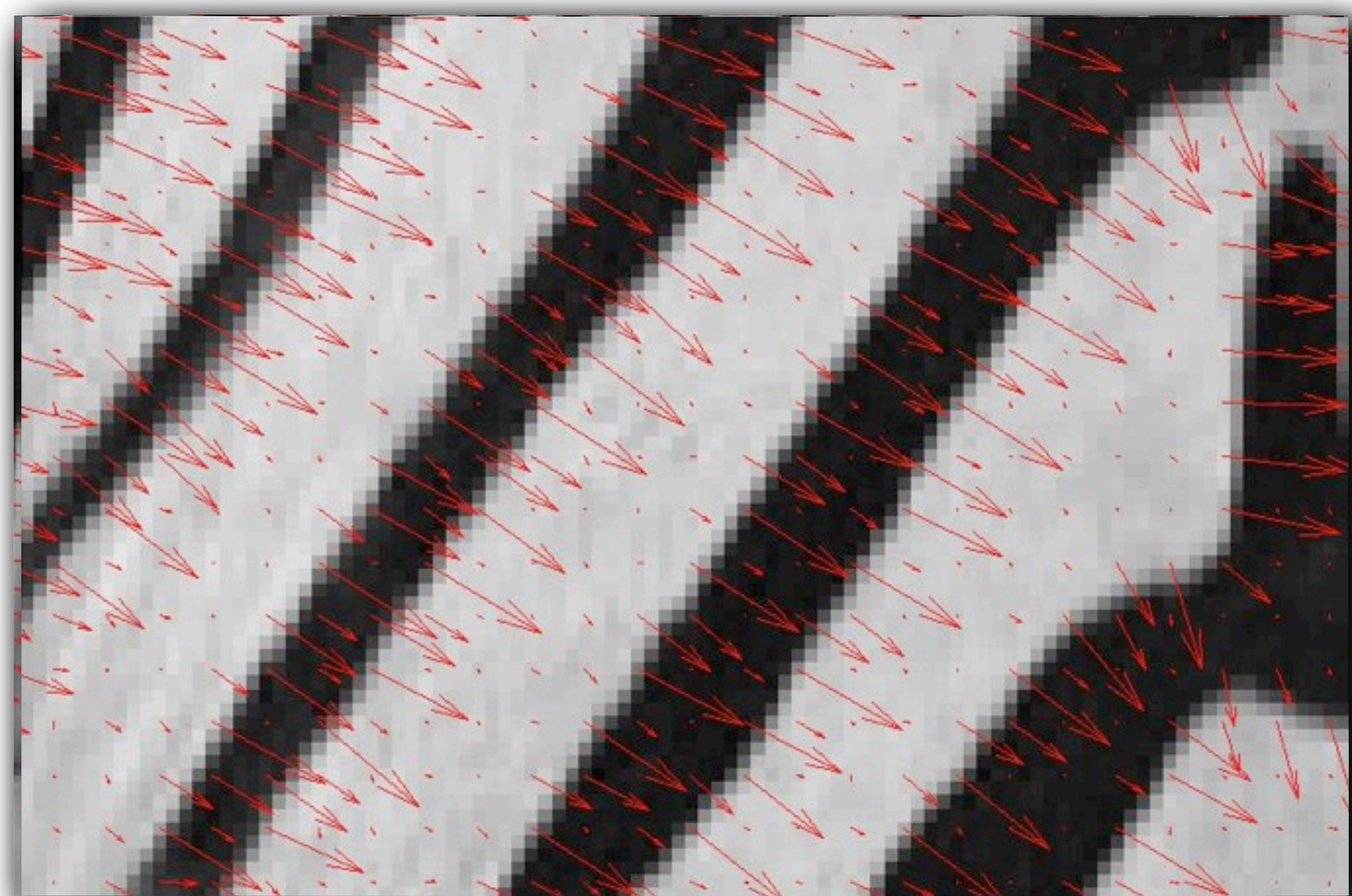
$$\| \nabla F \| = \sqrt{\left(\frac{\delta F}{\delta x}\right)^2 + \left(\frac{\delta F}{\delta y}\right)^2} \qquad \theta = \tan^{-1}\left[\frac{\delta F}{\delta y} \Big/ \frac{\delta F}{\delta x}\right]$$

Gradient Images



$$\| \nabla F \| = \sqrt{\left(\frac{\delta F}{\delta x}\right)^2 + \left(\frac{\delta F}{\delta y}\right)^2} \qquad \theta = \tan^{-1}\left[\frac{\delta F}{\delta y} / \frac{\delta F}{\delta x}\right]$$

Gradient Images



$$\| \nabla F \| = \sqrt{\left(\frac{\delta F}{\delta x}\right)^2 + \left(\frac{\delta F}{\delta y}\right)^2} \qquad \theta = \tan^{-1}\left[\frac{\delta F}{\delta y} / \frac{\delta F}{\delta x}\right]$$

16

Summary

- ★ Introduced the concept of finding Good Features of an image to support matching across images.
- ★ Presented Edges as one such feature, that are determined by measuring changes across an image.
- ★ Presented the Concept of Gradients and introduced the use of derivatives over images to compute the gradient over an image.

Next Class

- ★ Image Analysis: Edge Detection using Gradients

Credits

- ★ Matlab™ software by Mathworks Inc.
- ★ Some Slides adapted from Aaron Bobick
- ★ For more information, see Szeliski OR Forsyth & Ponce Text Book.
- ★ Images
  - Image Courtesy Professor Henrik Christensen
  - Images used from USC's Signal and Image Processing Institute's Image Database
  - Creative Commons Search
  - http://www.flickr.com/photos/lipkee/2904603582/

Computational Photography

Dr. Irfan Essa
Professor
School of Interactive Computing

Study the basics of computation and its impact on the entire workflow of photography, from capturing, manipulating and collaborating on, and sharing photographs.