Georgia Institute of Technology

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.

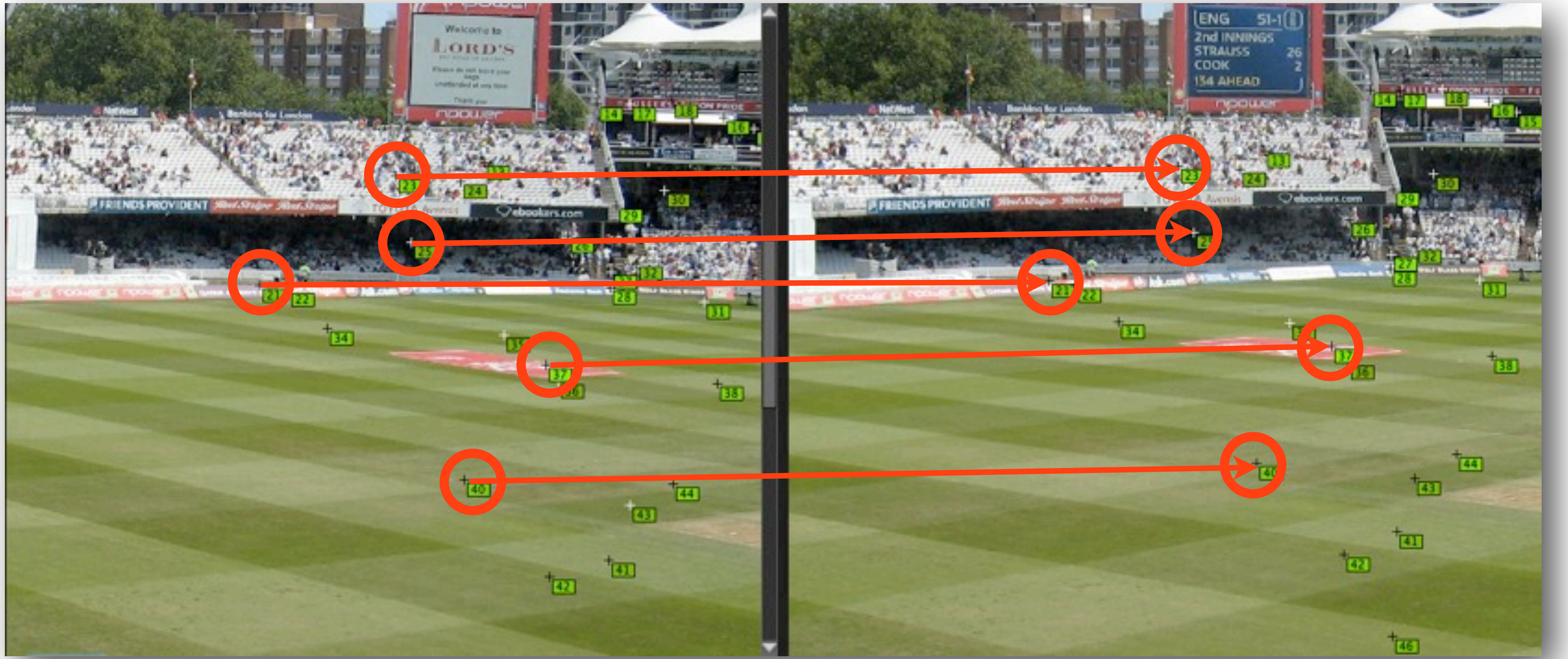
Feature Detection and Matching

Dr. Irfan Essa
Professor
School of Interactive Computing

Methods for Detecting Features in Images that can be Matched for Registration and Alignment.

Lesson Objectives

- ★ Describe in your own words two (2) benefits of Feature Detection and Matching in images
- ★ Describe in your own words the four (4) Characteristics of Good Features.
- ★ Explain in your own words why corners are Good Features.
- ★ Outline the five (5) steps of Harris Corner Detector Algorithm
- ★ Outline the four (4) stages of a SIFT detector



REVIEW: Detection and Matching

Image Matching

Image Matching

Image Matching

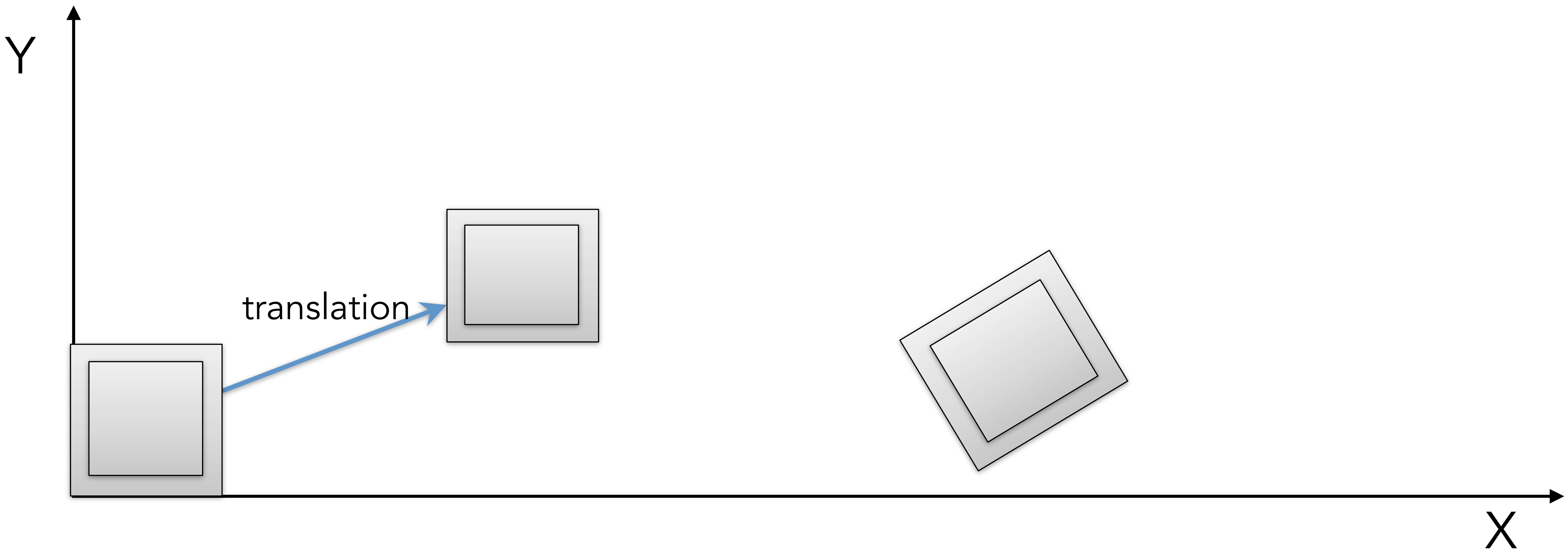


Image Matching

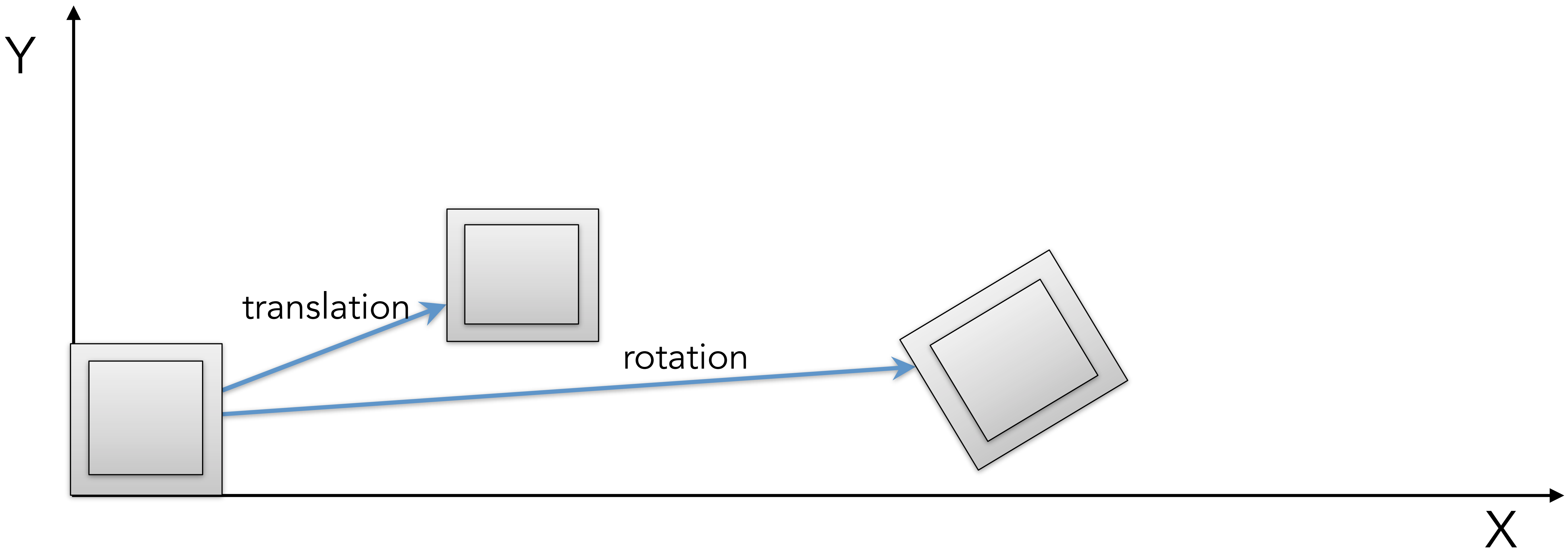


Image Matching

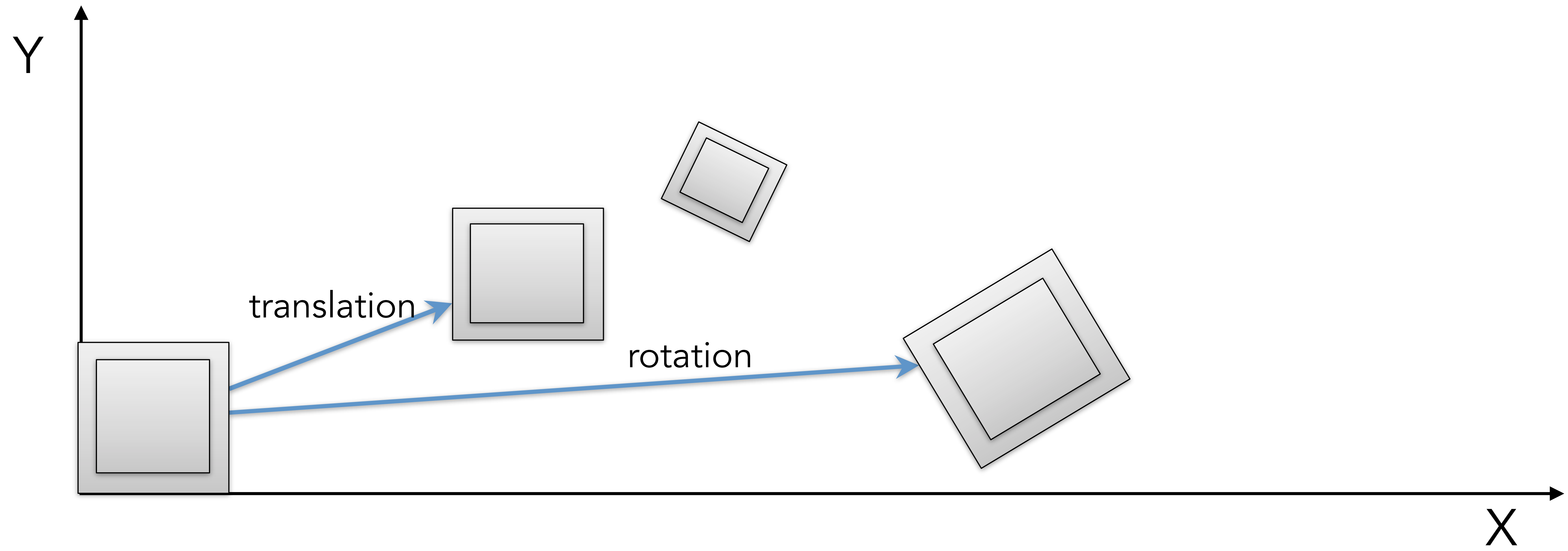


Image Matching

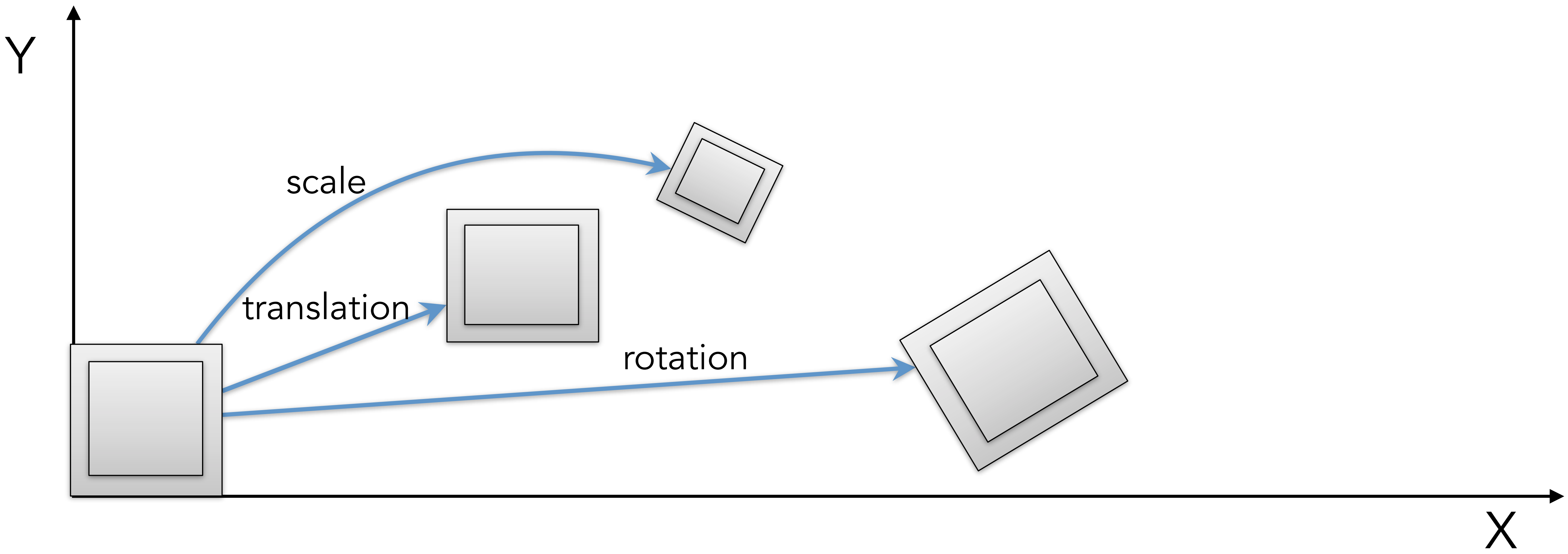


Image Matching

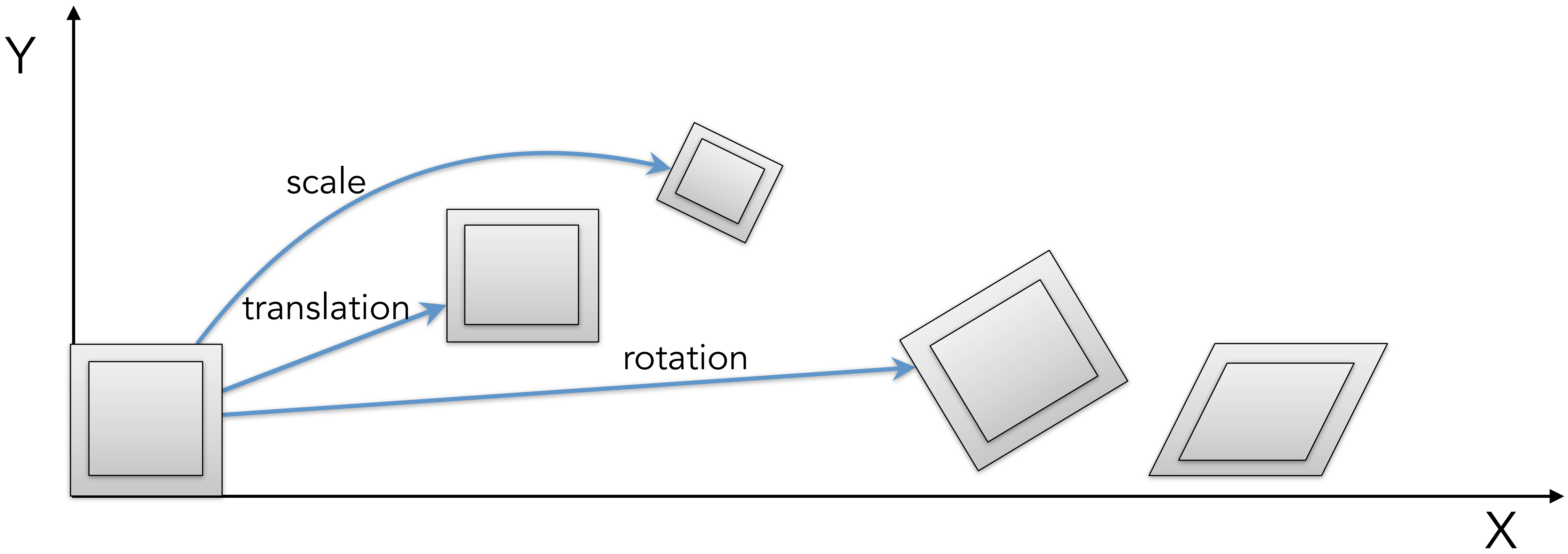


Image Matching

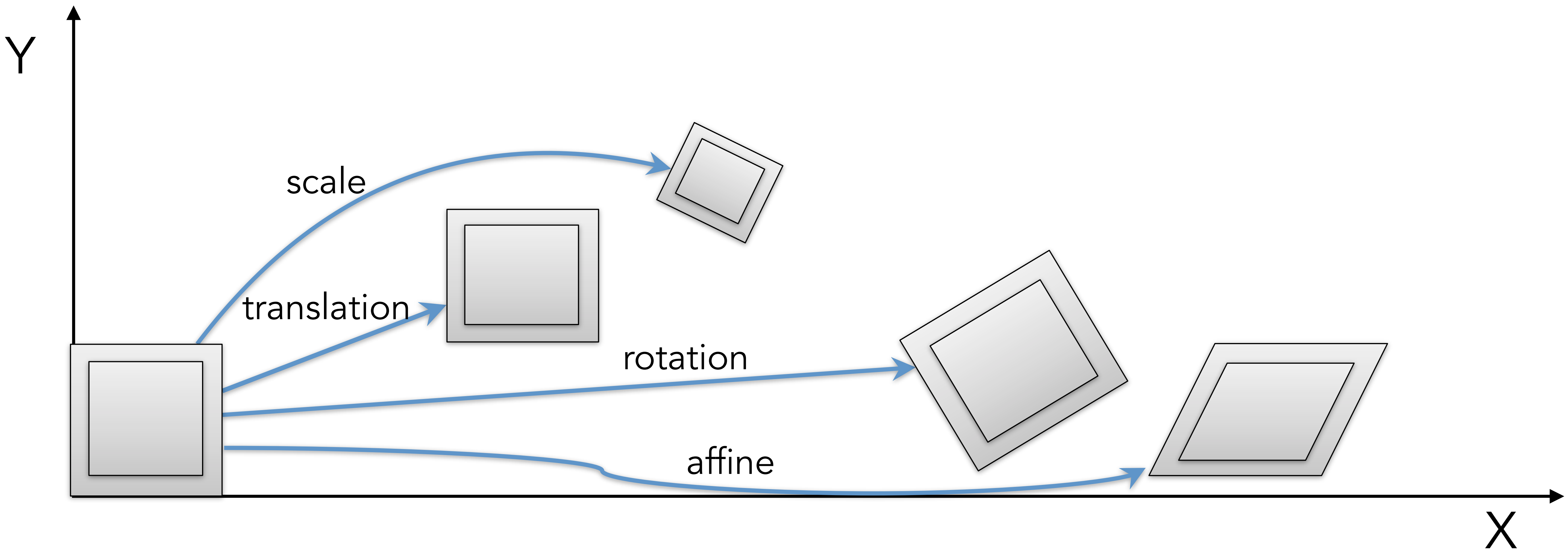


Image Matching

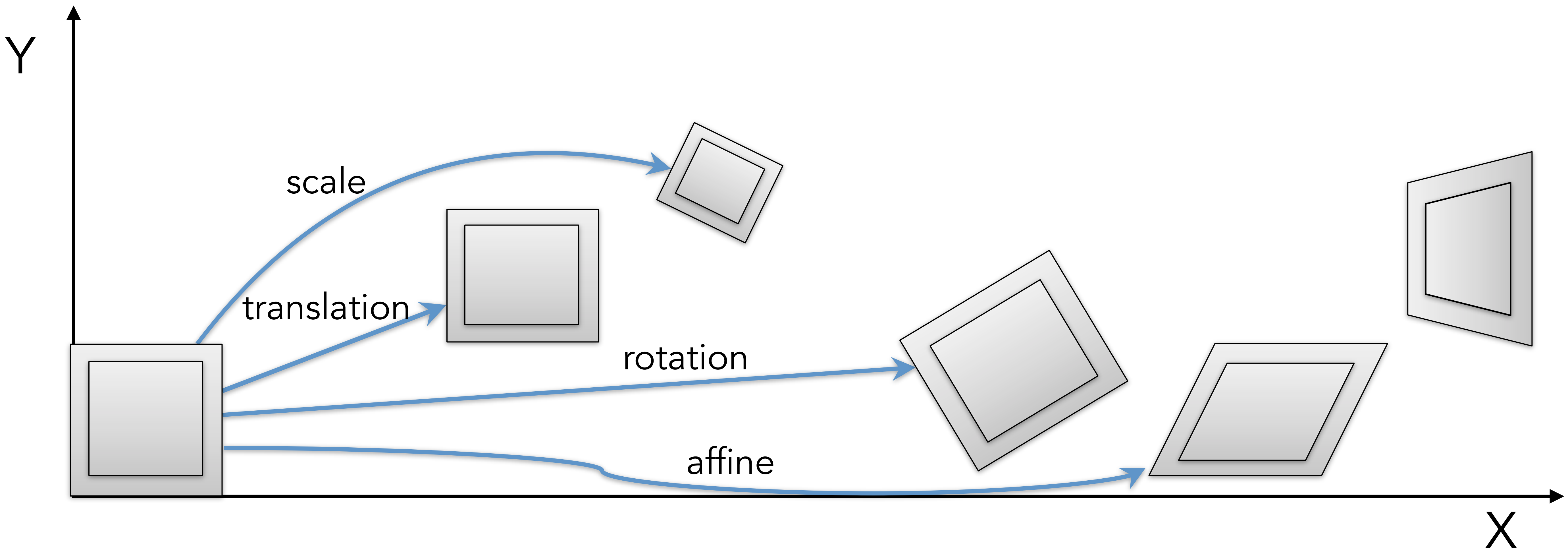


Image Matching

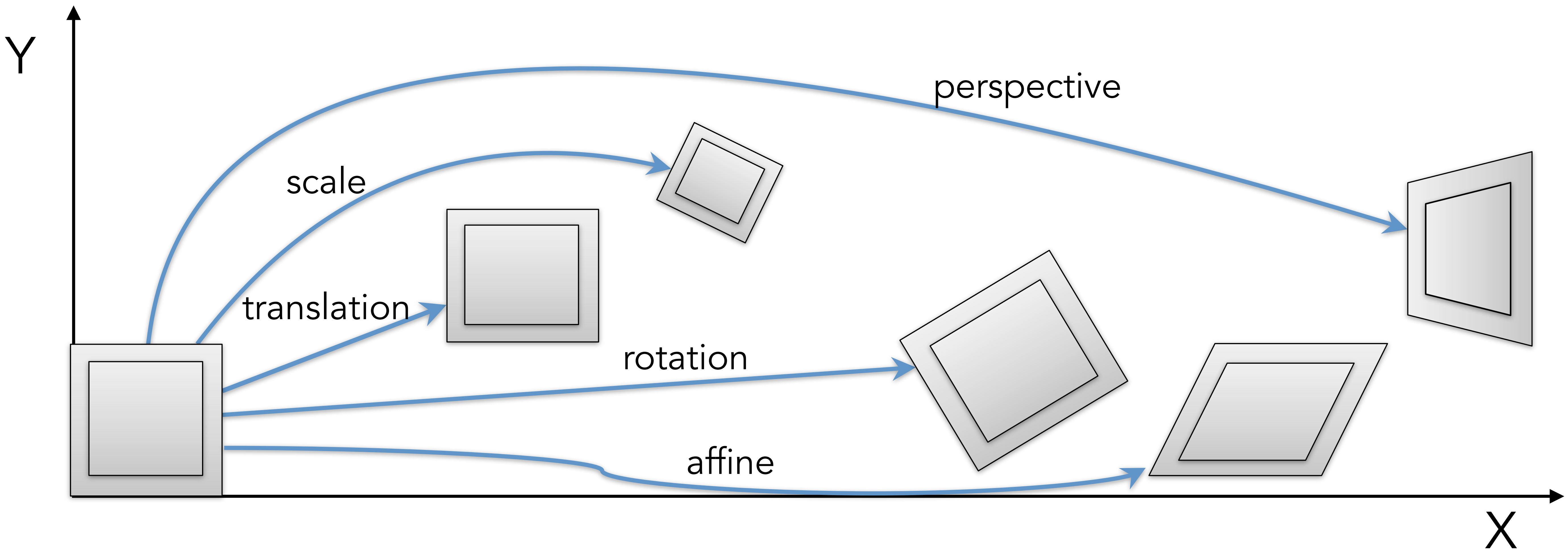
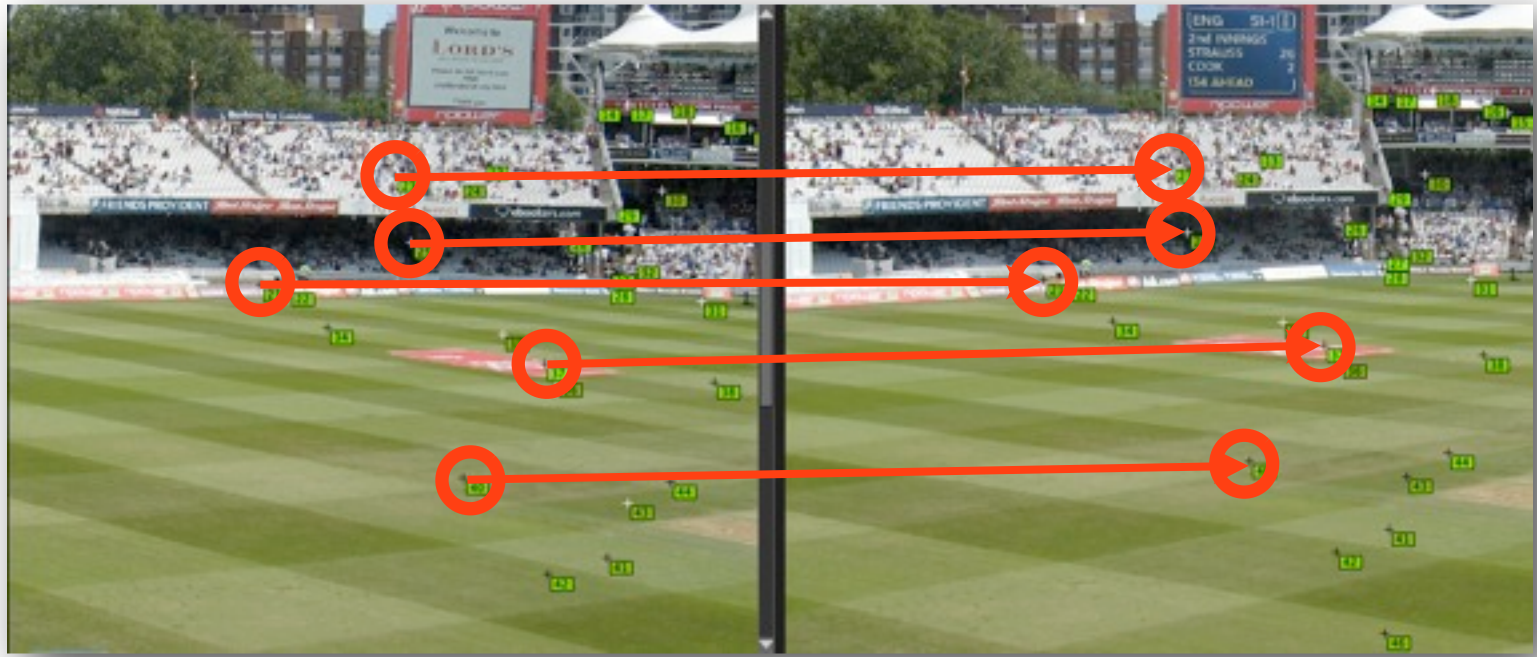


Image Matching

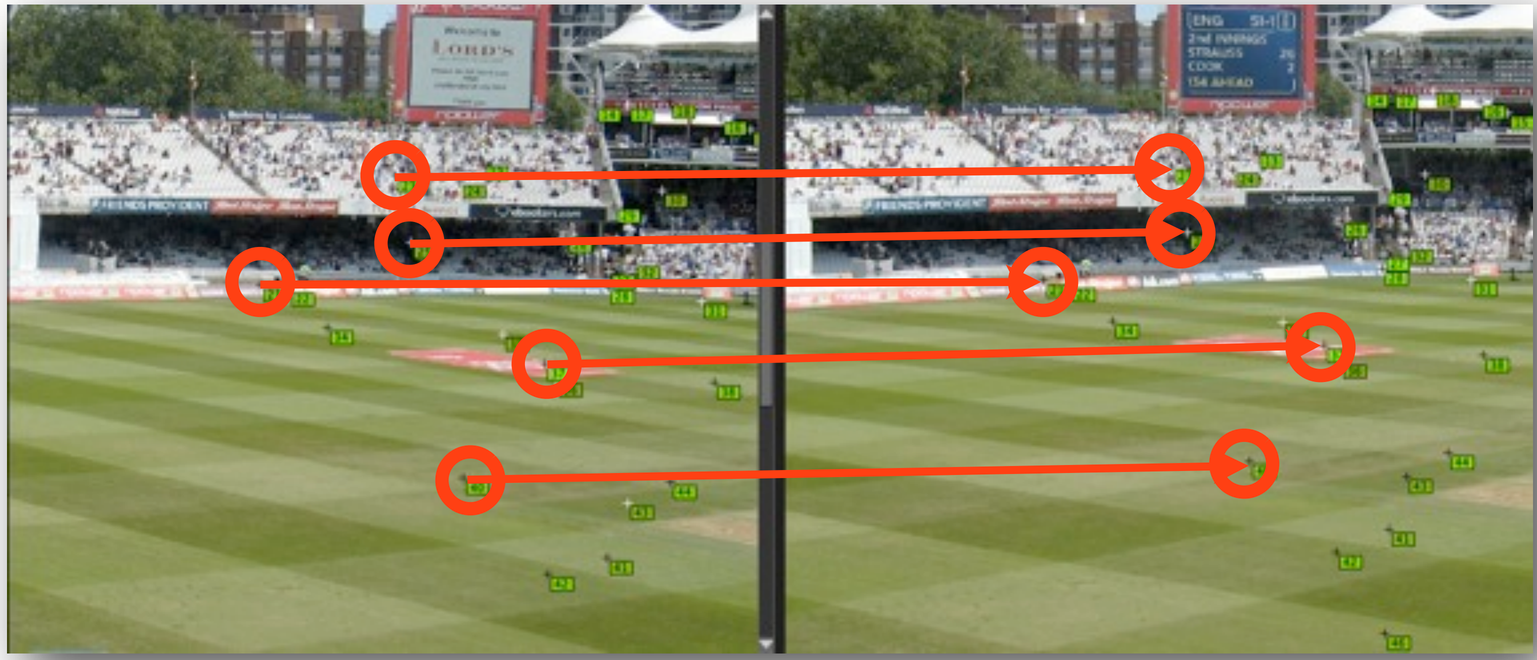
Finding Features



Finding Features

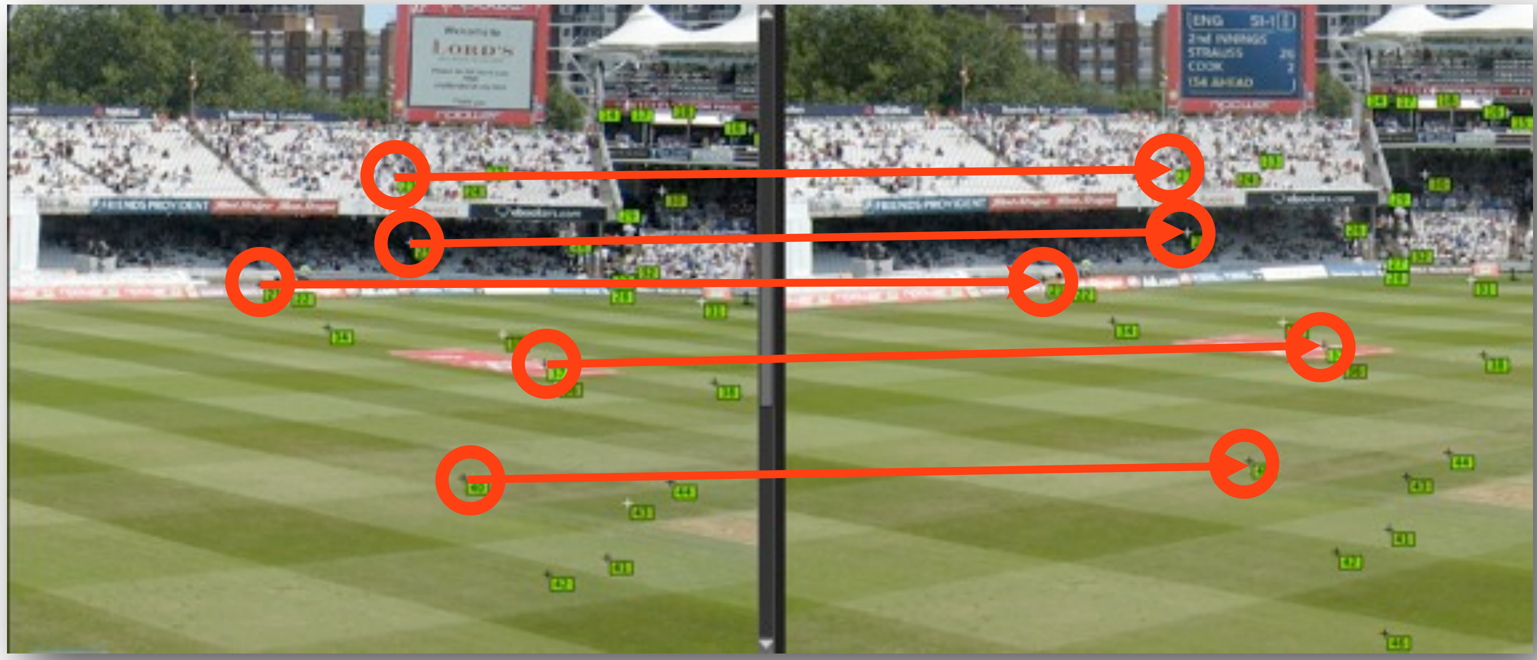
★ Goal: Find points in an image that can be:

- Found in other images
- Found precisely – well localized
- Found reliably – well matched



Finding Features

- Goal: Find points in an image that can be:
  - Found in other images
  - Found precisely – well localized
  - Found reliably – well matched
- Why?
  - Align Images
  - Build a Panorama,
  - Etc.



Georgia Institute of Technology

Characteristics of Good Features

Slide motivated by Aaron Bobick

Characteristics of Good Features

★ Repeatability/Precision

- The same feature can be found in several images despite geometric and photometric transformations

Slide motivated by Aaron Bobick

Characteristics of Good Features

- ★ Repeatability/Precision
  - The same feature can be found in several images despite geometric and photometric transformations
- ★ Saliency/Matchability
  - Each feature has a distinctive description

Slide motivated by Aaron Bobick

Characteristics of Good Features

- ★ Repeatability/Precision
  - The same feature can be found in several images despite geometric and photometric transformations
- ★ Saliency/Matchability
  - Each feature has a distinctive description
- ★ Compactness and efficiency
  - Many fewer features than image pixels

Slide motivated by Aaron Bobick

Characteristics of Good Features

- Repeatability/Precision
  - The same feature can be found in several images despite geometric and photometric transformations
- Saliency/Matchability
  - Each feature has a distinctive description
- Compactness and efficiency
  - Many fewer features than image pixels
- Locality
  - A feature occupies a relatively small area of the image; robust to clutter and occlusion

Slide motivated by Aaron Bobick

Find Corners

Find Corners

★ Key property: in the region around a corner, image gradient has two or more dominant directions

Find Corners

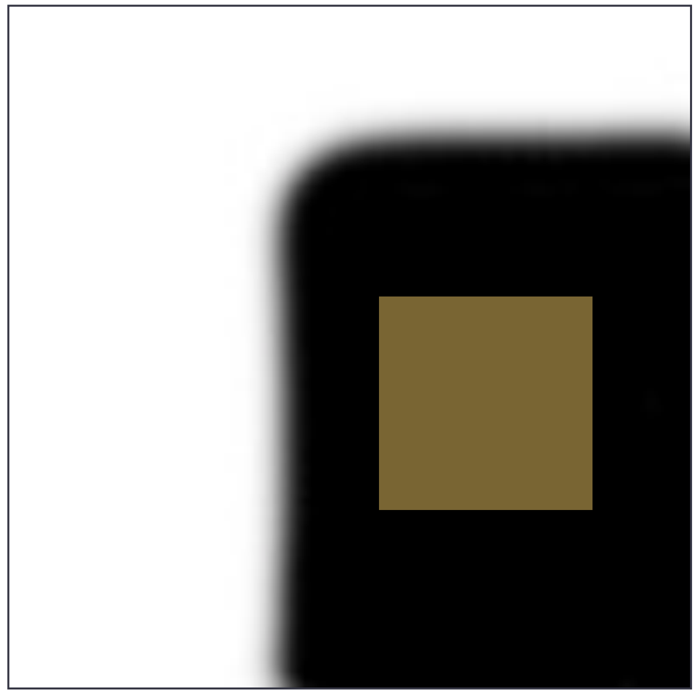
- ★ Key property: in the region around a corner, image gradient has two or more dominant directions
- ★ Corners are repeatable and distinctive

Find Corners

- ★ Key property: in the region around a corner, image gradient has two or more dominant directions
- ★ Corners are repeatable and distinctive
- ★ Harris and Stephens (1988)

8

Corner Detection: The Basics

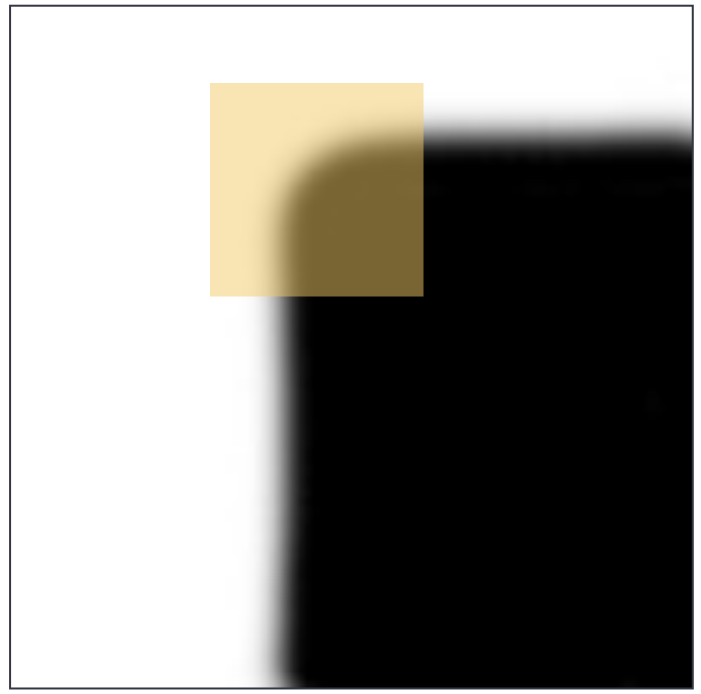


"flat"

No change in any directions

"edge"

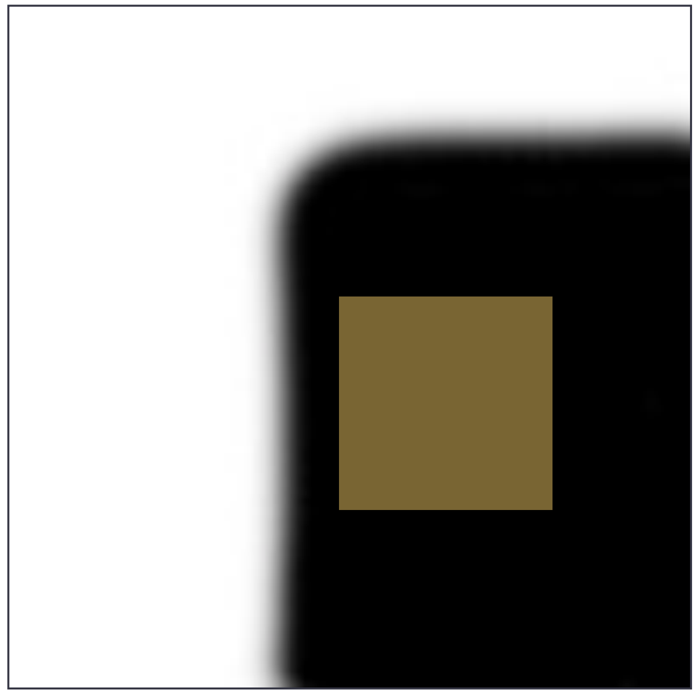
no change along the edge direction

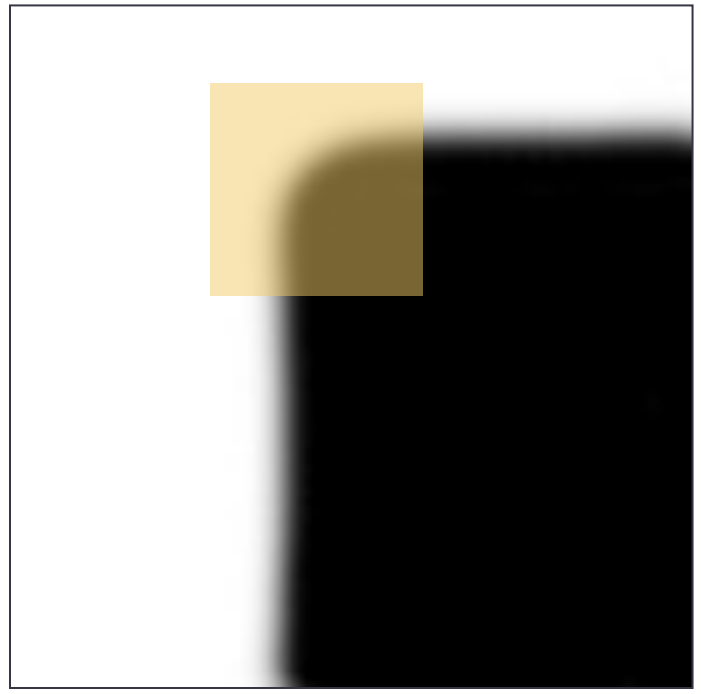


"corner"

change in all directions

Slide motivated by Alyosha Efros





Corner Detection: The Basics

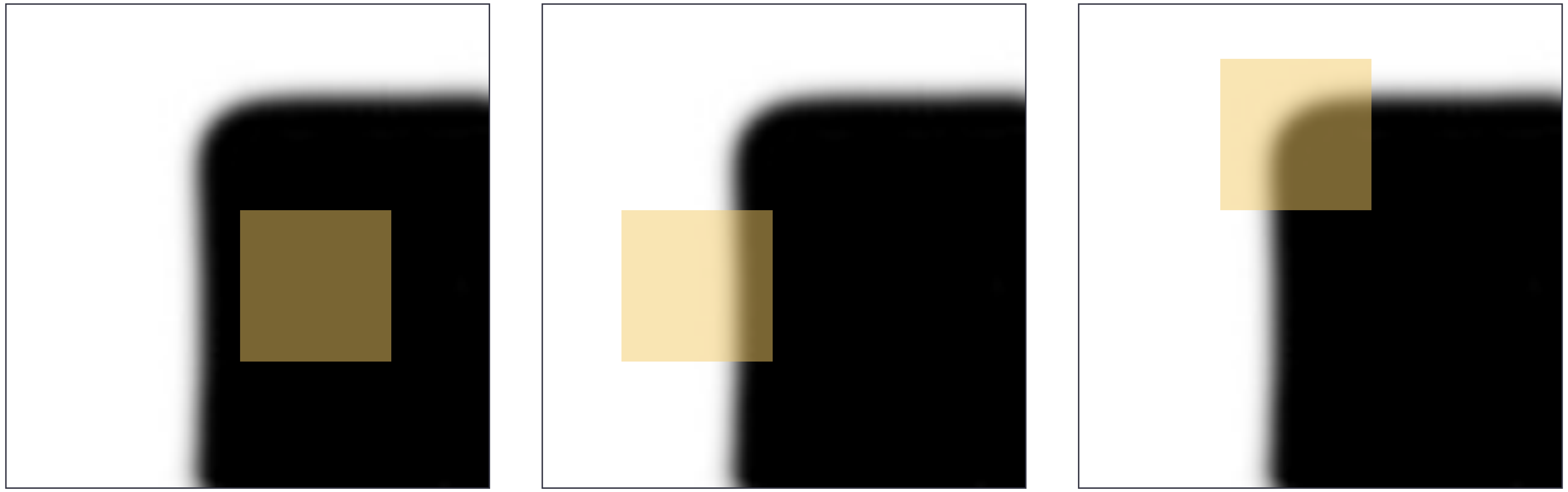
"flat"
No change in any directions

"edge"
no change along the edge direction

"corner"
change in all directions

Slide motivated by Alyosha Efros

Corner Detection: The Basics



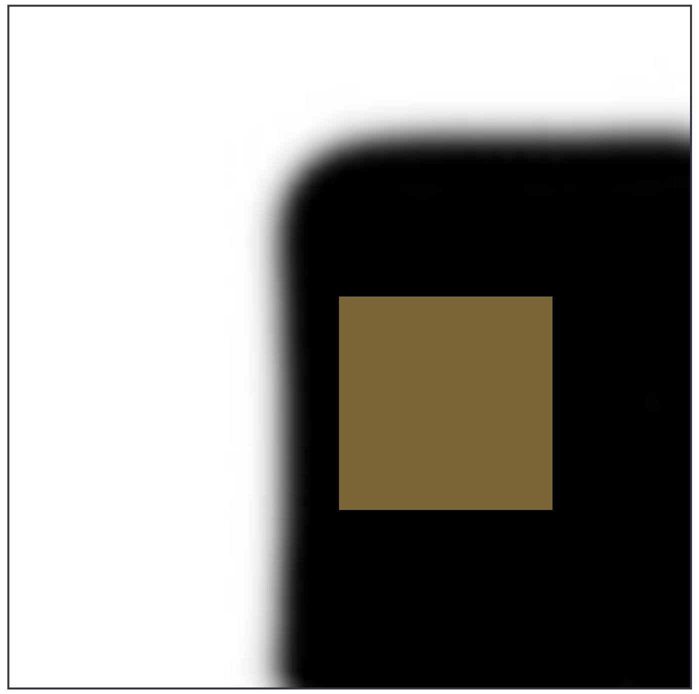
"flat"
No change in any directions

"edge"
no change along the edge direction

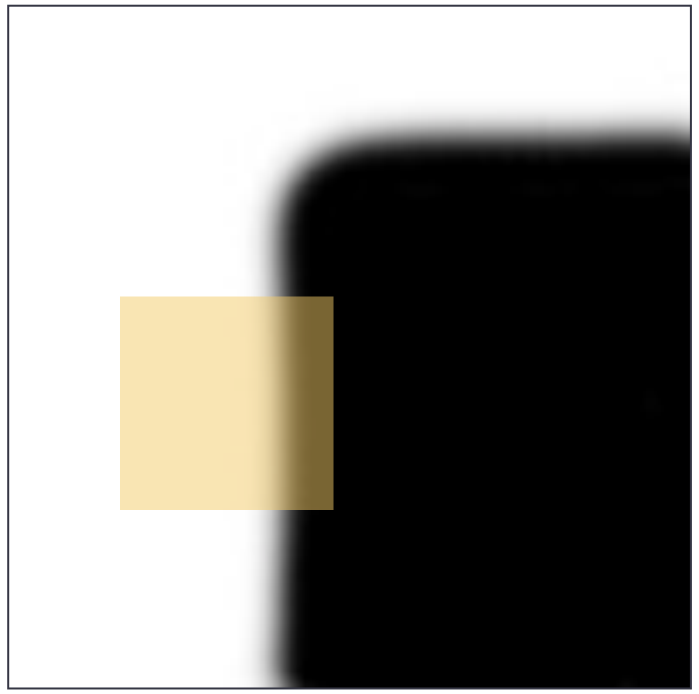
"corner"
change in all directions

Slide motivated by Alyosha Efros

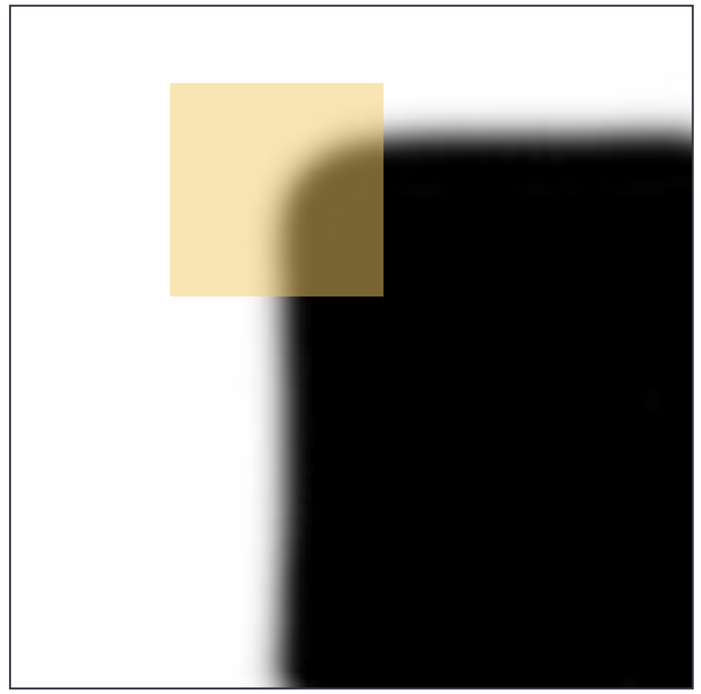
Corner Detection: The Basics



"flat"
No change in any directions



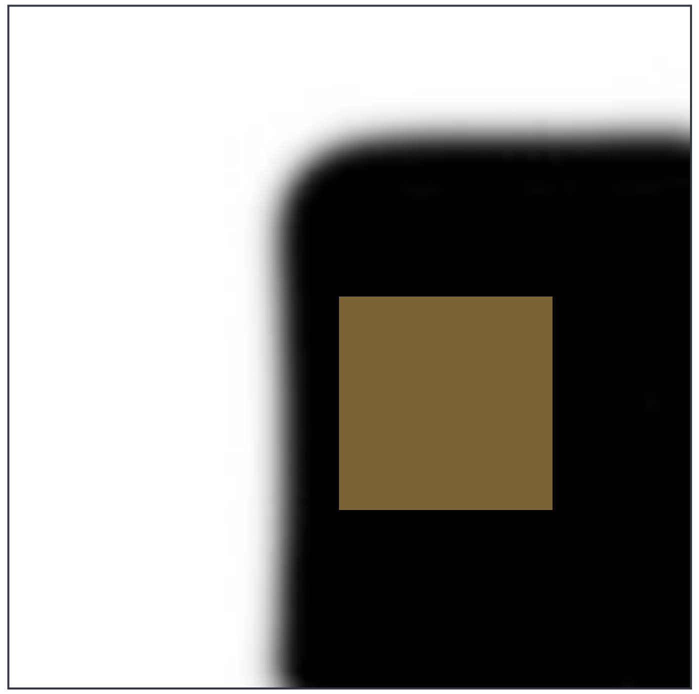
"edge"
no change along the edge direction



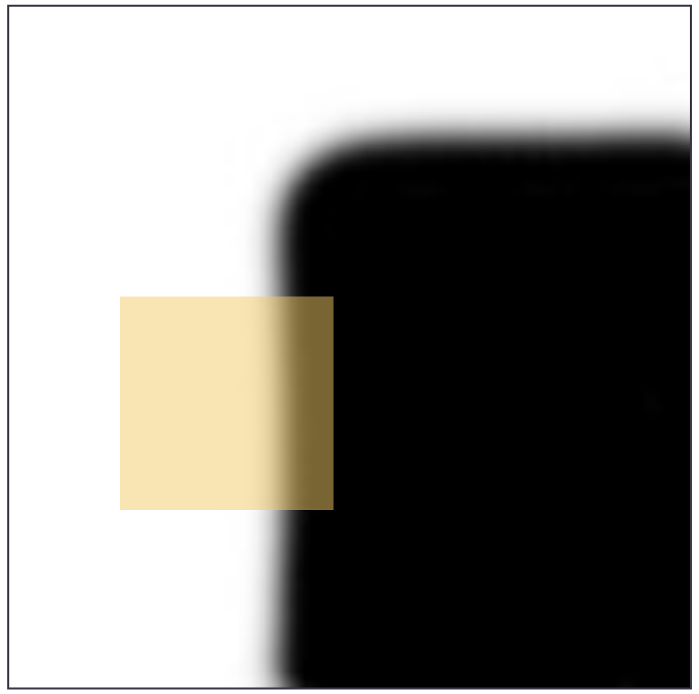
"corner"
change in all directions

Slide motivated by Alyosha Efros

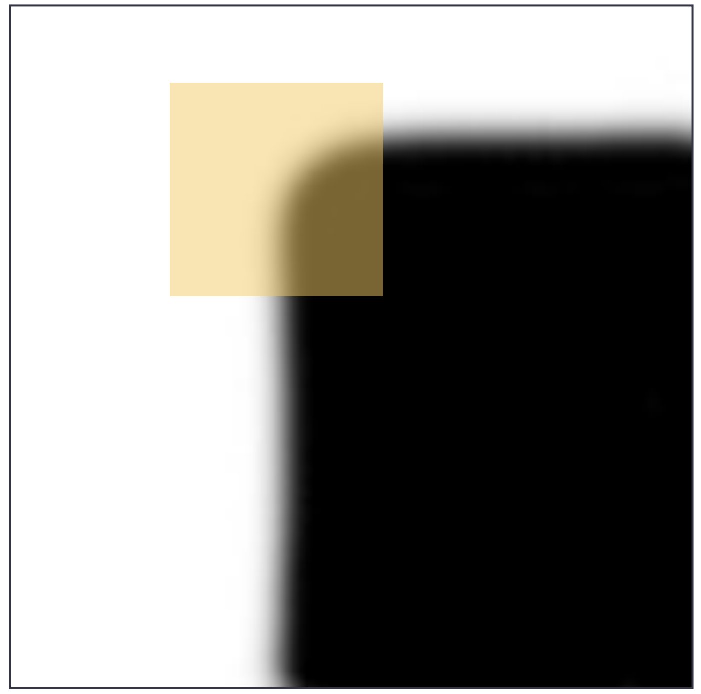
Corner Detection: The Basics



"flat"
No change in any directions



"edge"
no change along the edge direction

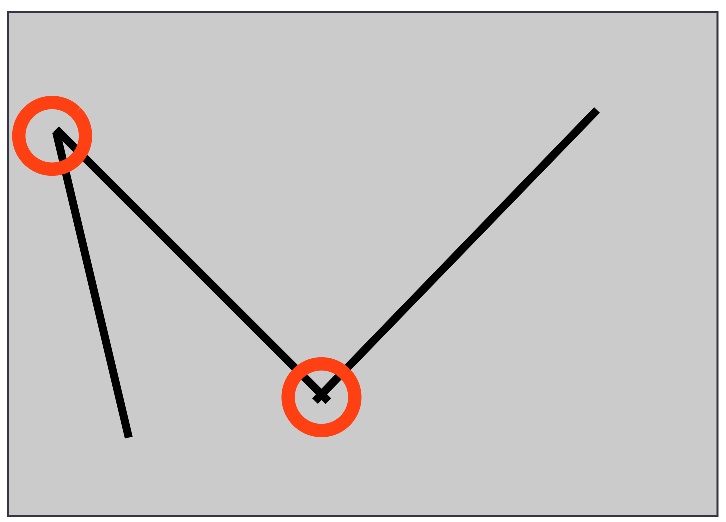


"corner"
change in all directions

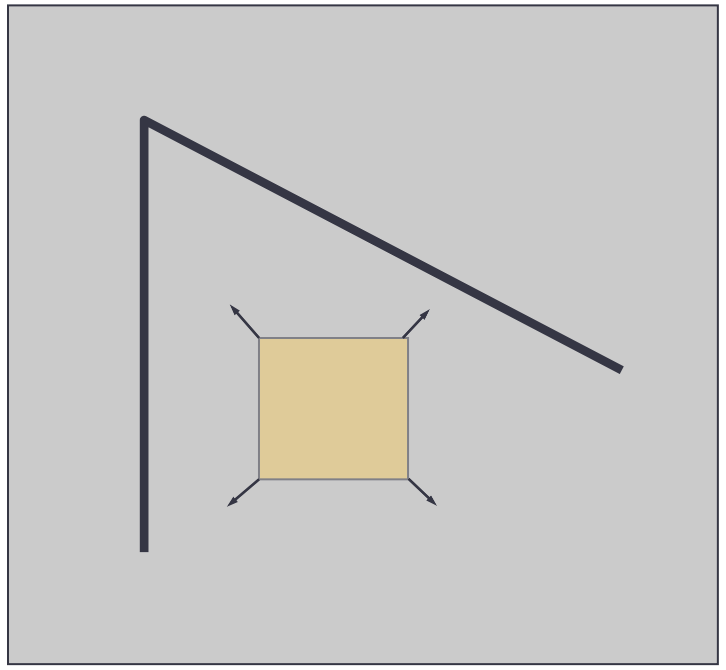
- ★ Possible to recognize a point by looking through a small window
- ★ Shifting a window in any direction gives a large change in intensity

Slide motivated by Alyosha Efros

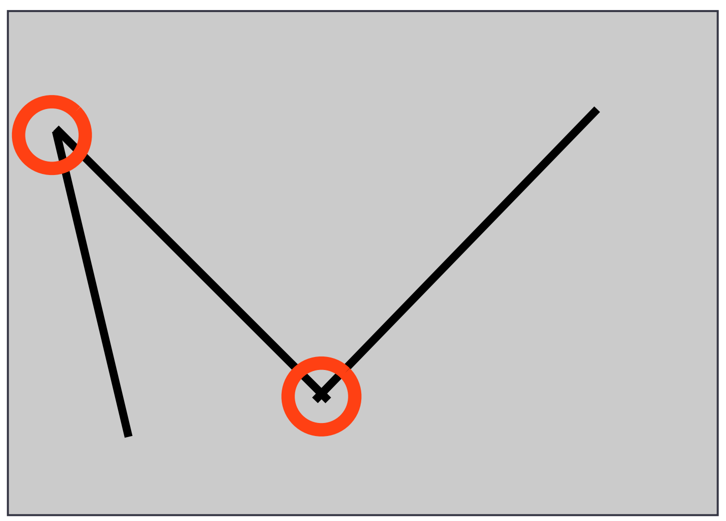
Corner Detection: The Basics



Slide motivated by Alyosha Efros

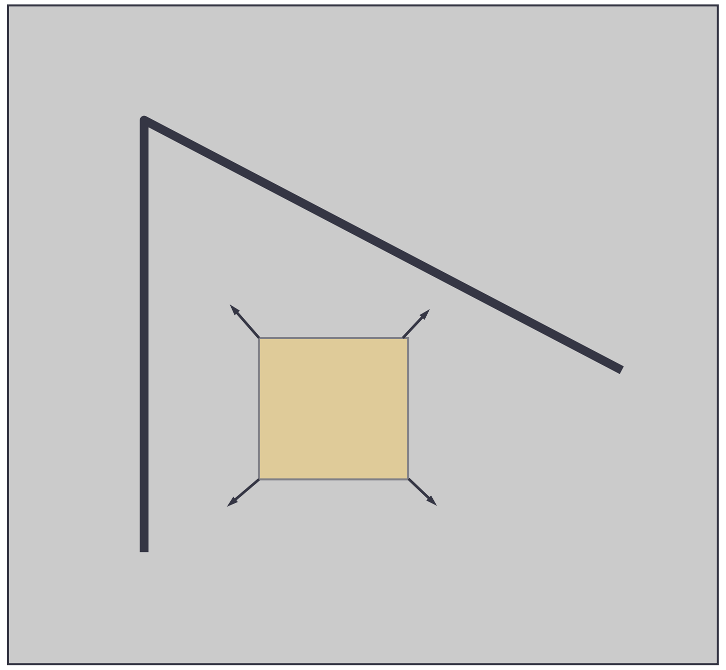


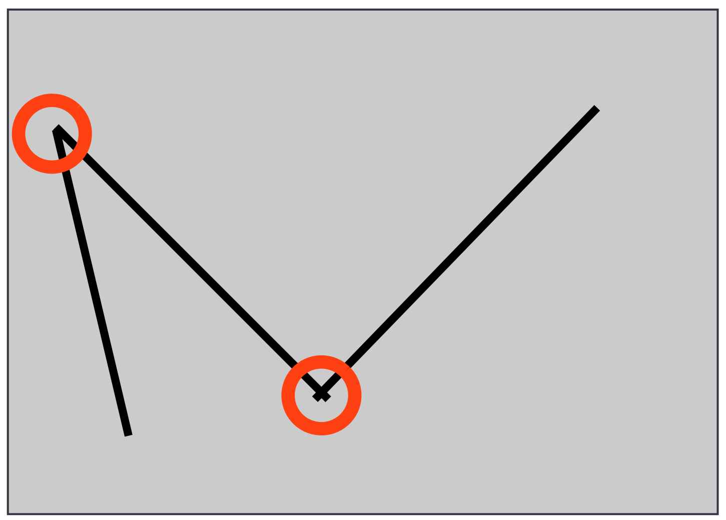
Corner Detection: The Basics



Slide motivated by Alyosha Efros

Corner Detection: The Basics

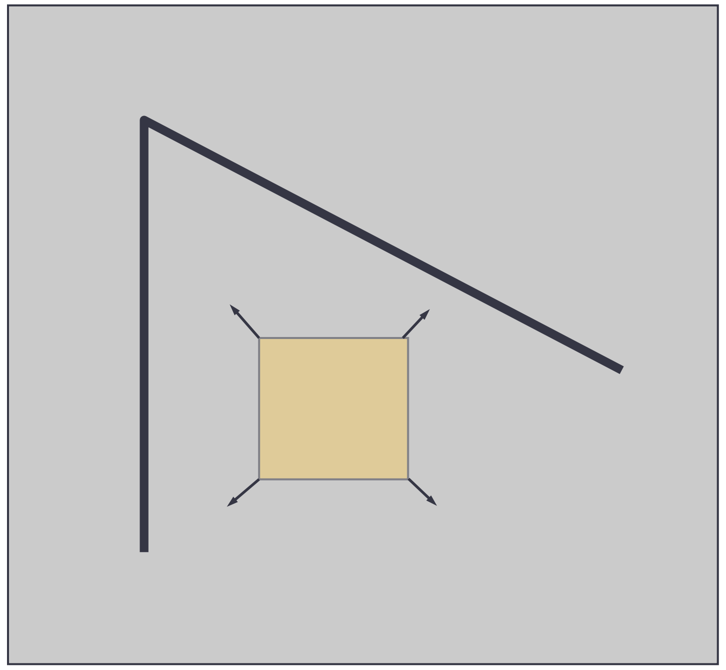




Slide motivated by Alyosha Efros

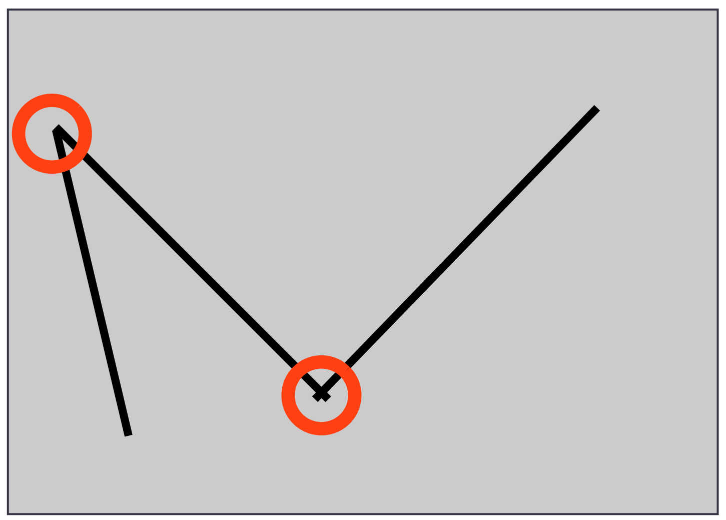
10

Corner Detection: The Basics

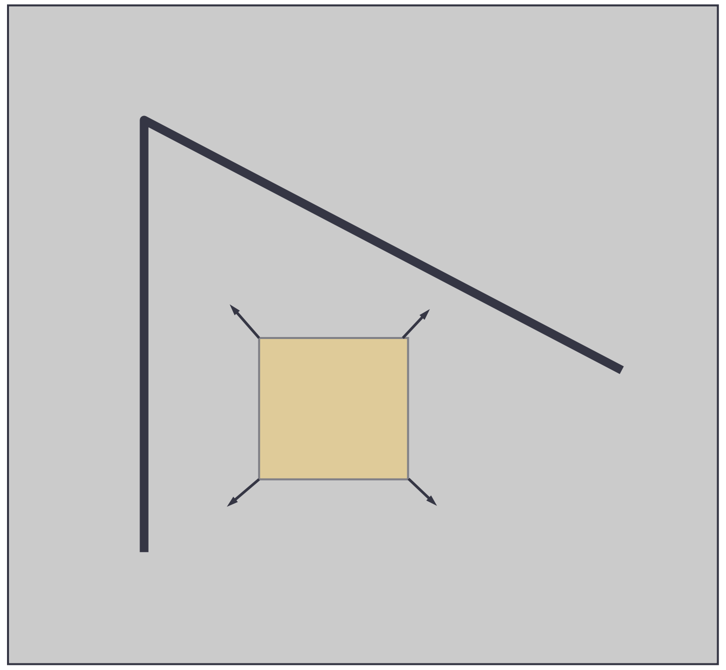


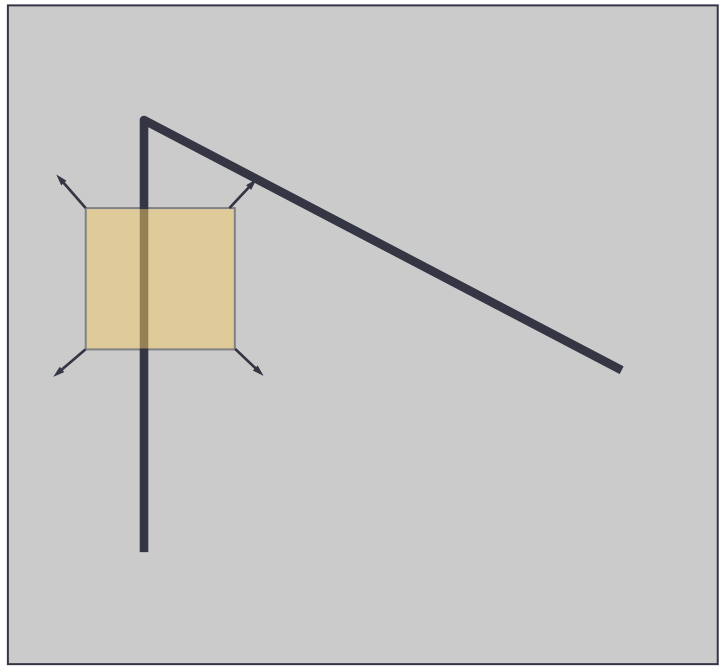
"flat"

No change in any directions



Slide motivated by Alyosha Efros

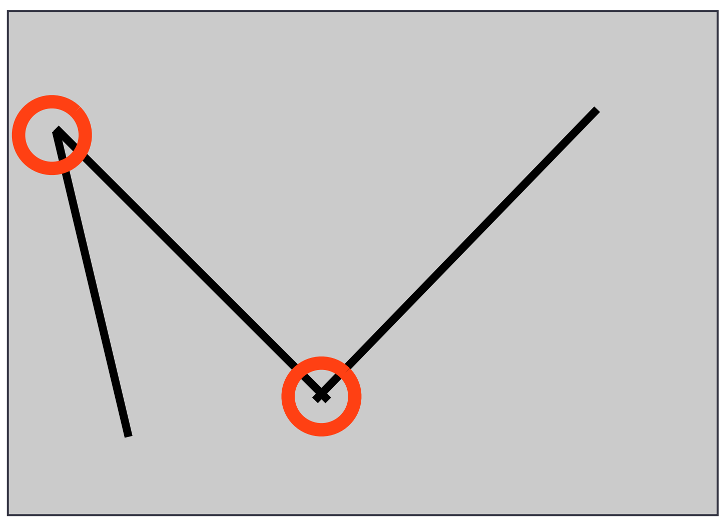




Corner Detection: The Basics

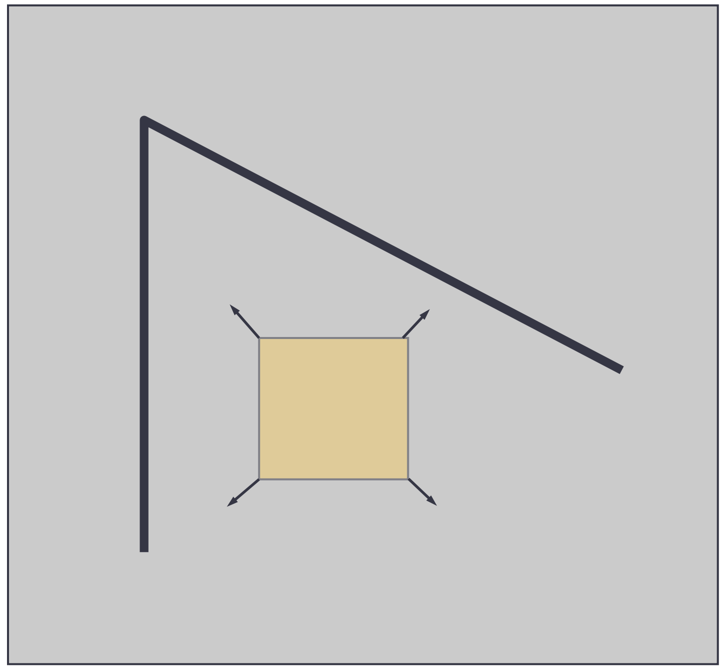
"flat"

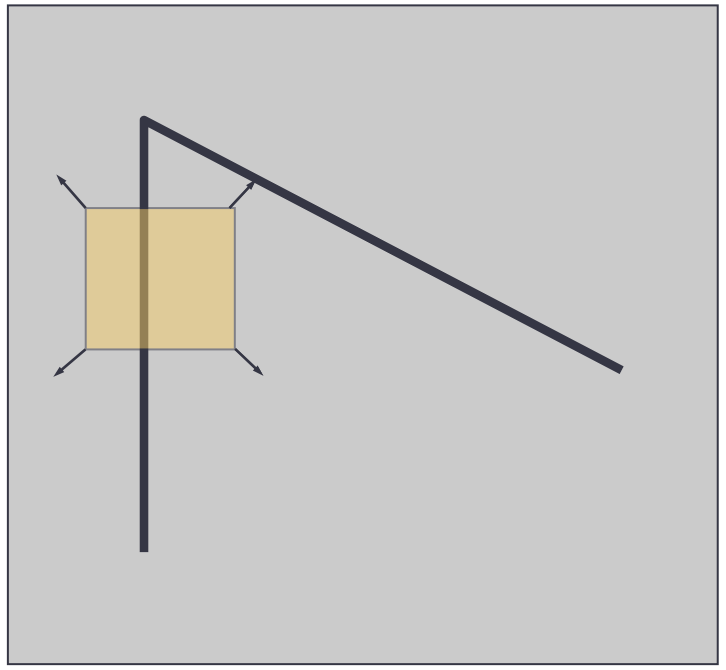
No change in any directions



Slide motivated by Alyosha Efros

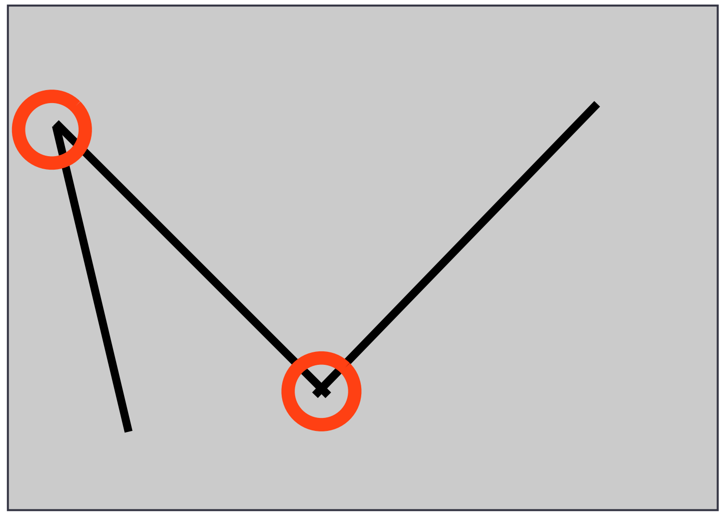
Corner Detection: The Basics





"flat"

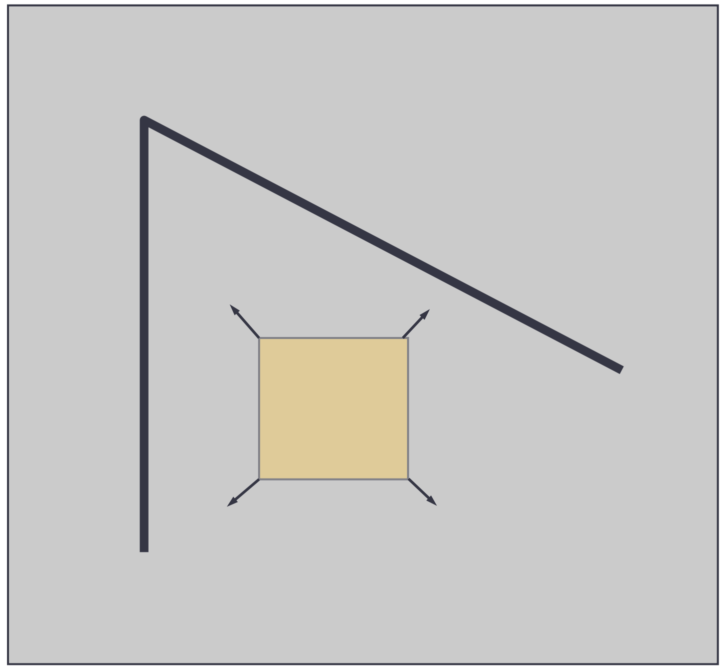
No change in any directions



Slide motivated by Alyosha Efros

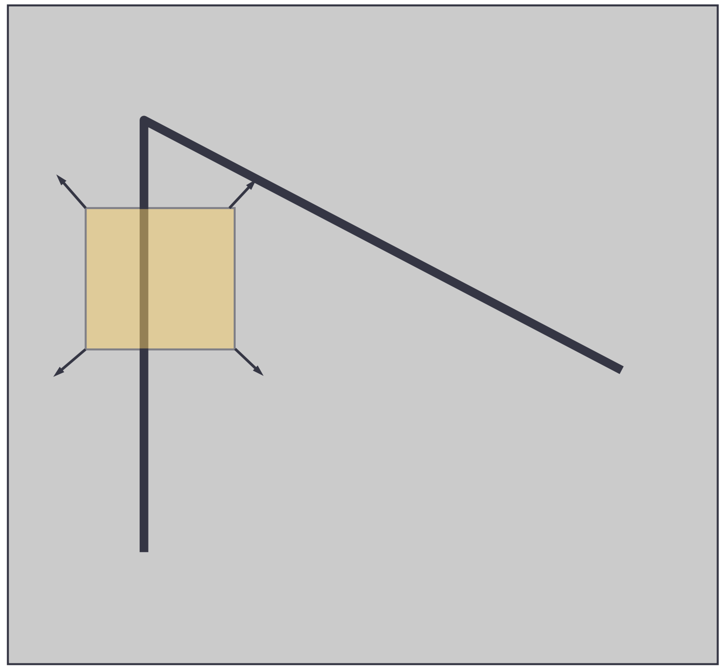
10

Corner Detection: The Basics



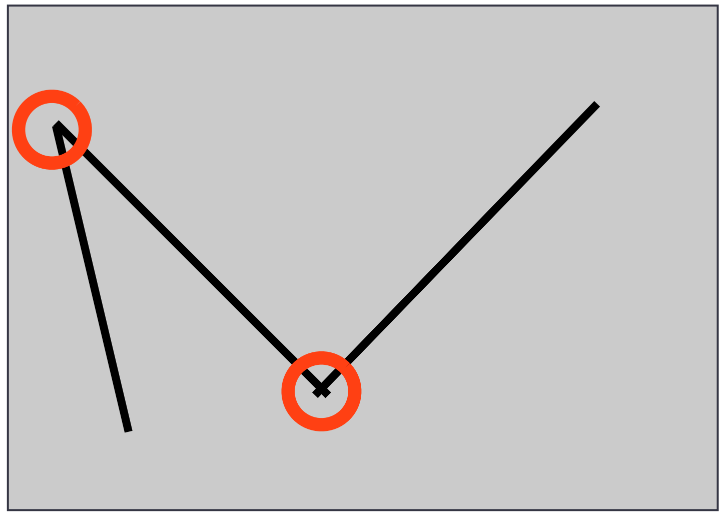
"flat"

No change in any directions



"edge"

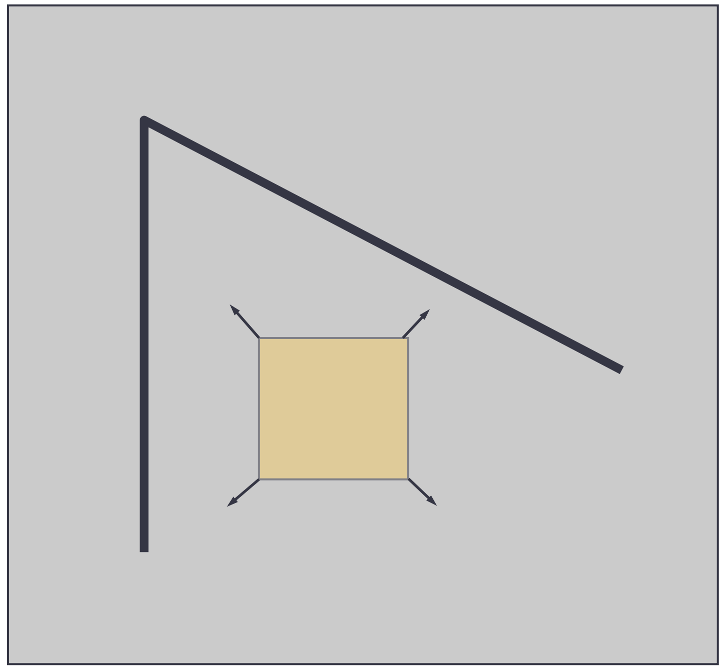
no change along the edge direction



Slide motivated by Alyosha Efros

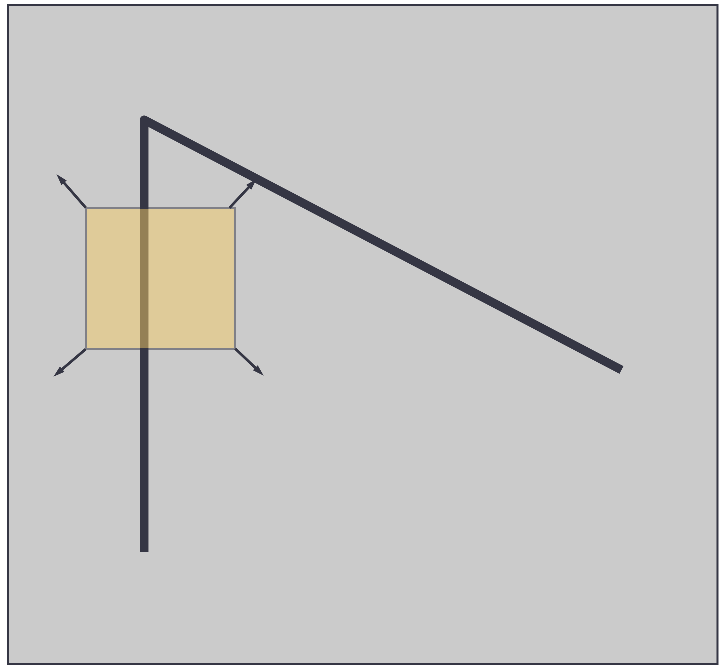
10

Corner Detection: The Basics



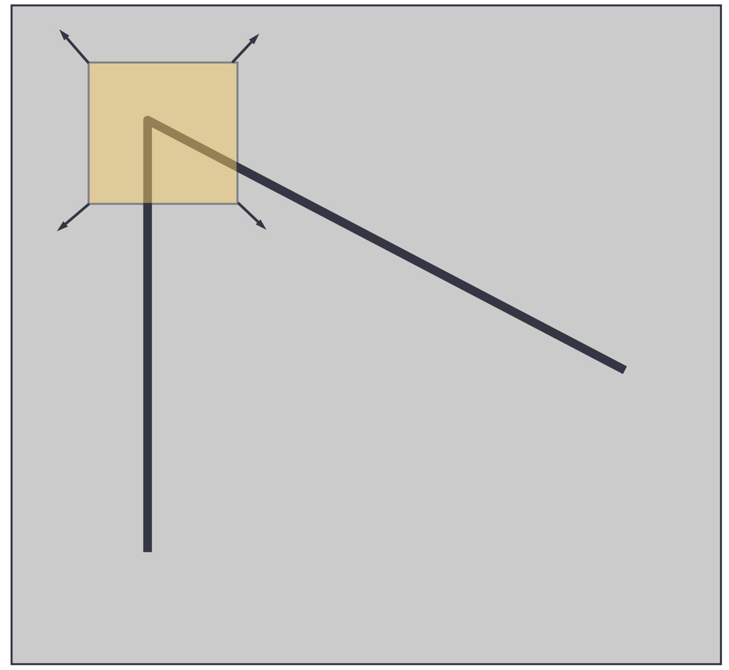
“flat”

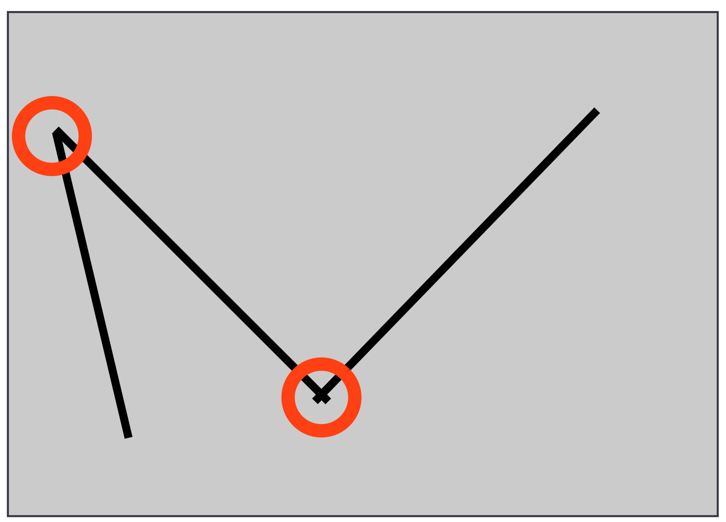
No change in any directions



“edge”

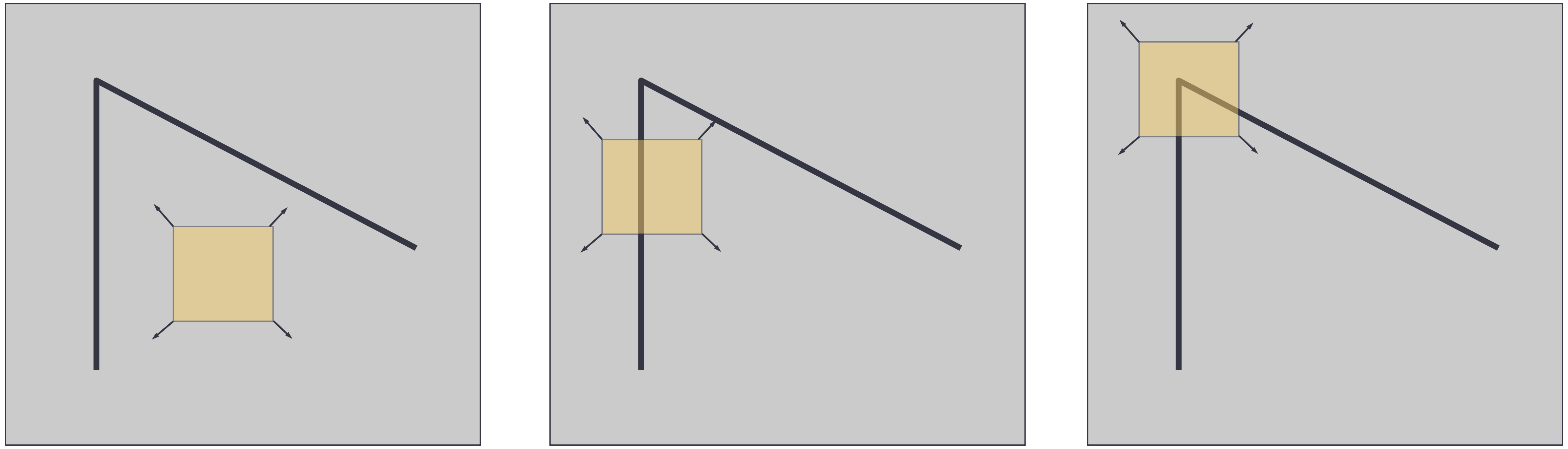
no change along the edge direction





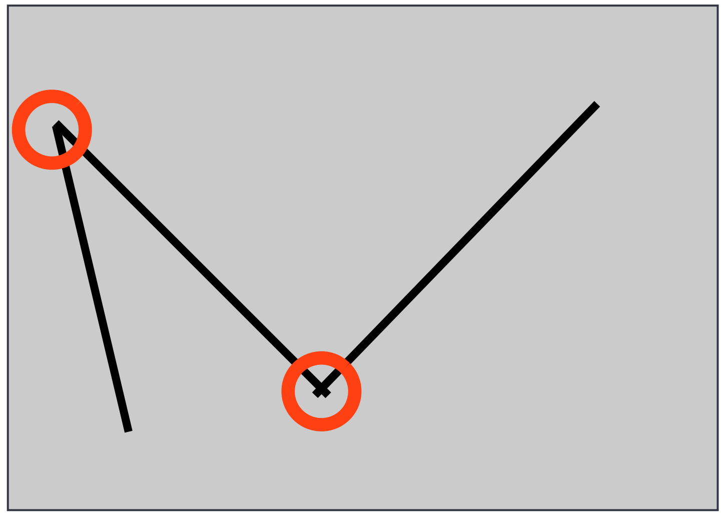
Slide motivated by Alyosha Efros

Corner Detection: The Basics



"flat"
No change in any directions

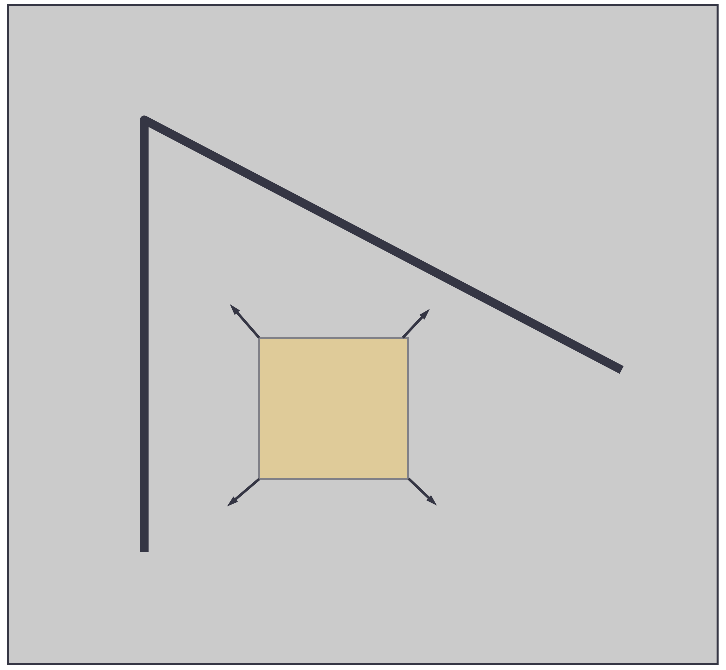
"edge"
no change along the edge direction



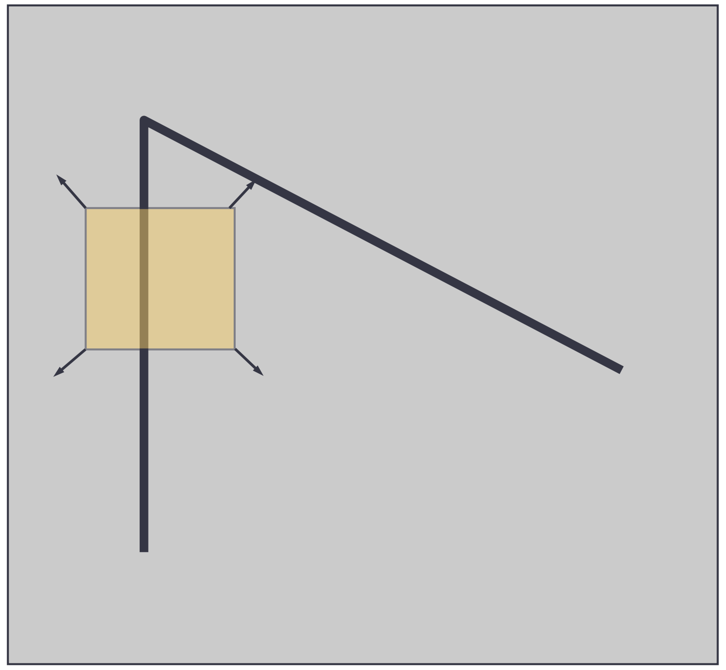
Slide motivated by Alyosha Efros

10

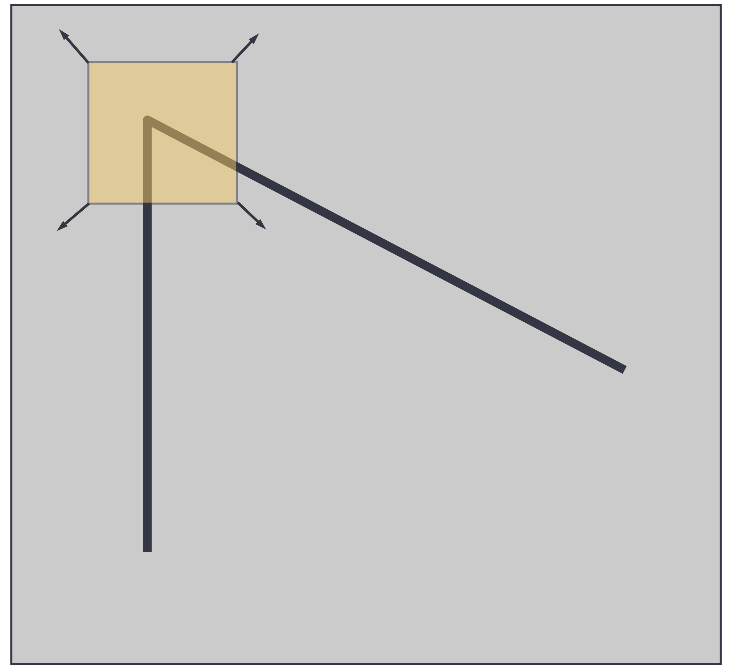
Corner Detection: The Basics



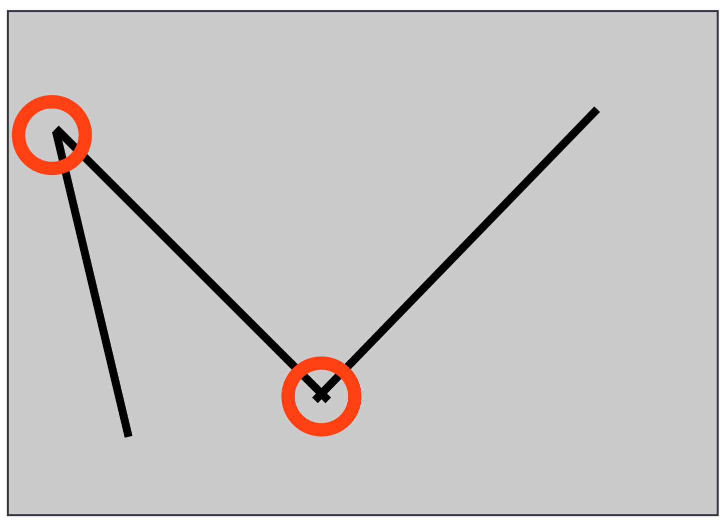
"flat"
No change in any directions



"edge"
no change along the edge direction

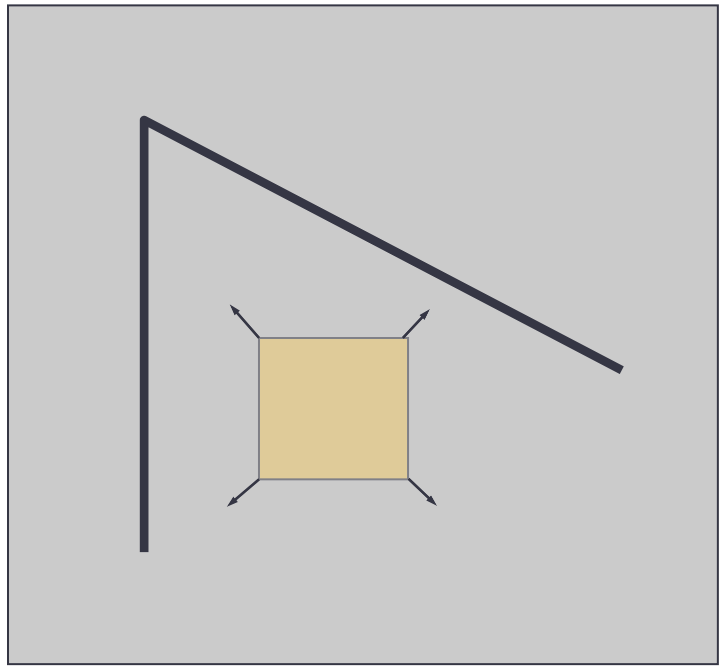
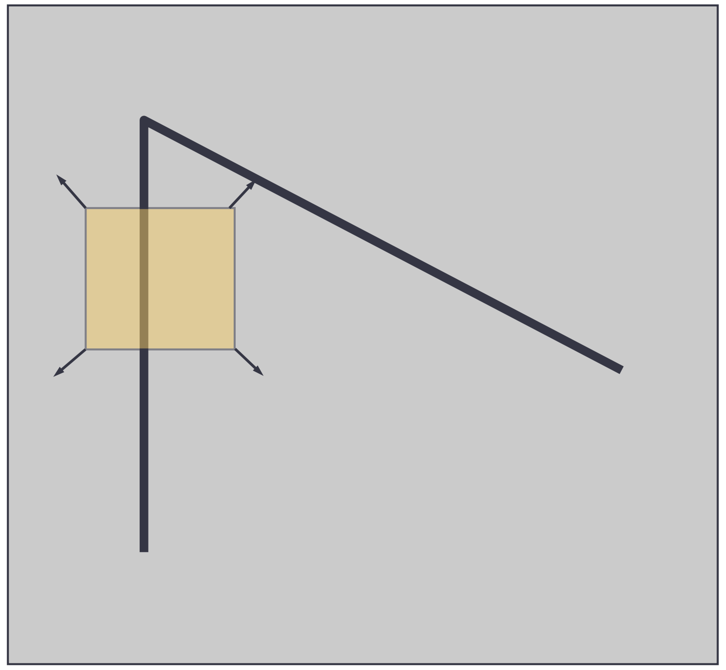
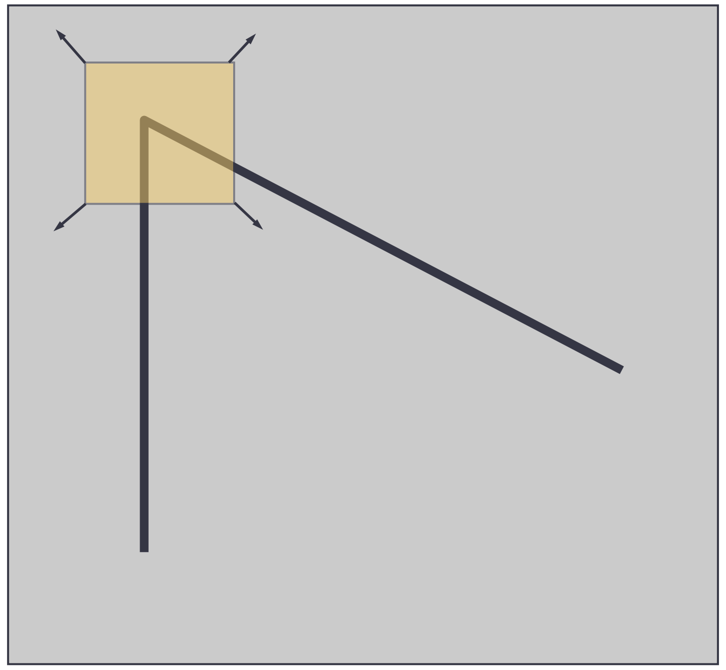


"corner"
change in all directions



Slide motivated by Alyosha Efros

Corner Detection: The Basics

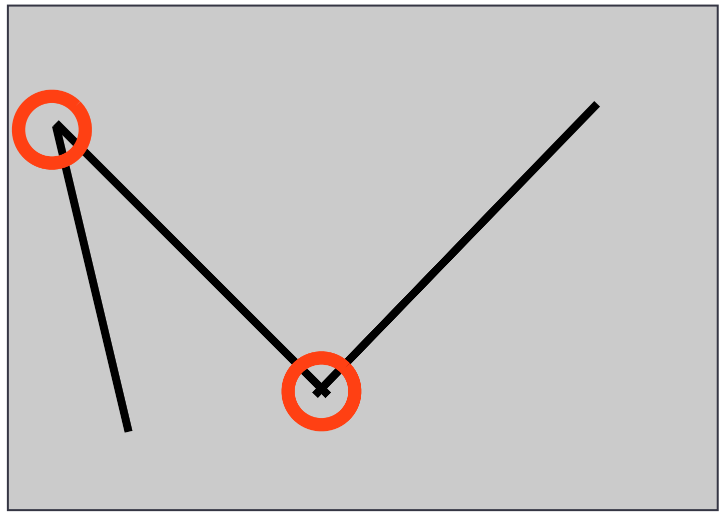




"flat"
No change in any directions

"edge"
no change along the edge direction

"corner"
change in all directions

- ★ Harris Corners (Harris and Stephens,1988)



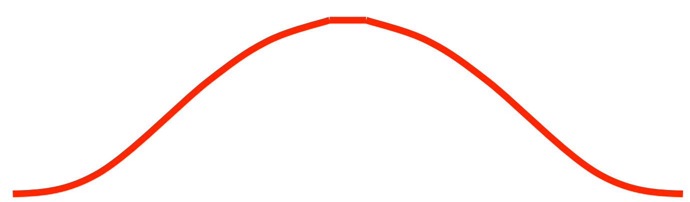
Slide motivated by Alyosha Efros

Corner Detection: Mathematics

Computation of the change in appearance by shifting the window by *u*,*v*:

$$E(u,v) = \sum_{x,y} w(x,y)\left[I(x+u, y+v) - I(x,y)\right]^2$$

$$w(x,y) \begin{cases} \text{box function} \\ \\ \text{a Gaussian} \end{cases}$$



1 inside, 0 outside

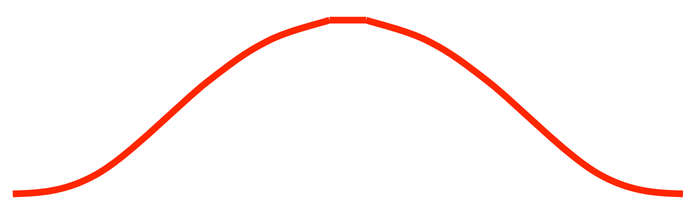
Corner Detection: Mathematics

Computation of the change in appearance by shifting the window by *u*,*v*:

$$E(u,v) = \sum_{x,y} w(x,y)\,[I(x+u,y+v) - I(x,y)]^2$$

intensity

$$w(x,y) \begin{cases} \text{box function} \\ \text{a Gaussian} \end{cases}$$



1 inside, 0 outside

Corner Detection: Mathematics

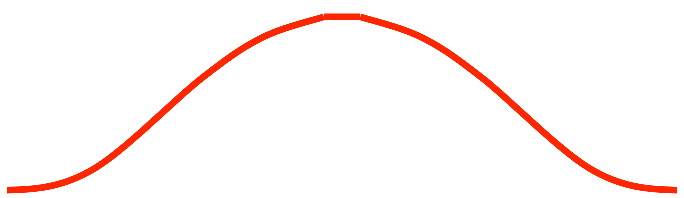
Computation of the change in appearance by shifting the window by *u*,*v*:

$$E(u,v) = \sum_{x,y} w(x,y)\,[\underbrace{I(x+u,y+v)}_{\text{shifted intensity}} - \underbrace{I(x,y)}_{\text{intensity}}]^2$$

shifted intensity

intensity

$$w(x,y) \begin{cases} \text{box function} \\ \text{a Gaussian} \end{cases}$$



1 inside, 0 outside

Corner Detection: Mathematics

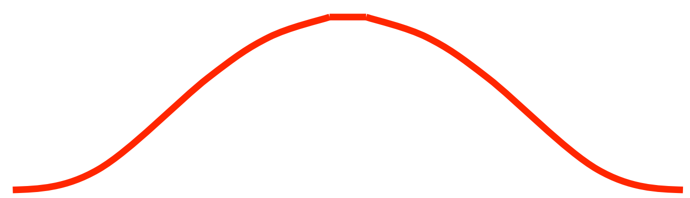
Computation of the change in appearance by shifting the window by *u,v*:

$$E(u,v) = \sum_{x,y} \underbrace{w(x,y)}_{\text{window function}} \, [\underbrace{I(x+u,y+v)}_{\text{shifted intensity}} - \underbrace{I(x,y)}_{\text{intensity}}]^2$$

$w(x,y)$ $\begin{cases} \text{box function} \\ \text{a Gaussian} \end{cases}$

box function

a Gaussian



1 inside, 0 outside

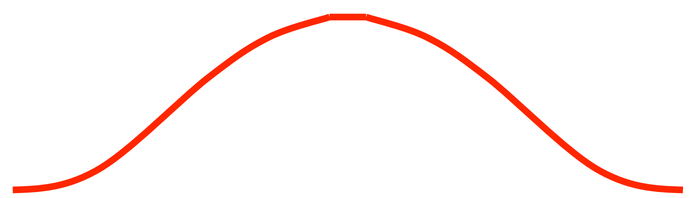
Corner Detection: Mathematics

Computation of the change in appearance by shifting the window by *u,v*:

$$E(u,v) = \sum_{x,y} w(x,y)\,[I(x+u,y+v) - I(x,y)]^2$$

change in appearance — window function — shifted intensity — intensity

$$w(x,y) \begin{cases} \text{box function} \\ \text{a Gaussian} \end{cases}$$



1 inside, 0 outside

$$E(u,v) = \sum_{x,y} w(x,y)\left[I(x+u,y+v) - I(x,y)\right]^2$$

Corner Detection: Mathematics

Corner Detection: Mathematics

$$E(u,v) = \sum_{x,y} w(x,y)\left[I(x+u,y+v) - I(x,y)\right]^2$$

The quadratic approximation, following Taylor Expansion, simplifies to:

Corner Detection: Mathematics

$$E(u,v) = \sum_{x,y} w(x,y)\left[I(x+u,y+v) - I(x,y)\right]^2$$

The quadratic approximation, following Taylor Expansion, simplifies to:

$$E(u,v) \approx \begin{bmatrix} u & v \end{bmatrix} M \begin{bmatrix} u \\ v \end{bmatrix}$$

Corner Detection: Mathematics

$$E(u,v) = \sum_{x,y} w(x,y)\left[I(x+u, y+v) - I(x,y)\right]^2$$

The quadratic approximation, following Taylor Expansion, simplifies to:

$$E(u,v) \approx \begin{bmatrix} u & v \end{bmatrix} M \begin{bmatrix} u \\ v \end{bmatrix}$$

where M is a *second moment matrix* computed from image derivatives I_x and I_y:

Corner Detection: Mathematics

$$E(u,v) = \sum_{x,y} w(x,y)\left[I(x+u,y+v) - I(x,y)\right]^2$$

The quadratic approximation, following Taylor Expansion, simplifies to:

$$E(u,v) \approx \left[\begin{array}{cc} u & v \end{array}\right] M \left[\begin{array}{c} u \\ v \end{array}\right]$$

where *M* is a *second moment matrix* computed from image derivatives I_x and I_y:

$$M = \sum_{x,y} \omega(x,y) \left[\begin{array}{cc} I_x^2 & I_x I y \\ I_x I y & I^2 y \end{array}\right]$$

Corner Detection: Mathematics

$$E(u,v) = \sum_{x,y} w(x,y) \left[I(x+u, y+v) - I(x,y) \right]^2$$

The quadratic approximation, following Taylor Expansion, simplifies to:

$$E(u,v) \approx \left[\begin{array}{cc} u & v \end{array} \right] M \left[\begin{array}{c} u \\ v \end{array} \right]$$

where *M* is a *second moment matrix* computed from image derivatives I_x and I_y:

$$M = \sum_{x,y} \omega(x,y) \left[\begin{array}{cc} I_x^2 & I_x I y \\ I_x I y & I^2 y \end{array} \right]$$

$$E(u,v) = \sum_{x,y} \omega(x,y) \left(u^2 I_x^2 + 2uv I_x I_y + v^2 I_y^2 \right)$$

Corner Detection: Mathematics

$$E(u,v) = \sum_{x,y} w(x,y)\left[I(x+u,y+v) - I(x,y)\right]^2$$

The quadratic approximation, following Taylor Expansion, simplifies to:

$$E(u,v) \approx \begin{bmatrix} u & v \end{bmatrix} M \begin{bmatrix} u \\ v \end{bmatrix}$$

where *M* is a *second moment matrix* computed from image derivatives I_x and I_y:

$$M = \sum_{x,y} \omega(x,y) \begin{bmatrix} I_x^2 & I_x Iy \\ I_x Iy & I^2 y \end{bmatrix}$$

$$E(u,v) = \sum_{x,y} \omega(x,y) \left(u^2 I_x^2 + 2uvI_xI_y + v^2 I_y^2\right)$$

The surface *E*(*u*,*v*) is locally approximated by a quadratic form.

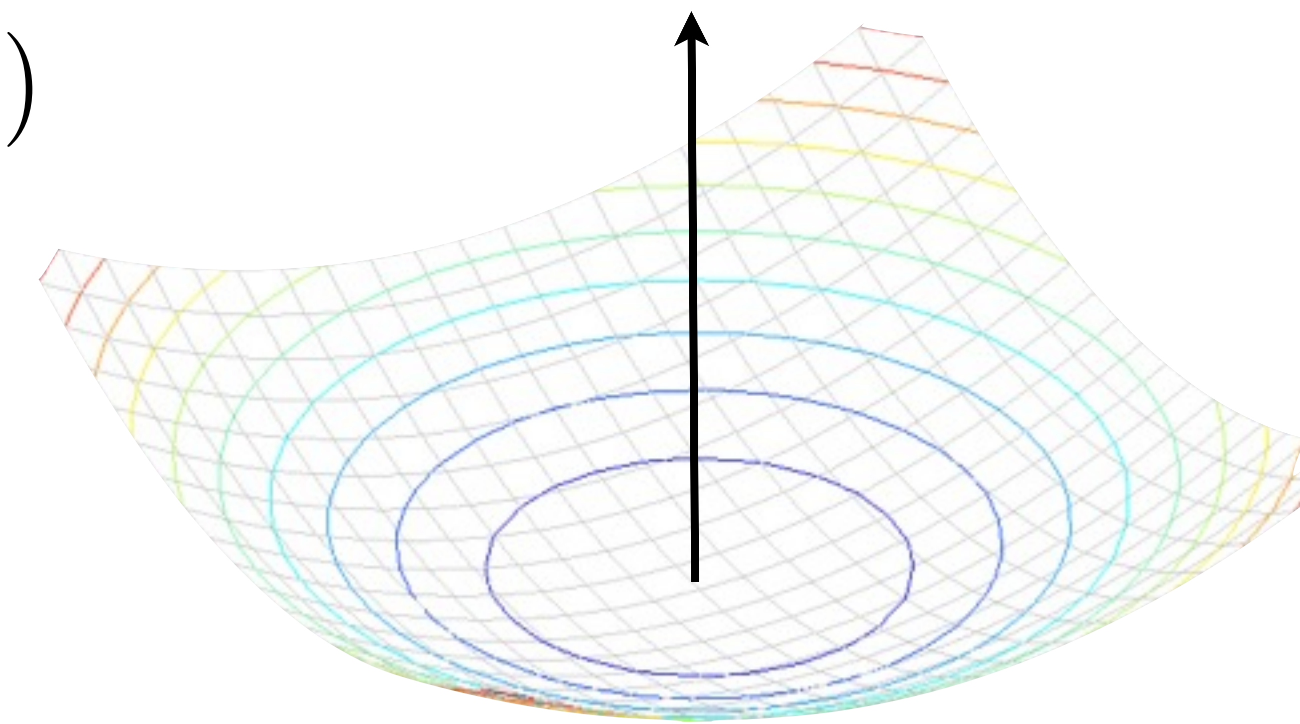
The surface *E*(*u*,*v*) is locally approximated by a quadratic form.

$$E(u,v) = \sum_{x,y} \omega(x,y) \left(u^2 I_x^2 + 2uvI_xI_y + v^2 I_y^2\right)$$

Slide motivated by Aaron Bobick

The surface *E*(*u*,*v*) is locally approximated by a quadratic form.

$$E(u,v) = \sum_{x,y} \omega(x,y) \left(u^2 I_x^2 + 2uvI_xI_y + v^2 I_y^2\right)$$

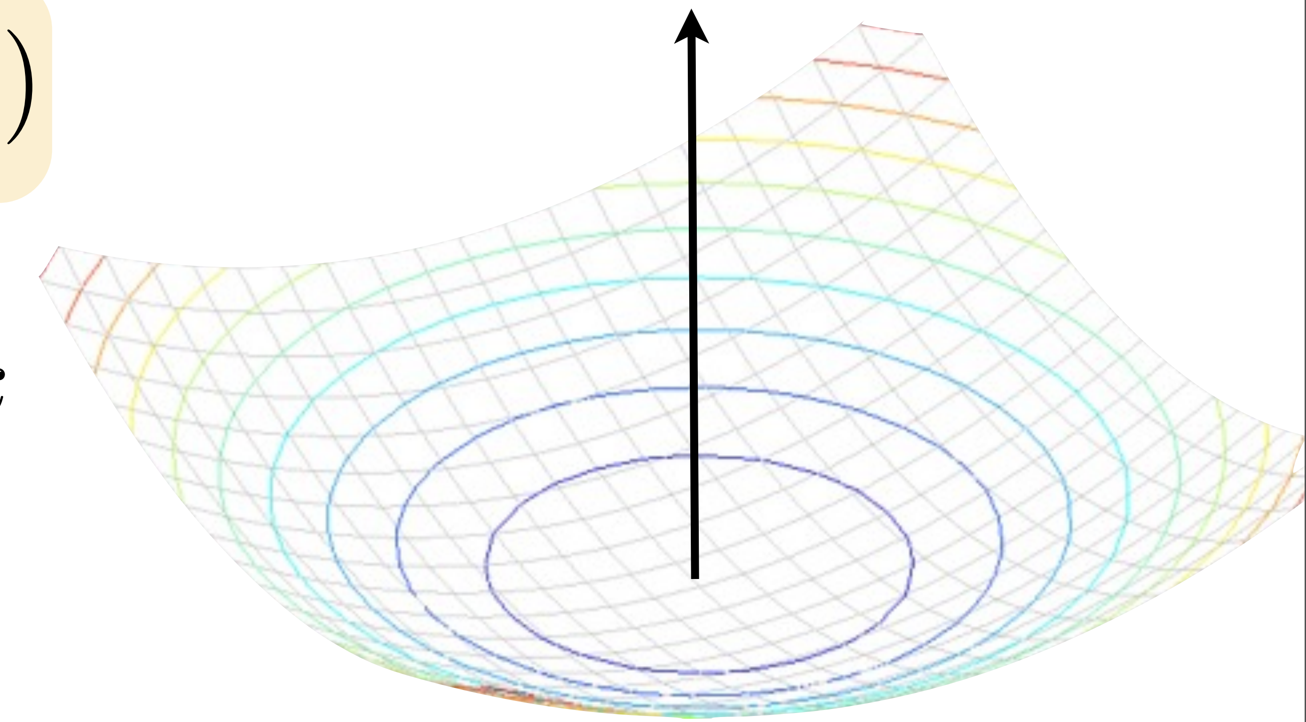


Slide motivated by Aaron Bobick

The surface $E(u,v)$ is locally approximated by a quadratic form.

$$E(u,v) = \sum_{x,y} \omega(x,y) \left(u^2 I_x^2 + 2uvI_xI_y + v^2 I_y^2\right)$$

consider a "slice": $u^2 I_x^2 + 2uvI_xI_y + v^2 I_y^2 = k$



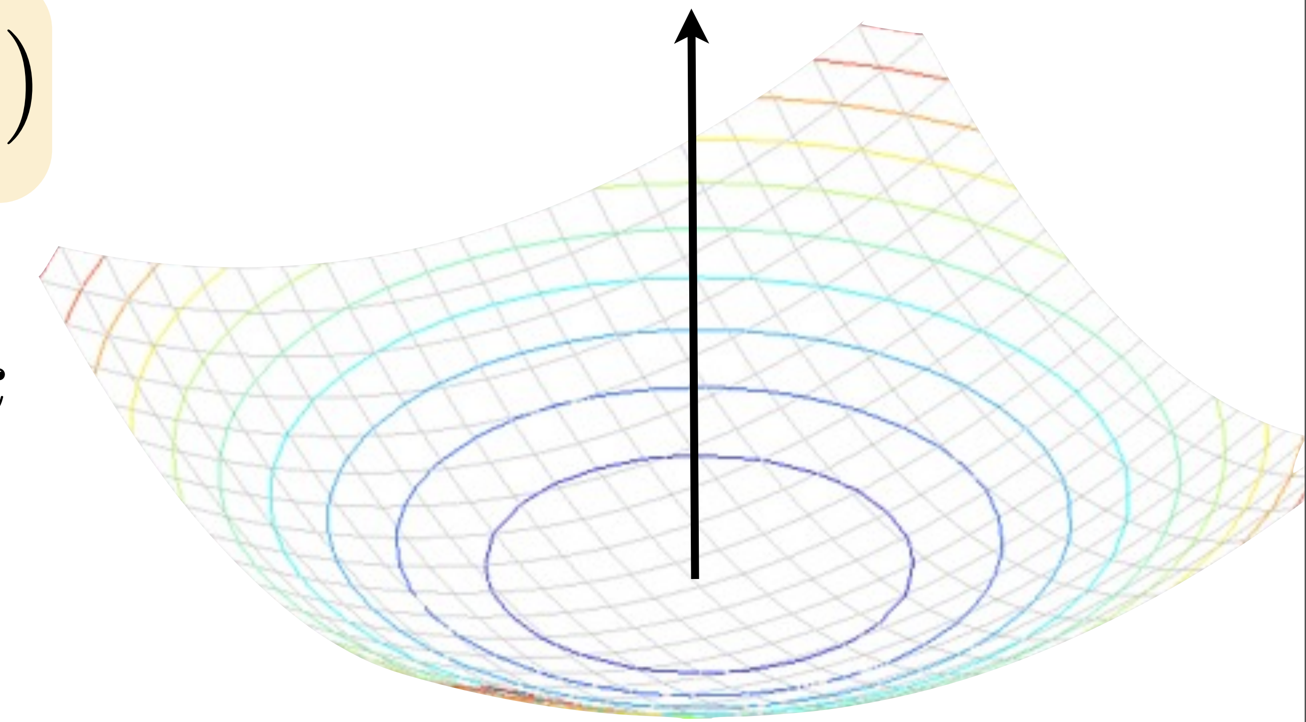
Slide motivated by Aaron Bobick

The surface $E(u,v)$ is locally approximated by a quadratic form.

$$E(u,v) = \sum_{x,y} \omega(x,y) \left(u^2 I_x^2 + 2uvI_xI_y + v^2 I_y^2\right)$$

consider a "slice": $u^2 I_x^2 + 2uvI_xI_y + v^2 I_y^2 = k$

which is an equation of an ellipse.



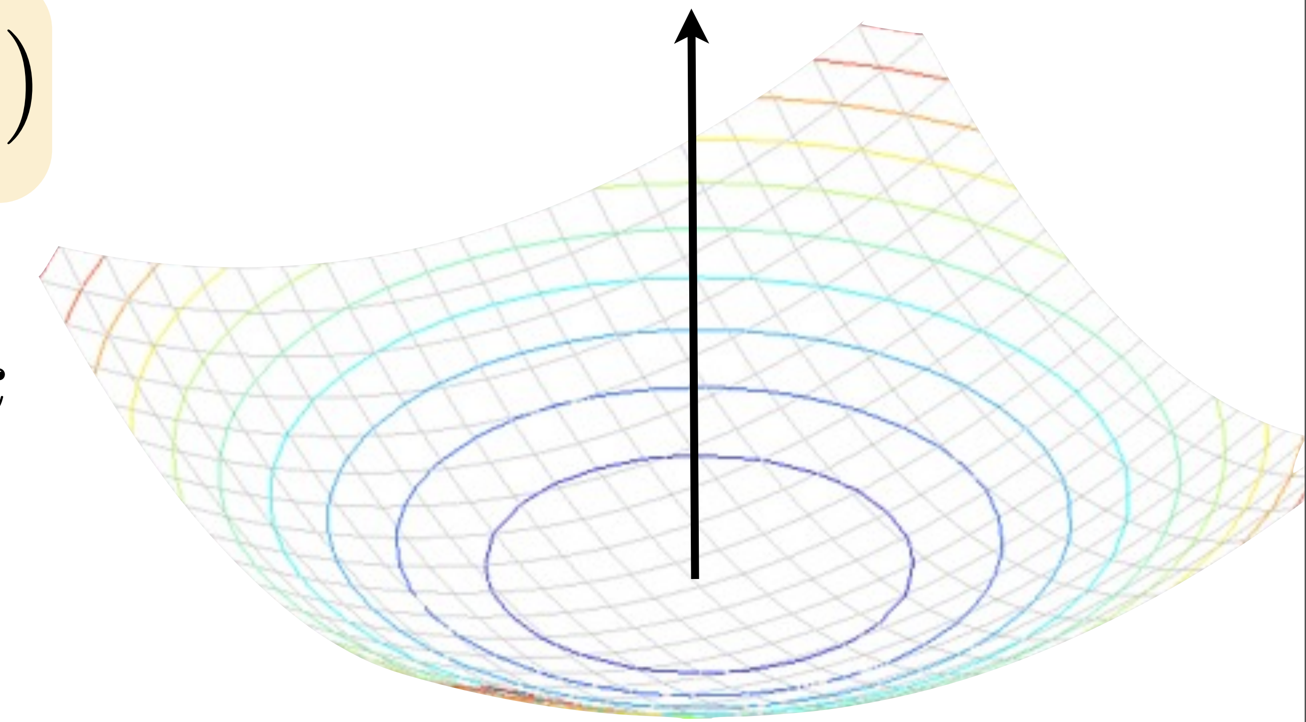
Slide motivated by Aaron Bobick

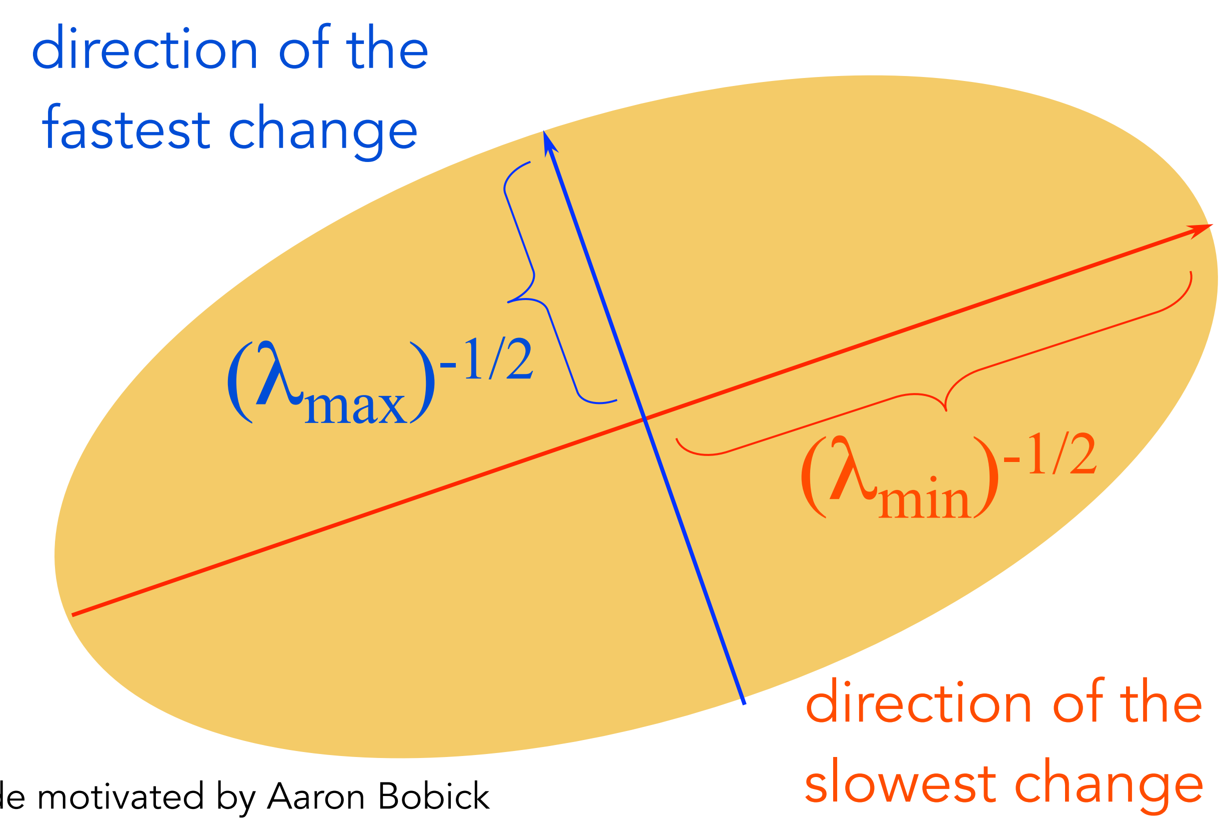
The surface $E(u,v)$ is locally approximated by a quadratic form.

$$E(u,v) = \sum_{x,y} \omega(x,y) \left(u^2 I_x^2 + 2uvI_xI_y + v^2 I_y^2\right)$$

consider a "slice": $u^2 I_x^2 + 2uvI_xI_y + v^2 I_y^2 = k$

which is an equation of an ellipse.

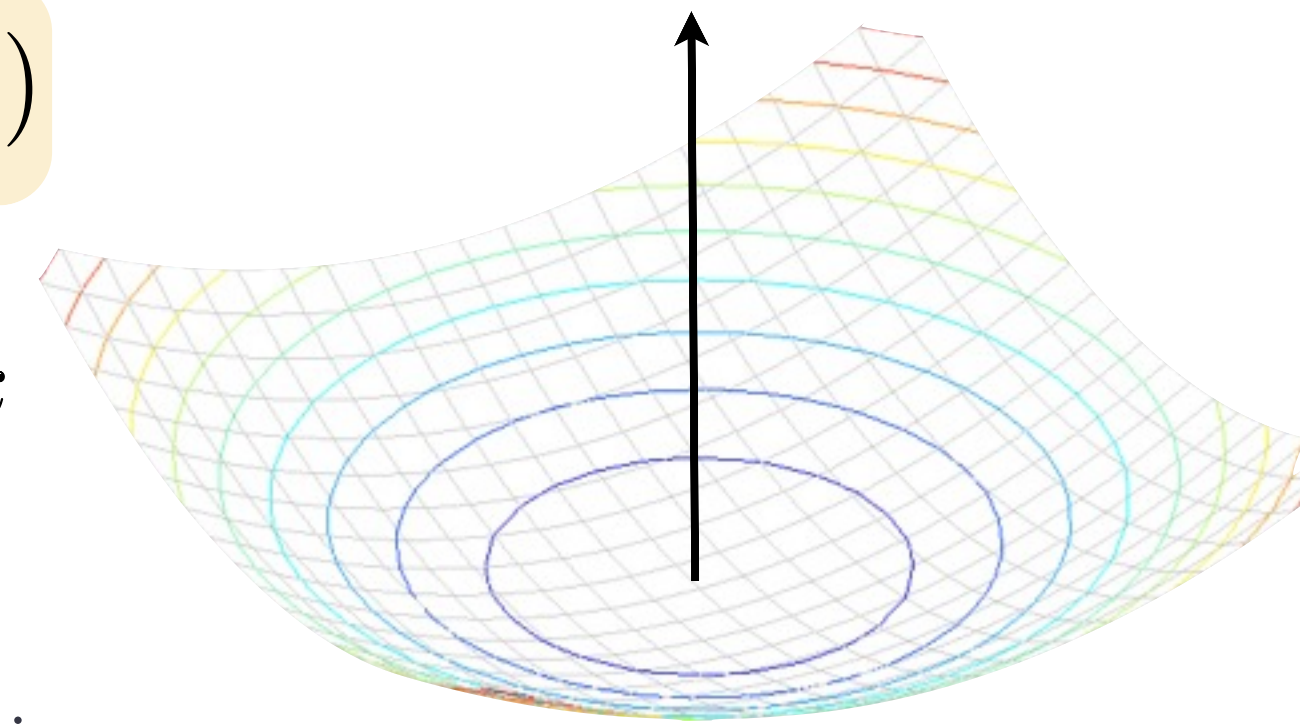




Slide motivated by Aaron Bobick

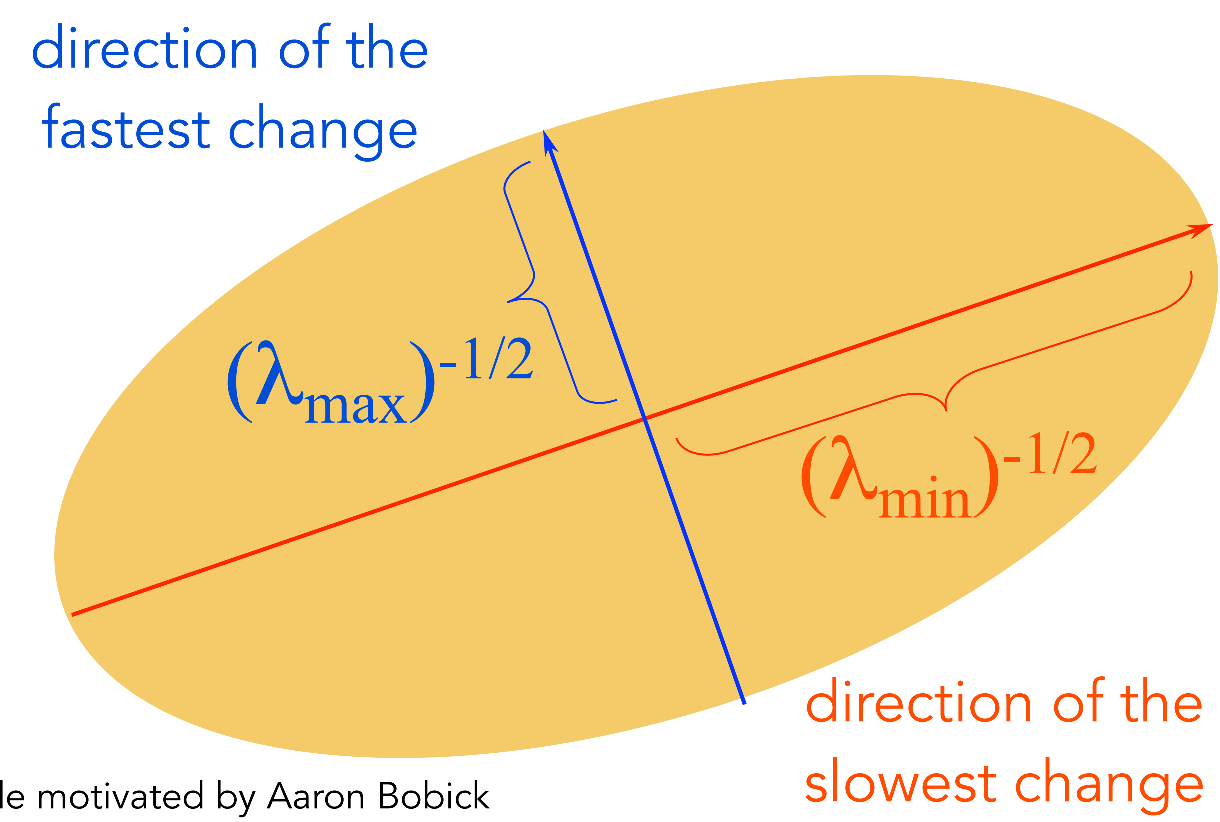
The surface *E*(*u*,*v*) is locally approximated by a quadratic form.

$$E(u,v) = \sum_{x,y} \omega(x,y) \left(u^2 I_x^2 + 2uvI_xI_y + v^2 I_y^2\right)$$



consider a "slice": $u^2 I_x^2 + 2uvI_xI_y + v^2 I_y^2 = k$

which is an equation of an ellipse.



Eigenvalue Analysis:

The axis lengths of the ellipse are determined by the eigenvalues and the orientation is determined by *eigenvectors*

Slide motivated by Aaron Bobick

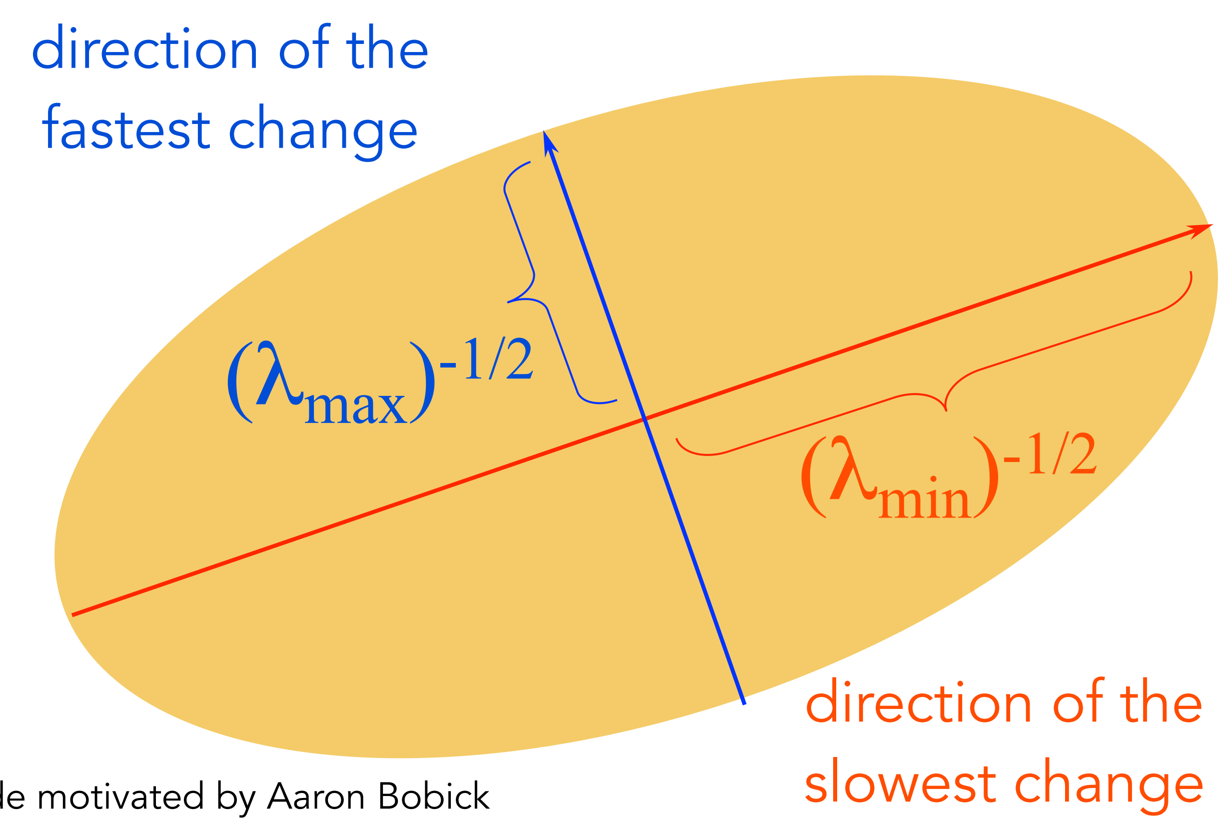
Consider a "slice": $u^2 I_x^2 + 2uvI_xI_y + v^2 I_y^2 = k$

which is an equation of an ellipse.

$$\begin{bmatrix} u & v \end{bmatrix} M \begin{bmatrix} u \\ v \end{bmatrix} = k$$

$$M = R^{-1} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} R$$

M is a diagonal matrix



Eigenvalue Analysis:

The axis lengths of the ellipse are determined by the eigenvalues and the orientation is determined by *eigenvectors, R.*

Slide motivated by Aaron Bobick

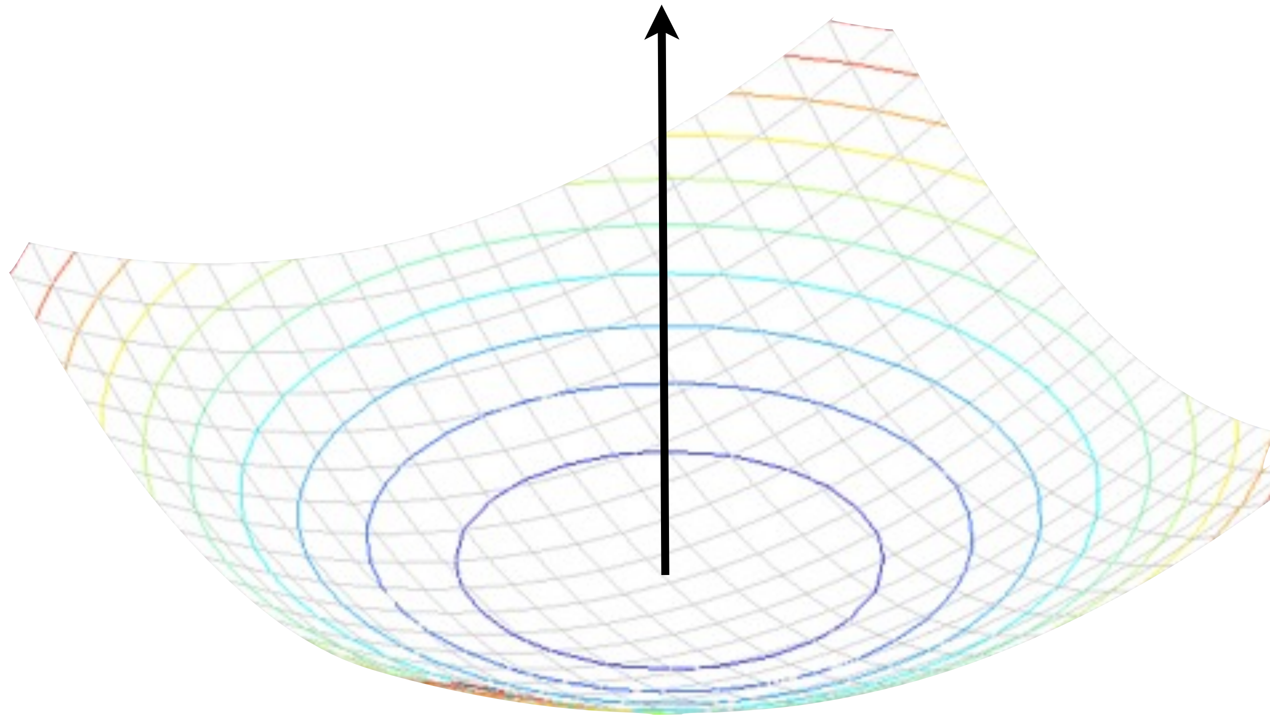
Consider a "slice": $u^2 I_x^2 + 2uvI_xI_y + v^2 I_y^2 = k$

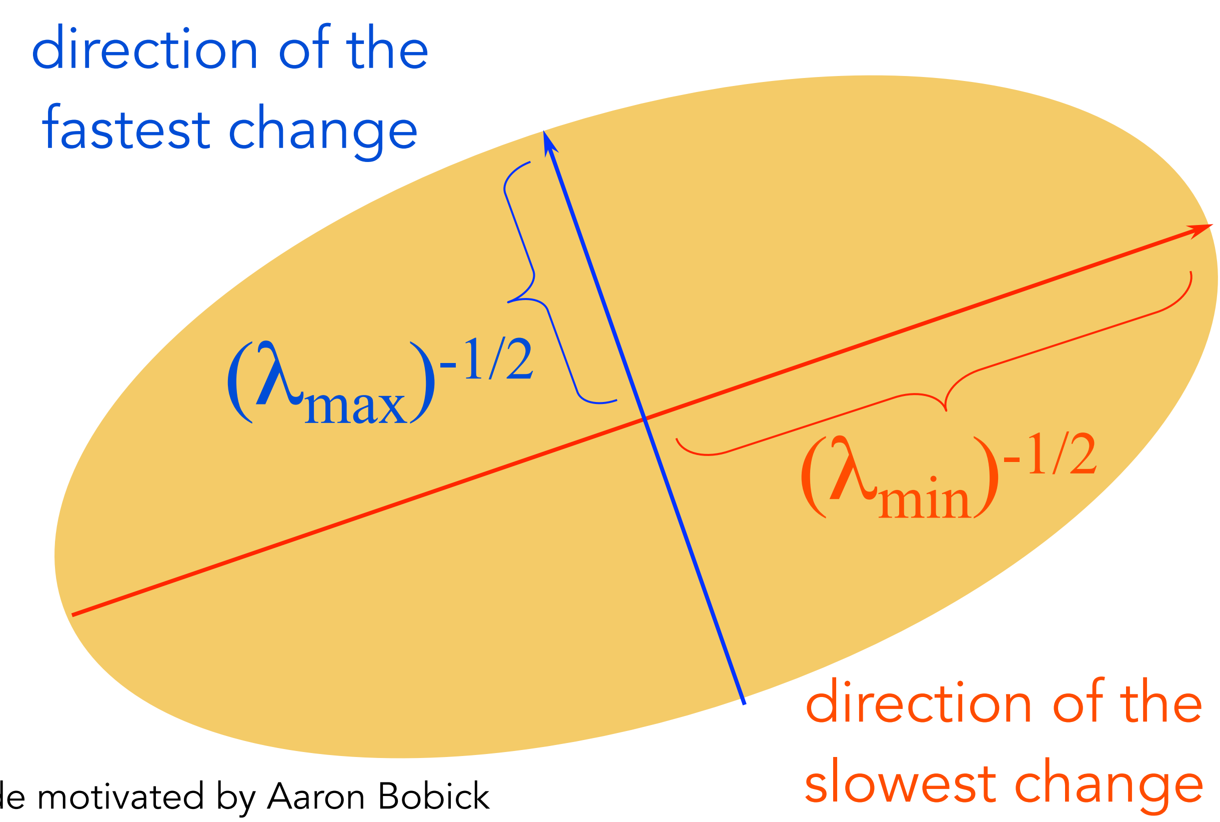
which is an equation of an ellipse.

$$\begin{bmatrix} u & v \end{bmatrix} M \begin{bmatrix} u \\ v \end{bmatrix} = k$$

$$M = R^{-1} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} R$$

M is a diagonal matrix





Eigenvalue Analysis:

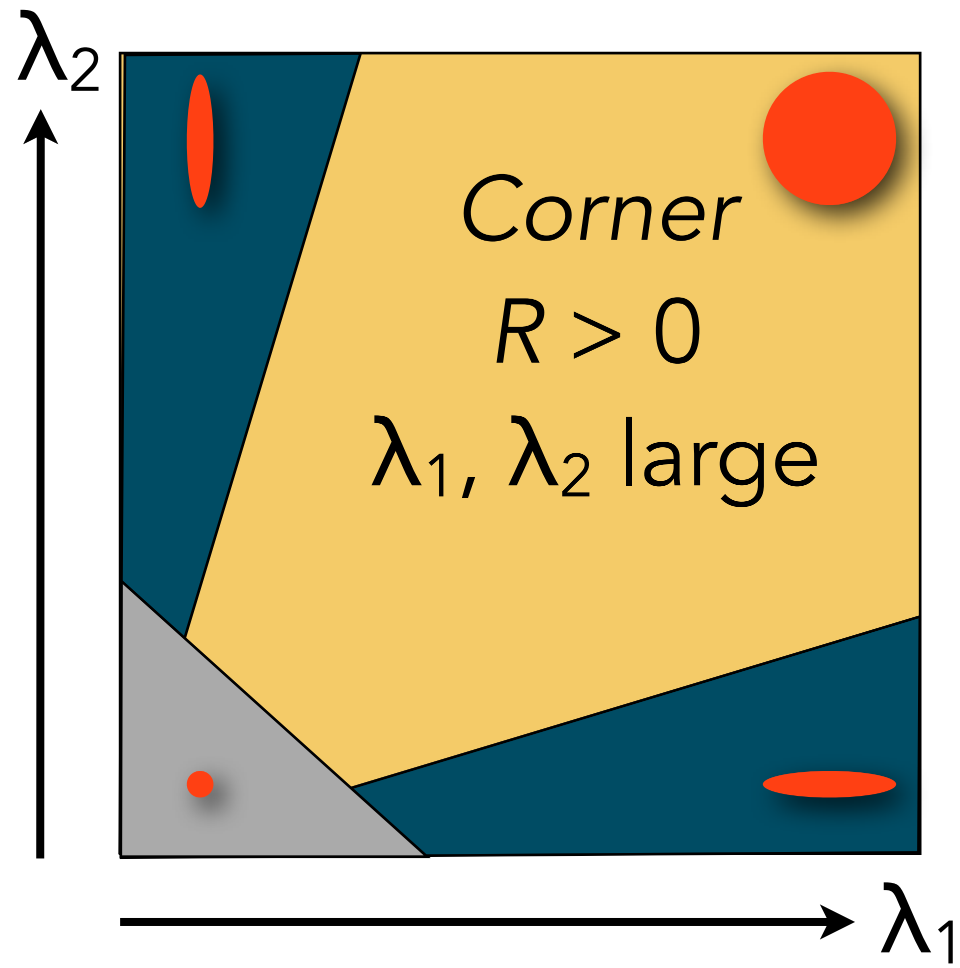
The axis lengths of the ellipse are determined by the eigenvalues and the orientation is determined by *eigenvectors, R.*

Slide motivated by Aaron Bobick

Eigenvalues

$$R = \det(M) - \alpha \operatorname{trace}(M)^2 = \lambda_1\lambda_2 - \alpha(\lambda_1 + \lambda_2)^2$$

α: constant (0.04 to 0.06)

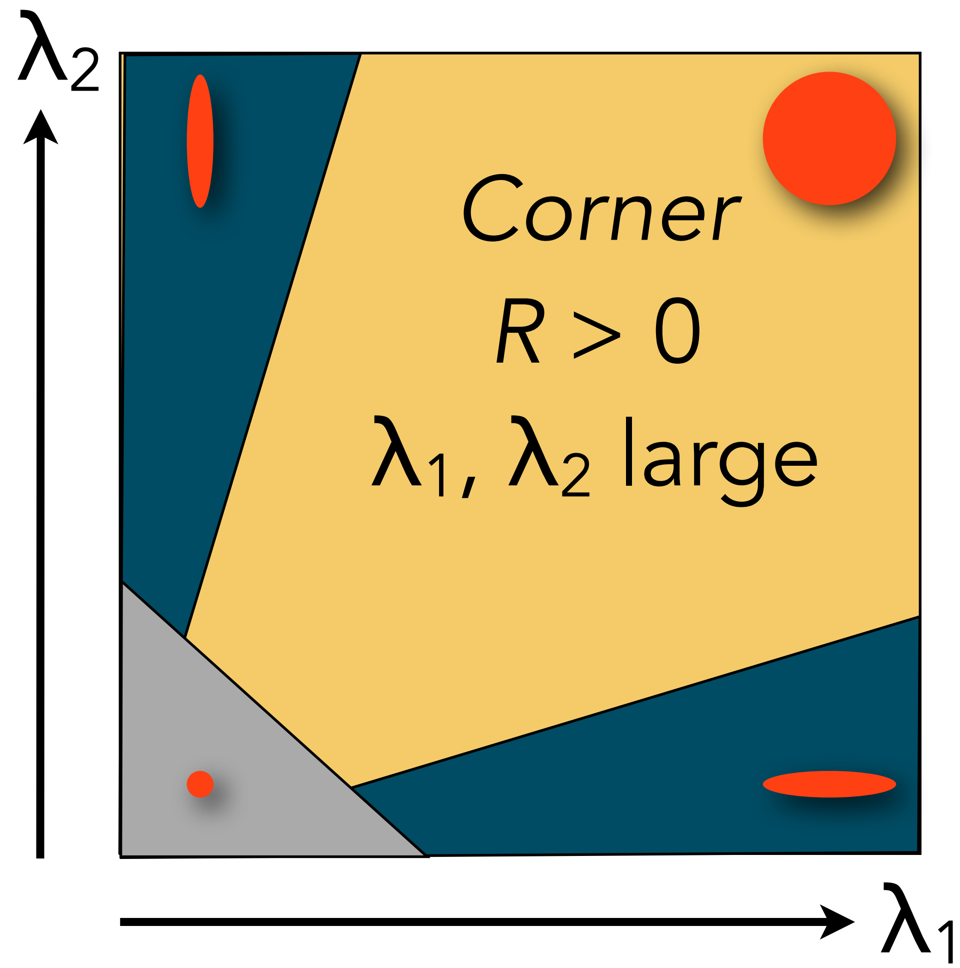


Eigenvalues

$$R = \det(M) - \alpha \ \mathrm{trace}(M)^2 = \lambda_1\lambda_2 - \alpha(\lambda_1 + \lambda_2)^2$$

α: constant (0.04 to 0.06)

★ R depends only on eigenvalues of M

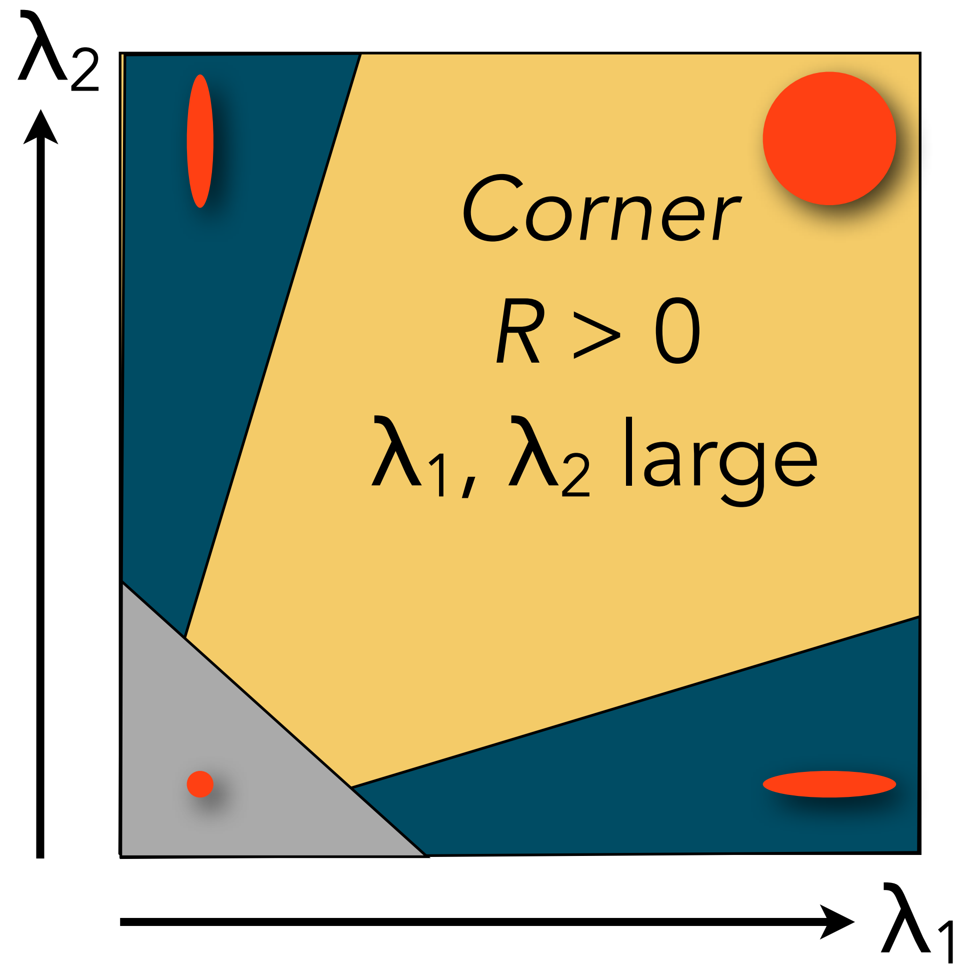


Eigenvalues

$$R = \det(M) - \alpha\ \mathrm{trace}(M)^2 = \lambda_1\lambda_2 - \alpha(\lambda_1 + \lambda_2)^2$$

α: constant (0.04 to 0.06)

- ★ *R* depends only on eigenvalues of *M*
- ★ *R* is large for a corner

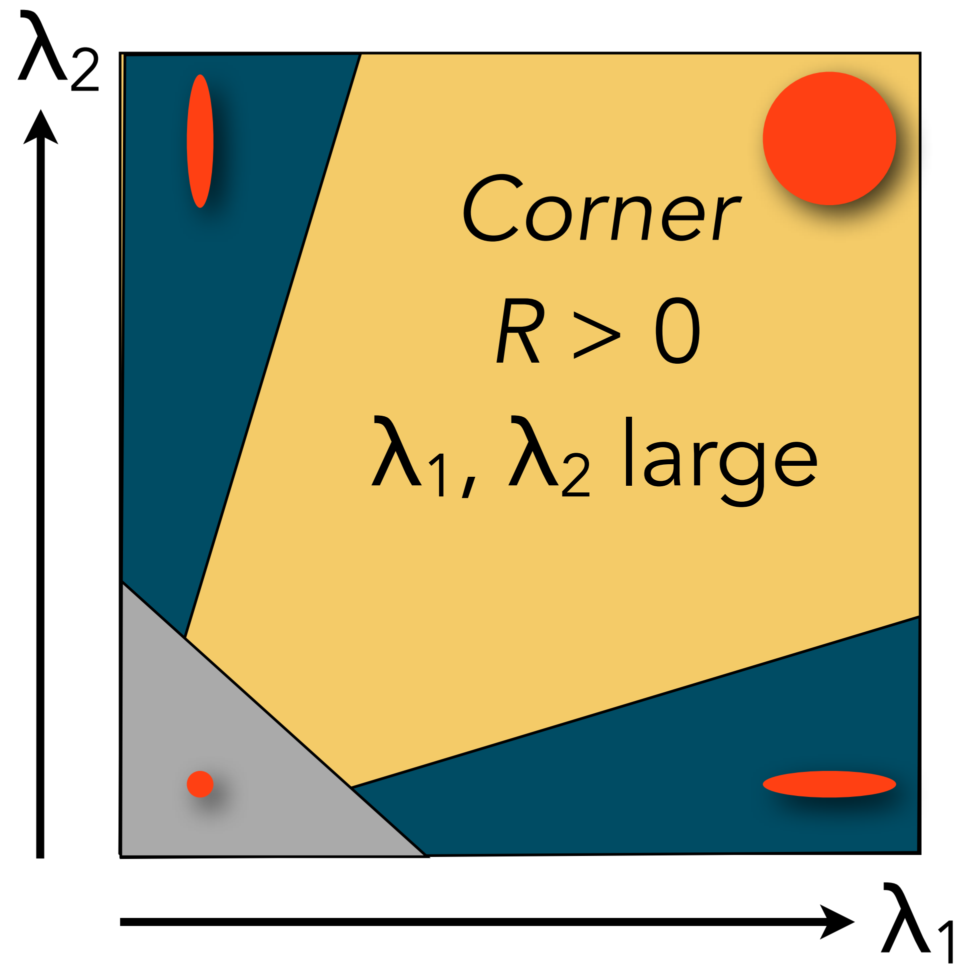


Eigenvalues

$$R = \det(M) - \alpha\ \mathrm{trace}(M)^2 = \lambda_1\lambda_2 - \alpha(\lambda_1 + \lambda_2)^2$$

α: constant (0.04 to 0.06)

- ★ *R* depends only on eigenvalues of *M*
- ★ *R* is large for a corner
- ★ *R* is negative with large magnitude for an edge

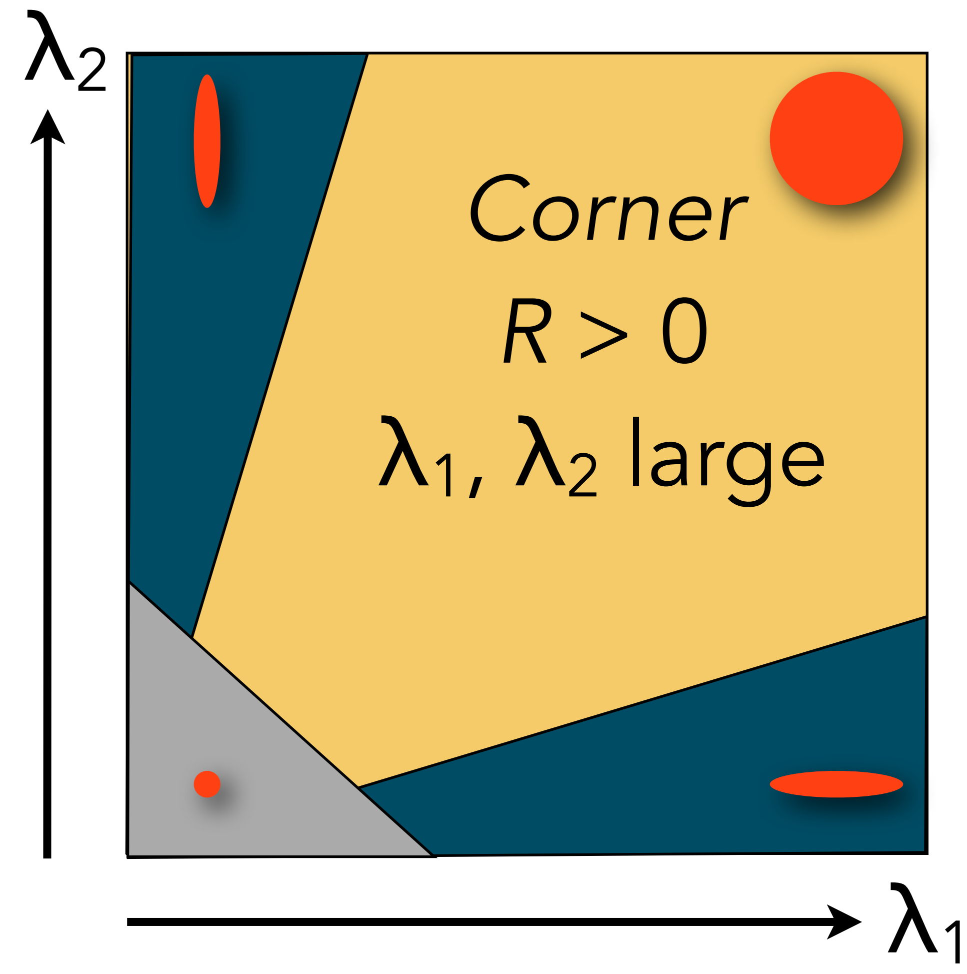


Eigenvalues

$$R = \det(M) - \alpha\ \mathrm{trace}(M)^2 = \lambda_1\lambda_2 - \alpha(\lambda_1 + \lambda_2)^2$$

α: constant (0.04 to 0.06)

- ★ *R* depends only on eigenvalues of *M*
- ★ *R* is large for a corner
- ★ *R* is negative with large magnitude for an edge
- ★ *|R|* is small for a flat region

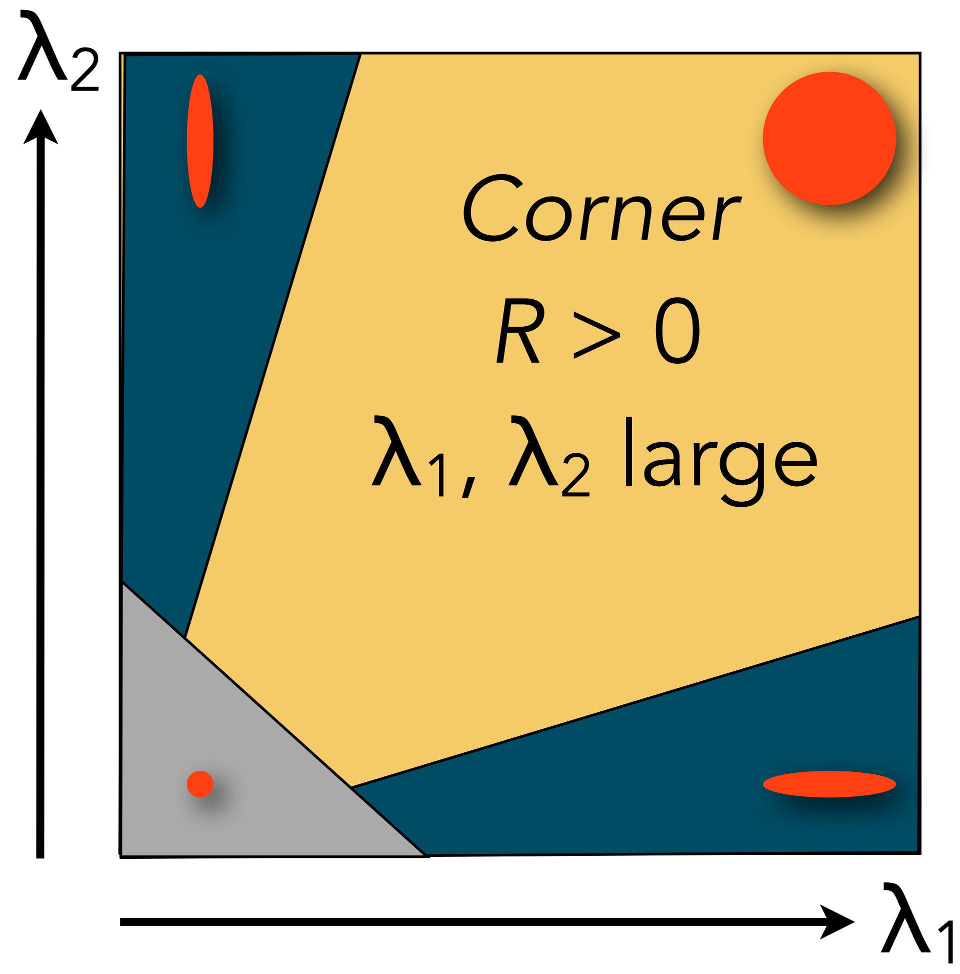


Eigenvalues

$$R = \det(M) - \alpha \, \text{trace}(M)^2 = \lambda_1 \lambda_2 - \alpha(\lambda_1 + \lambda_2)^2$$

α: constant (0.04 to 0.06)

- ★ *R* depends only on eigenvalues of *M*
- ★ *R* is large for a corner
- ★ *R* is negative with large magnitude for an edge
- ★ *|R|* is small for a flat region
- ★ Note: No explicit computation of eigenvalues required.



Harris Detector Algorithm

Harris Detector Algorithm

1. Compute Gaussian derivatives at each pixel

Harris Detector Algorithm

1. Compute Gaussian derivatives at each pixel
2. Compute second moment matrix *M* in a Gaussian window around each pixel

Harris Detector Algorithm

1. Compute Gaussian derivatives at each pixel
2. Compute second moment matrix *M* in a Gaussian window around each pixel
3. Compute corner response function *R*

Harris Detector Algorithm

1. Compute Gaussian derivatives at each pixel
2. Compute second moment matrix *M* in a Gaussian window around each pixel
3. Compute corner response function *R*
4. Threshold *R*

Harris Detector Algorithm

1. Compute Gaussian derivatives at each pixel
2. Compute second moment matrix *M* in a Gaussian window around each pixel
3. Compute corner response function *R*
4. Threshold *R*
5. Find local maxima of response function (non-maximum suppression)

Properties of the Harris Detector

Properties of the Harris Detector

★ Rotation Invariant?

- Ellipses rotate, but shape (eigenvalues) remain same
- Corner Response R is invariant

Properties of the Harris Detector

- Rotation Invariant?
  - Ellipses rotate, but shape (eigenvalues) remain same
  - Corner Response R is invariant

Properties of the Harris Detector

- Rotation Invariant?
  - Ellipses rotate, but shape (eigenvalues) remain same
  - Corner Response R is invariant
- Intensity Invariant?
  - Partial invariance to additive and multiplicative intensity changes (threshold issue for multiplicative)
  - Only Image derivatives are used

Properties of the Harris Detector

- Rotation Invariant?
  - Ellipses rotate, but shape (eigenvalues) remain same
  - Corner Response R is invariant
- Intensity Invariant?
  - Partial invariance to additive and multiplicative intensity changes (threshold issue for multiplicative)
  - Only Image derivatives are used

Properties of the Harris Detector

★ Rotation Invariant?

- Ellipses rotate, but shape (eigenvalues) remain same
- Corner Response R is invariant

★ Intensity Invariant?

- Partial invariance to additive and multiplicative intensity changes (threshold issue for multiplicative)
- Only Image derivatives are used

Properties of the Harris Detector

- Rotation Invariant?
  - Ellipses rotate, but shape (eigenvalues) remain same
  - Corner Response R is invariant
- Intensity Invariant?
  - Partial invariance to additive and multiplicative intensity changes (threshold issue for multiplicative)
  - Only Image derivatives are used
- Scale Invariant
  - NO! Dependent on Window Size!
  - USE Pyramids (or Frequency Domain!)

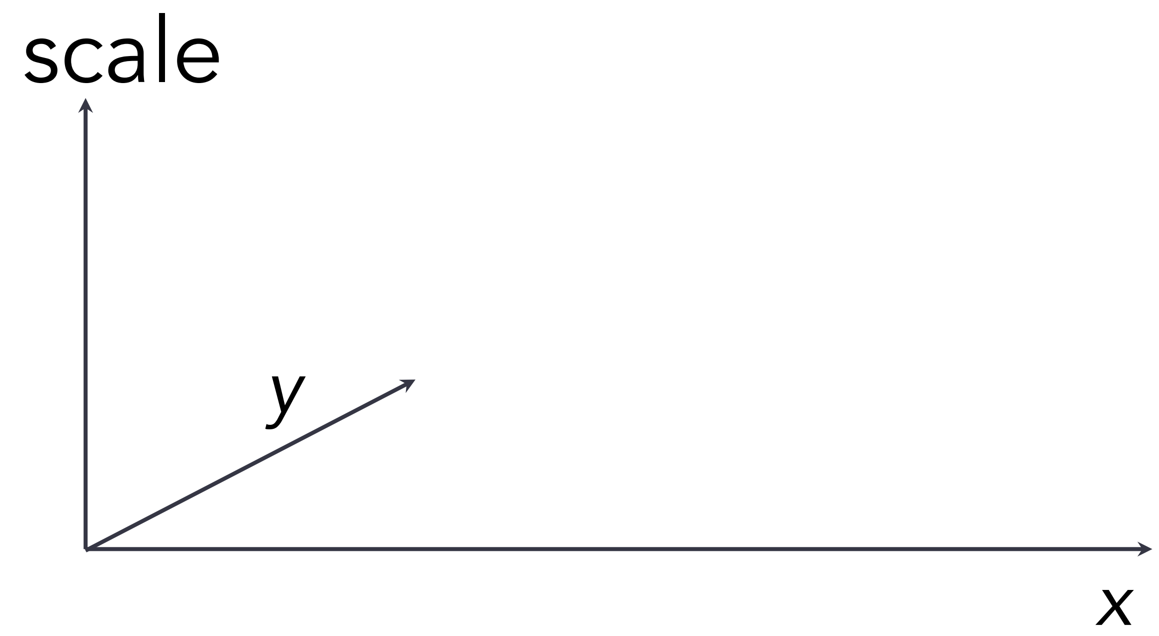
Scale Invariant Detectors

Scale Invariant Detectors

- Harris-Laplacian (Mikolajczyk and Schmid, 2001)
  Find local maximum of:
  - Harris corner detector in space (image coordinates)
  - Laplacian in scale

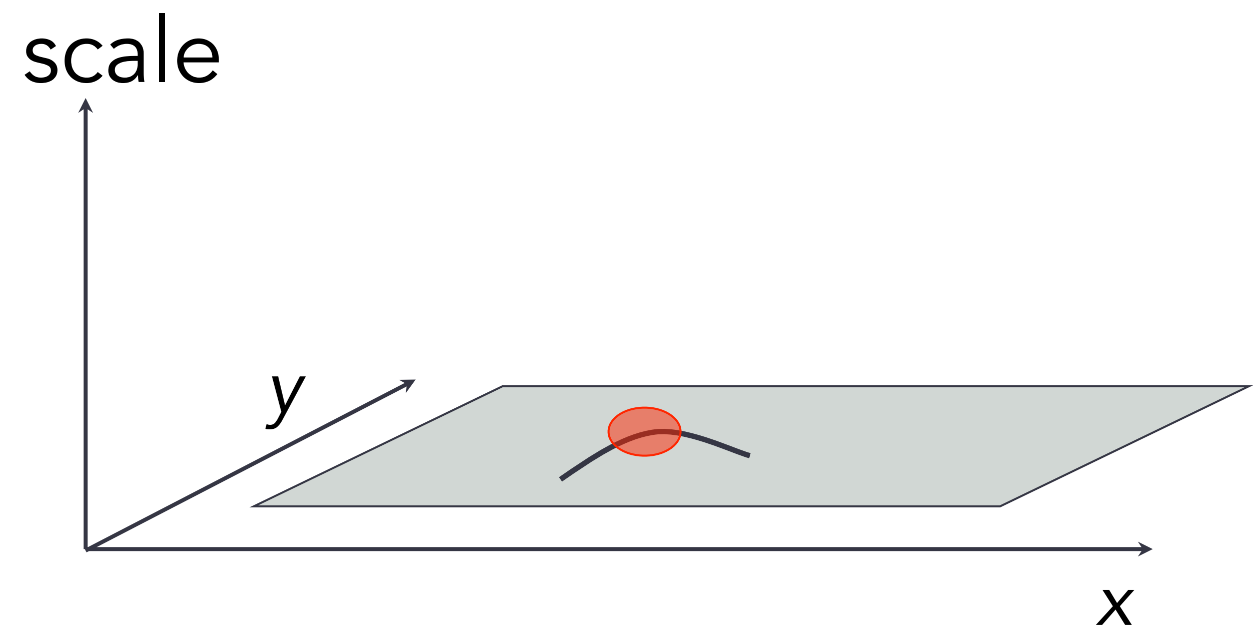
Scale Invariant Detectors

- Harris-Laplacian (Mikolajczyk and Schmid, 2001)
  Find local maximum of:
  - Harris corner detector in space (image coordinates)
  - Laplacian in scale



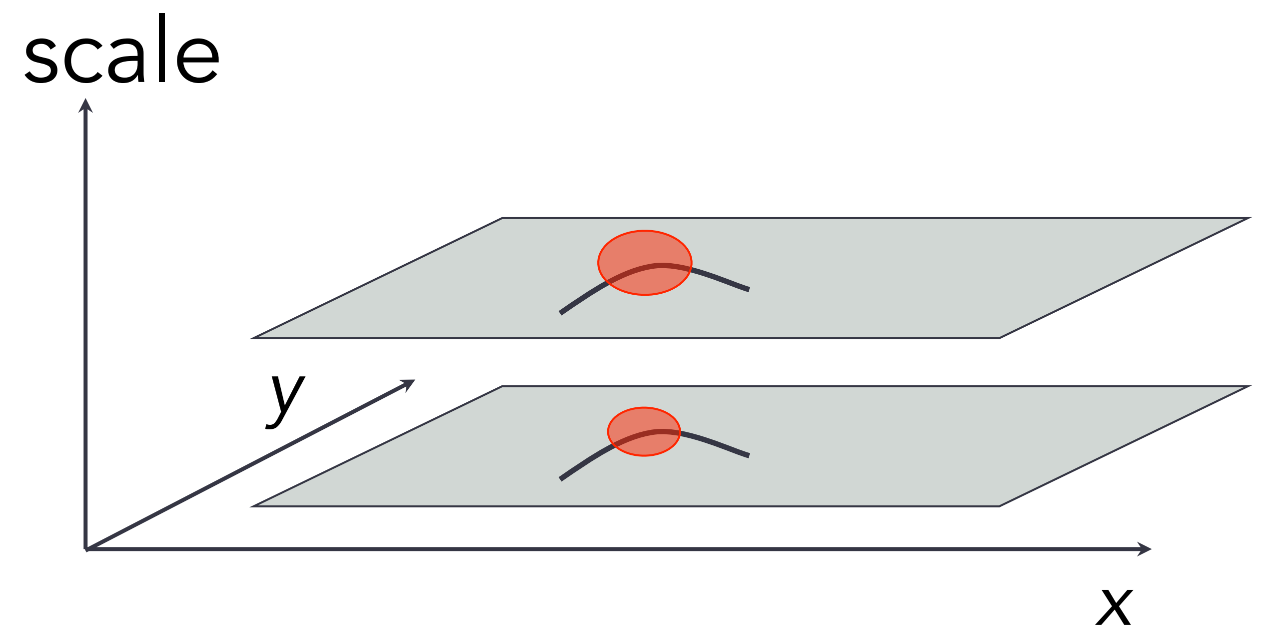
Scale Invariant Detectors

- Harris-Laplacian (Mikolajczyk and Schmid, 2001)
  Find local maximum of:
  - Harris corner detector in space (image coordinates)
  - Laplacian in scale



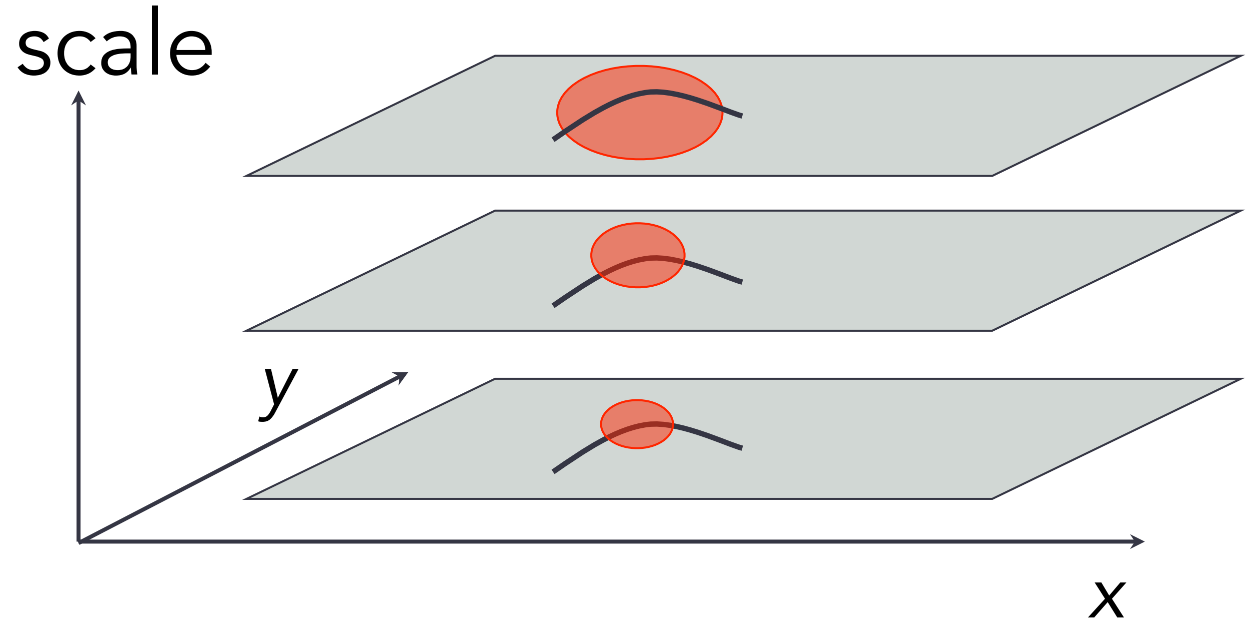
Scale Invariant Detectors

- Harris-Laplacian (Mikolajczyk and Schmid, 2001)
  Find local maximum of:
  - Harris corner detector in space (image coordinates)
  - Laplacian in scale



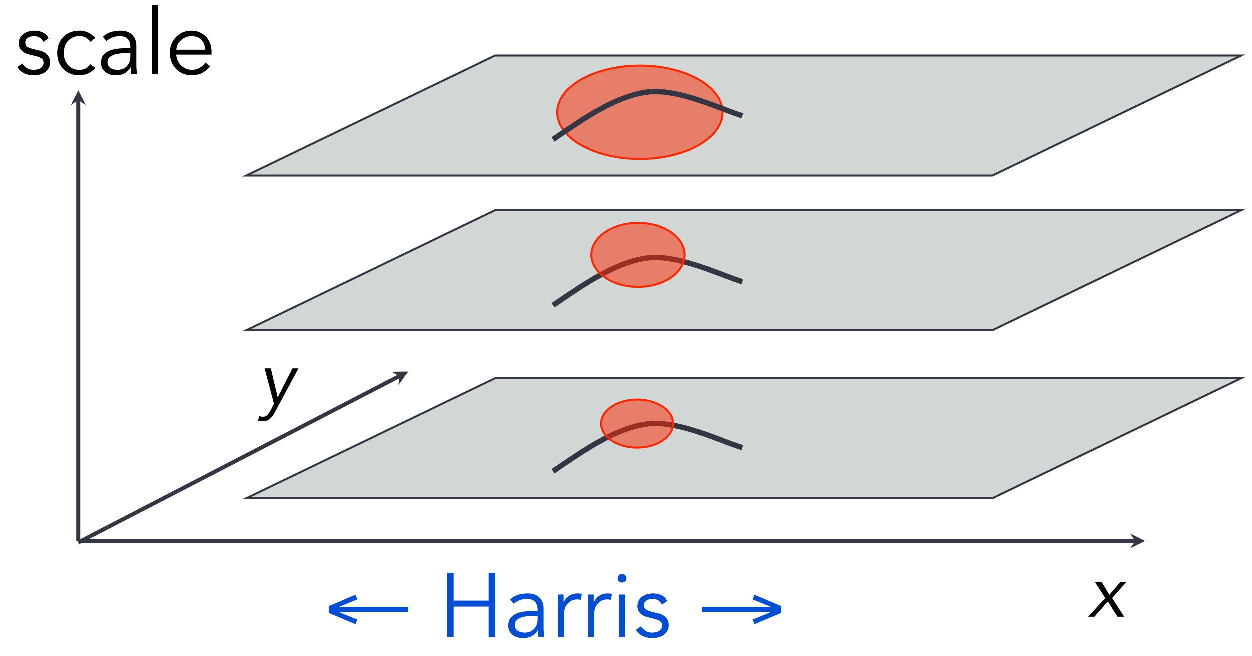
Scale Invariant Detectors

- Harris-Laplacian (Mikolajczyk and Schmid, 2001)
  Find local maximum of:
  - Harris corner detector in space (image coordinates)
  - Laplacian in scale



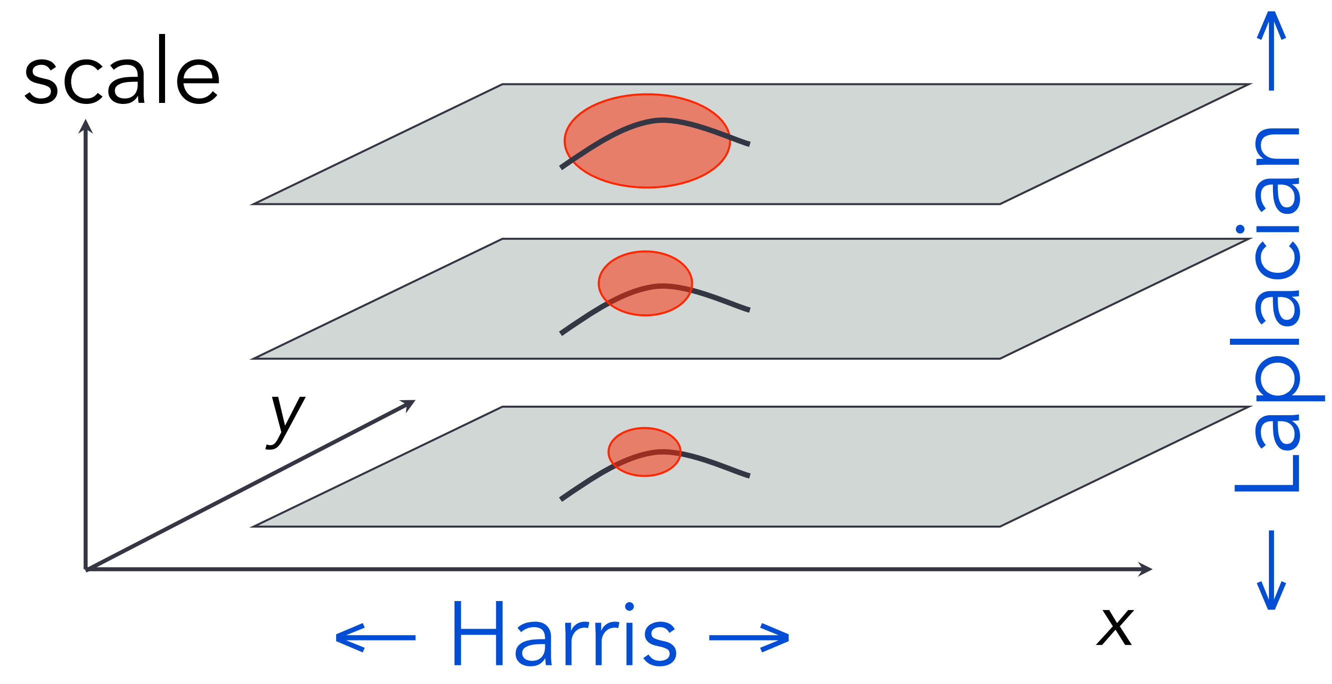
Scale Invariant Detectors

- Harris-Laplacian (Mikolajczyk and Schmid, 2001)
  Find local maximum of:
  - Harris corner detector in space (image coordinates)
  - Laplacian in scale



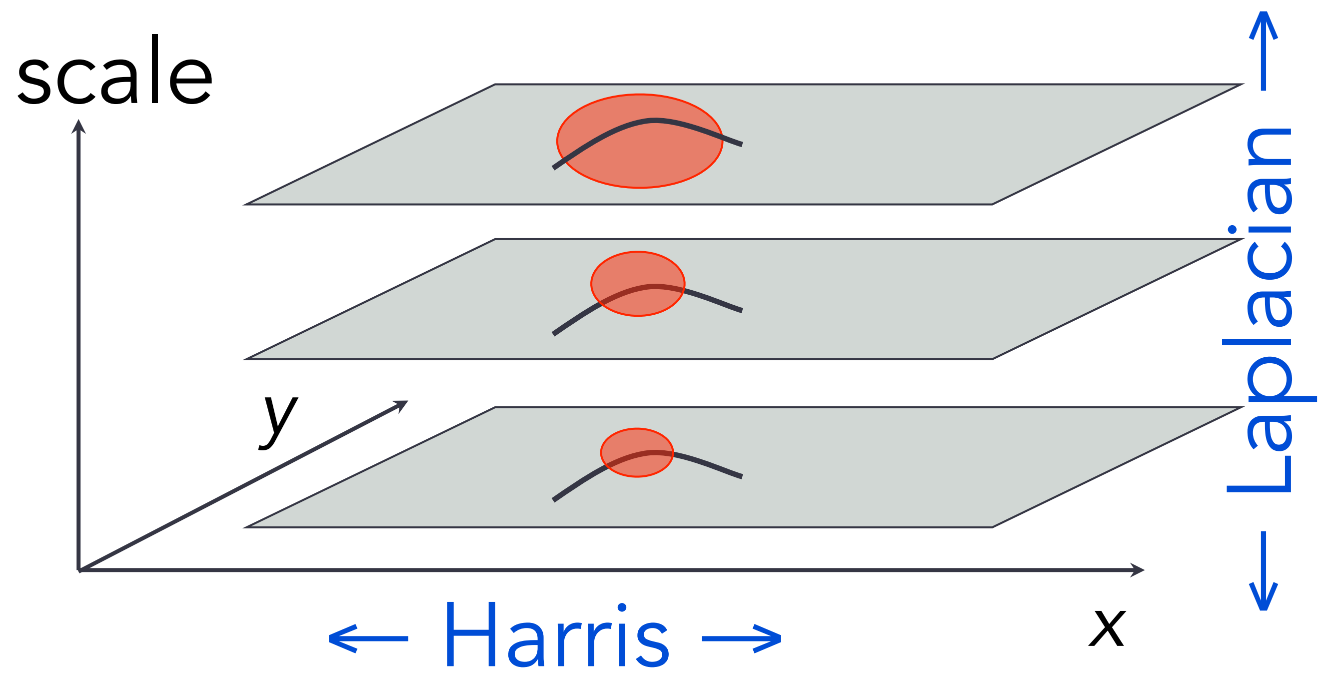
Scale Invariant Detectors

- Harris-Laplacian (Mikolajczyk and Schmid, 2001)
  Find local maximum of:
  - Harris corner detector in space (image coordinates)
  - Laplacian in scale



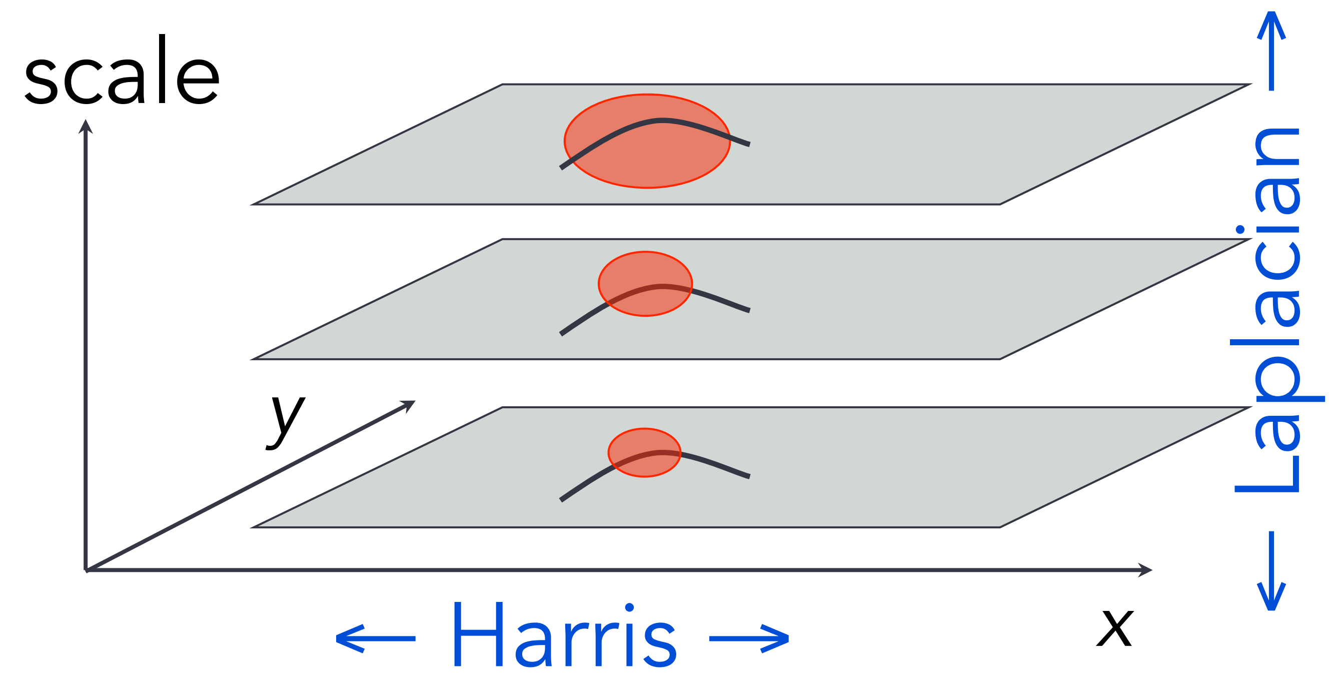
Scale Invariant Detectors

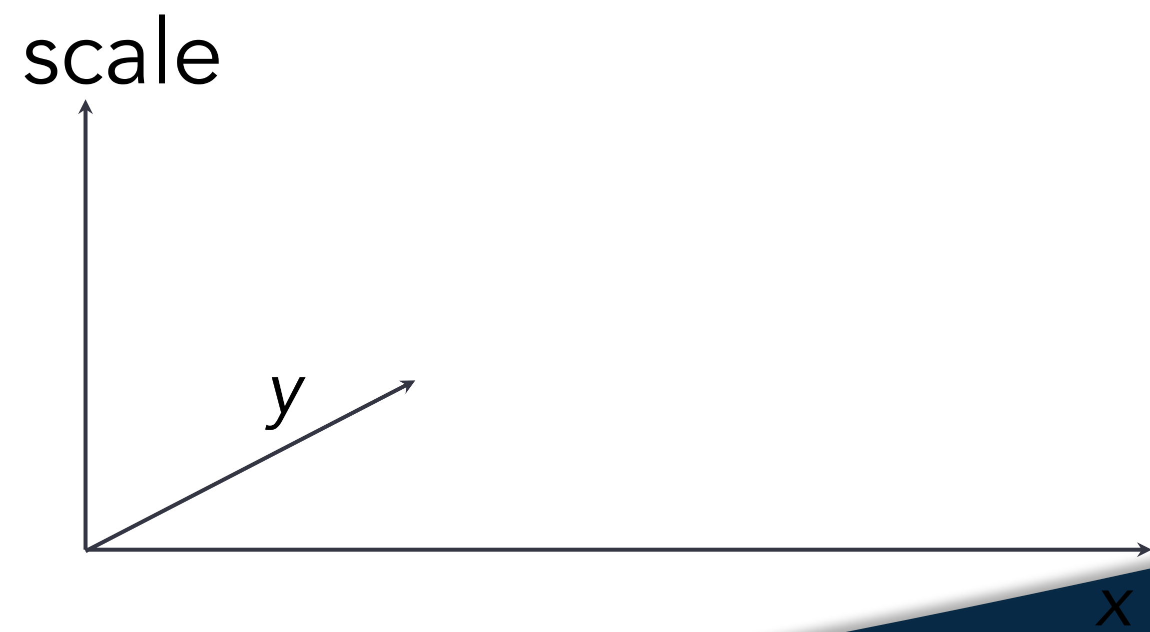
- Harris-Laplacian (Mikolajczyk and Schmid, 2001)
  Find local maximum of:
  - Harris corner detector in space (image coordinates)
  - Laplacian in scale
- SIFT (Lowe, 2004)
  Find local maximum of:
  - Difference of Gaussians (DoG) in space and scale
  - Where DoG is simply a pyramid of the difference of Gaussians within each octave



Scale Invariant Detectors

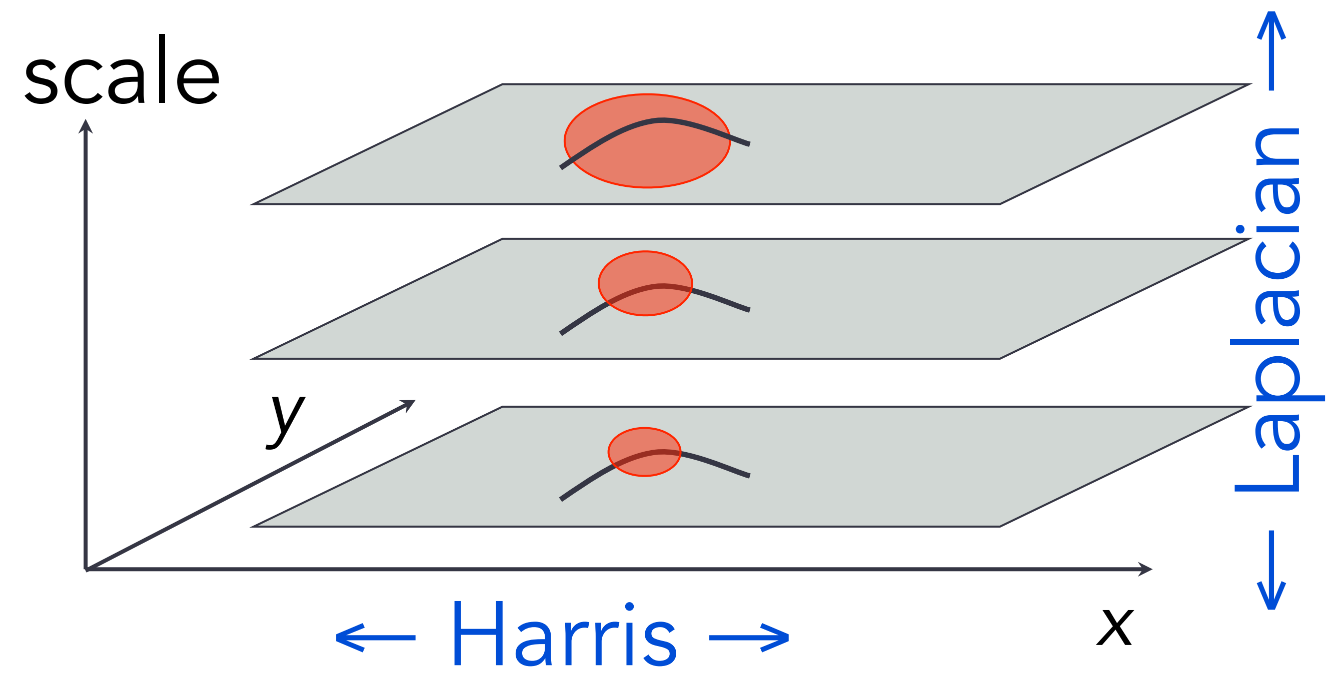
- Harris-Laplacian (Mikolajczyk and Schmid, 2001)
  Find local maximum of:
  - Harris corner detector in space (image coordinates)
  - Laplacian in scale
- SIFT (Lowe, 2004)
  Find local maximum of:
  - Difference of Gaussians (DoG) in space and scale
  - Where DoG is simply a pyramid of the difference of Gaussians within each octave

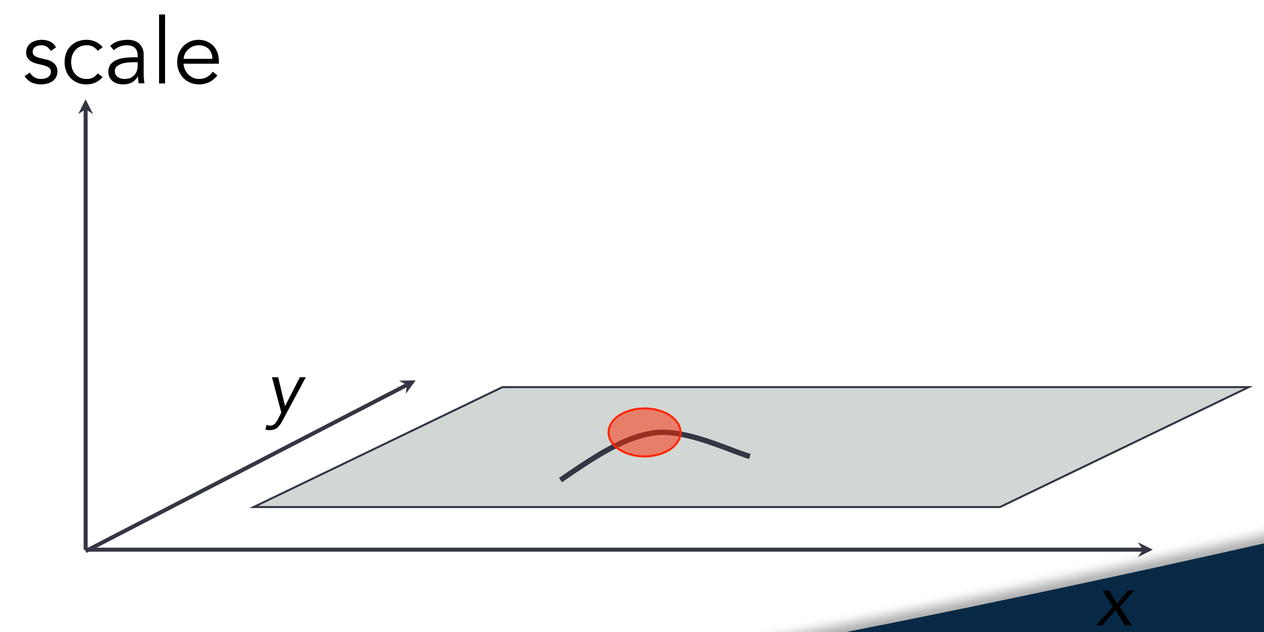




Scale Invariant Detectors

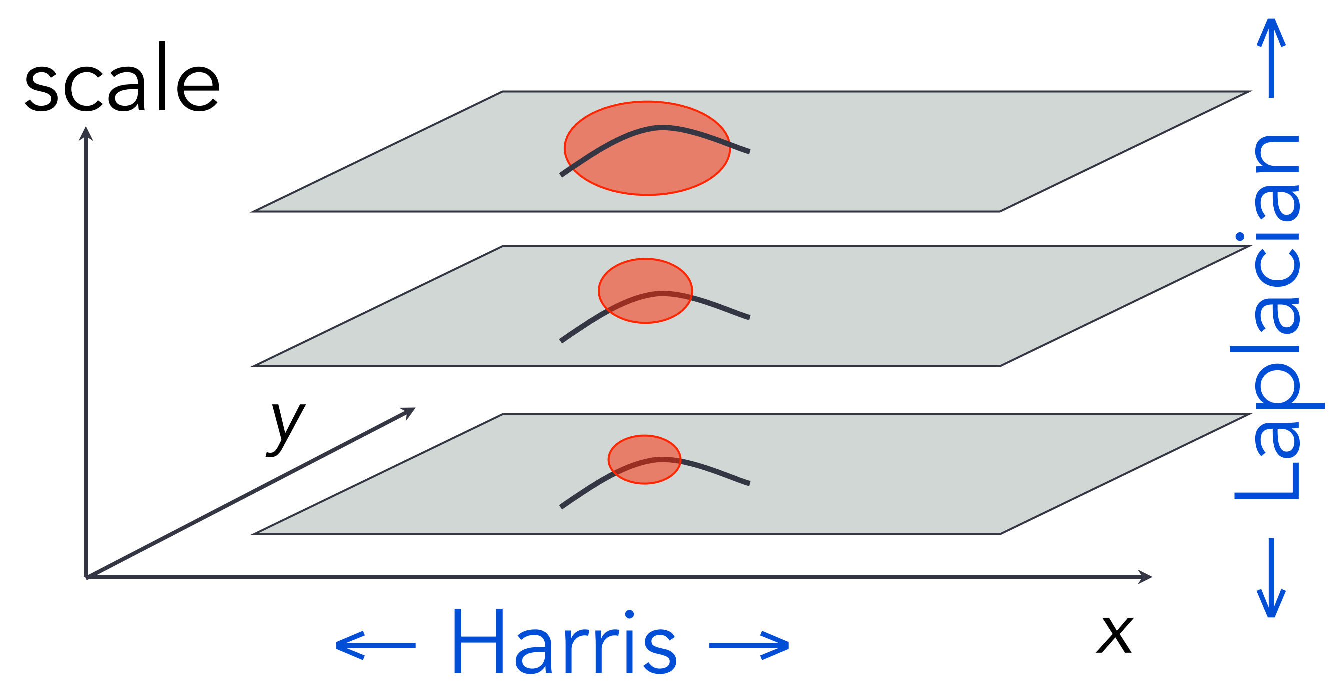
- Harris-Laplacian (Mikolajczyk and Schmid, 2001)
  Find local maximum of:
  - Harris corner detector in space (image coordinates)
  - Laplacian in scale
- SIFT (Lowe, 2004)
  Find local maximum of:
  - Difference of Gaussians (DoG) in space and scale
  - Where DoG is simply a pyramid of the difference of Gaussians within each octave

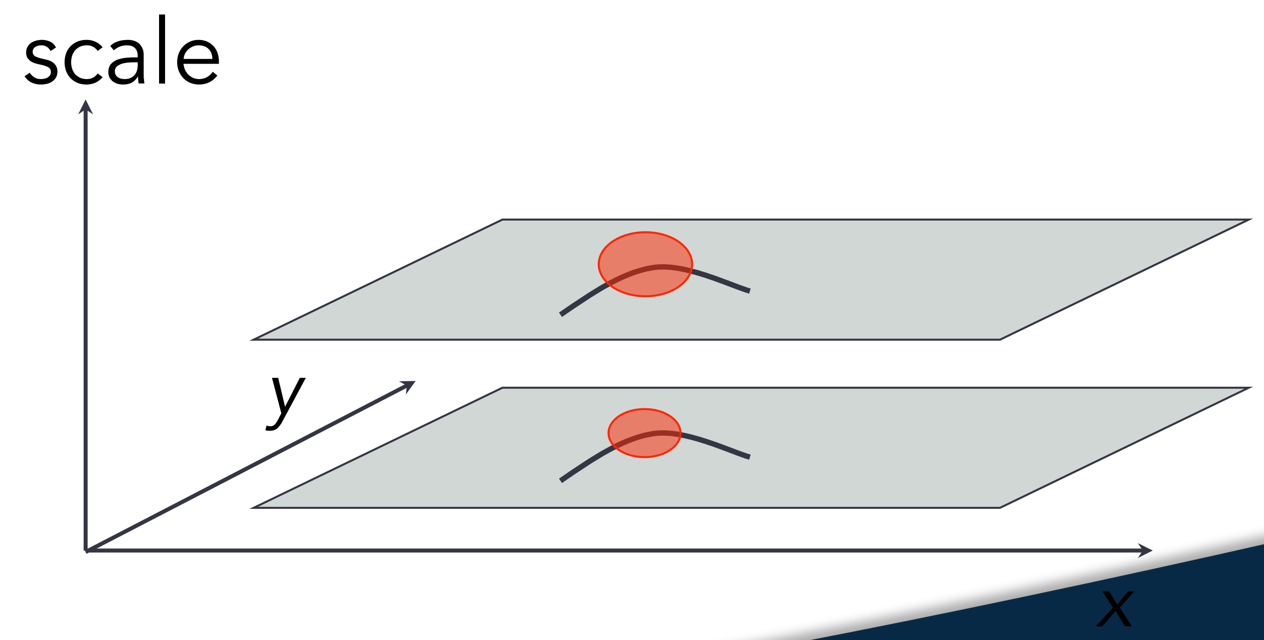




Scale Invariant Detectors

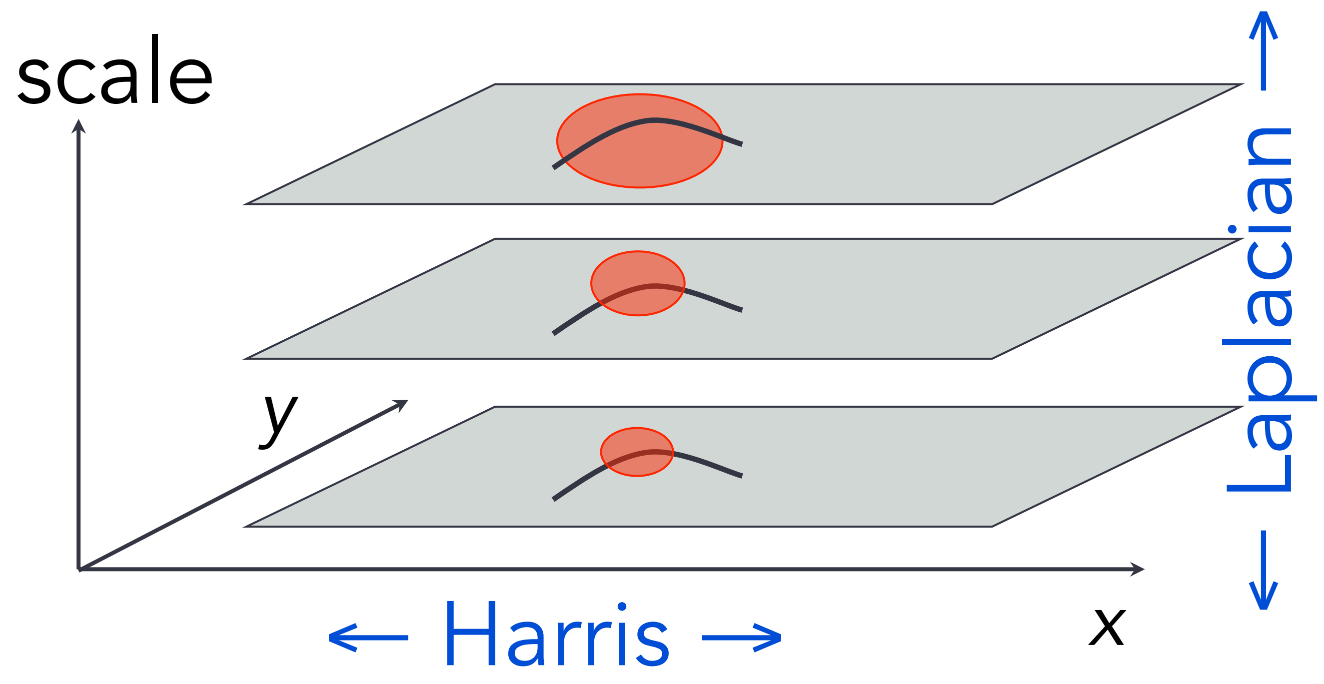
- ★ Harris-Laplacian (Mikolajczyk and Schmid, 2001)
  Find local maximum of:
  - ★ Harris corner detector in space (image coordinates)
  - ★ Laplacian in scale
- ★ SIFT (Lowe, 2004)
  Find local maximum of:
  - ★ Difference of Gaussians (DoG) in space and scale
  - ★ Where DoG is simply a pyramid of the difference of Gaussians within each octave

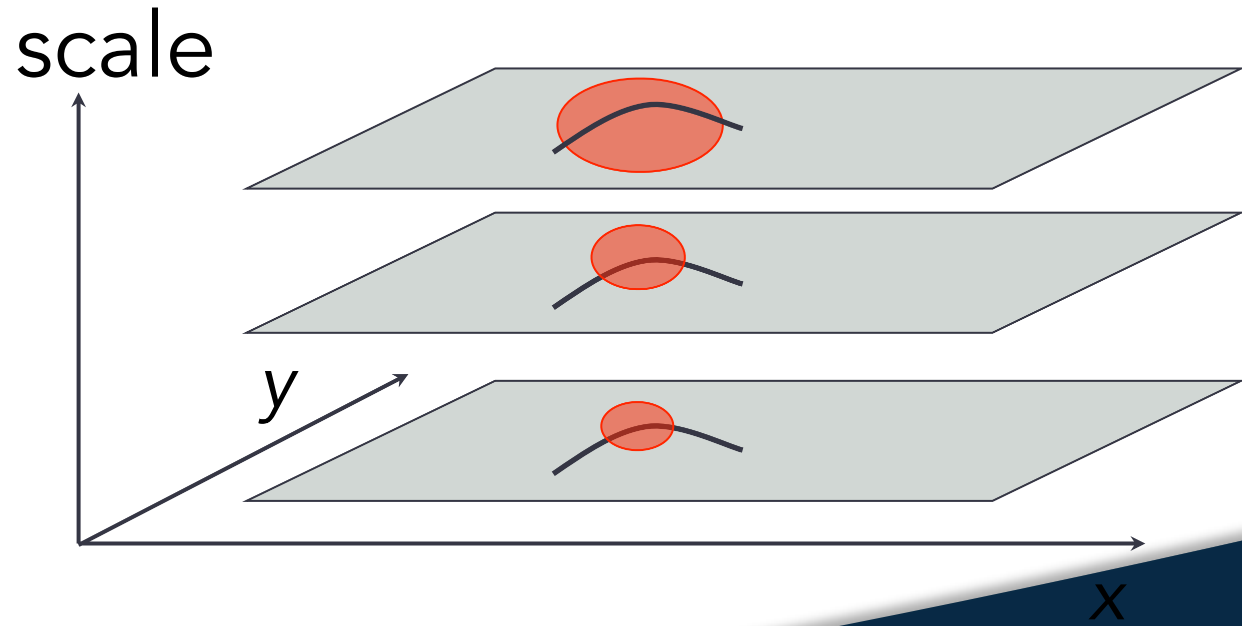




Scale Invariant Detectors

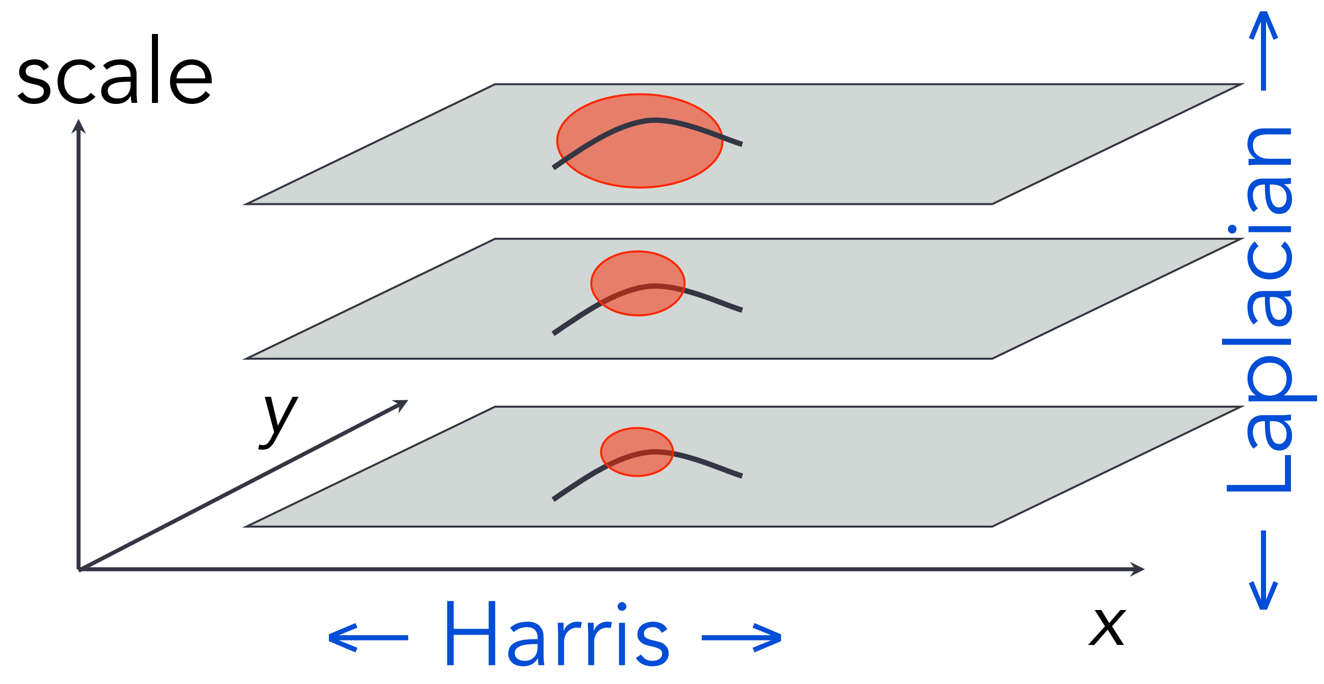
- Harris-Laplacian (Mikolajczyk and Schmid, 2001)
  Find local maximum of:
  - Harris corner detector in space (image coordinates)
  - Laplacian in scale
- SIFT (Lowe, 2004)
  Find local maximum of:
  - Difference of Gaussians (DoG) in space and scale
  - Where DoG is simply a pyramid of the difference of Gaussians within each octave

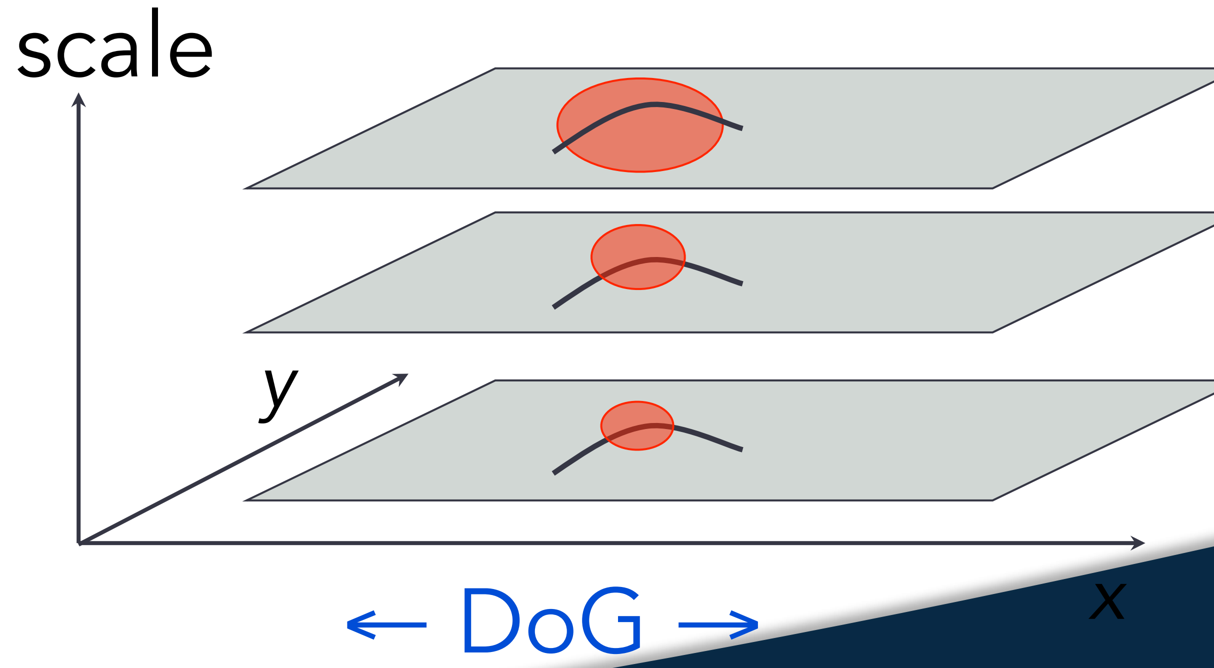




Scale Invariant Detectors

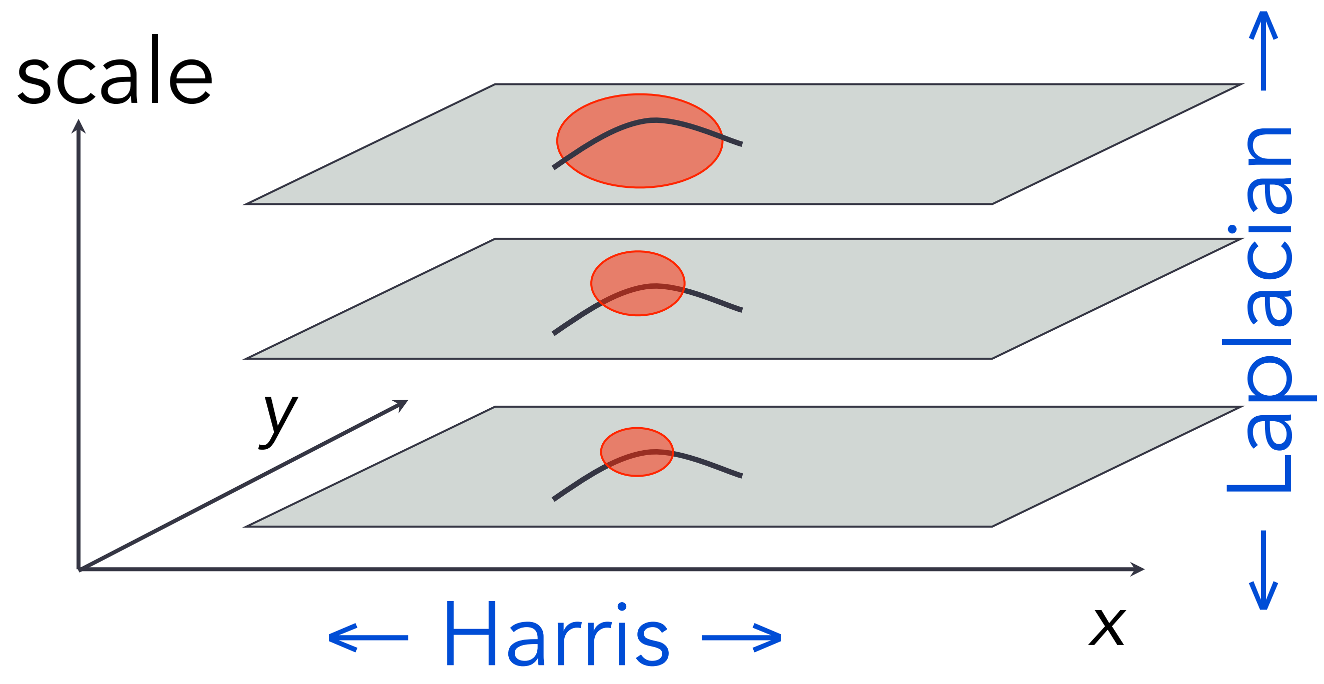
- Harris-Laplacian (Mikolajczyk and Schmid, 2001)
  Find local maximum of:
  - Harris corner detector in space (image coordinates)
  - Laplacian in scale
- SIFT (Lowe, 2004)
  Find local maximum of:
  - Difference of Gaussians (DoG) in space and scale
  - Where DoG is simply a pyramid of the difference of Gaussians within each octave

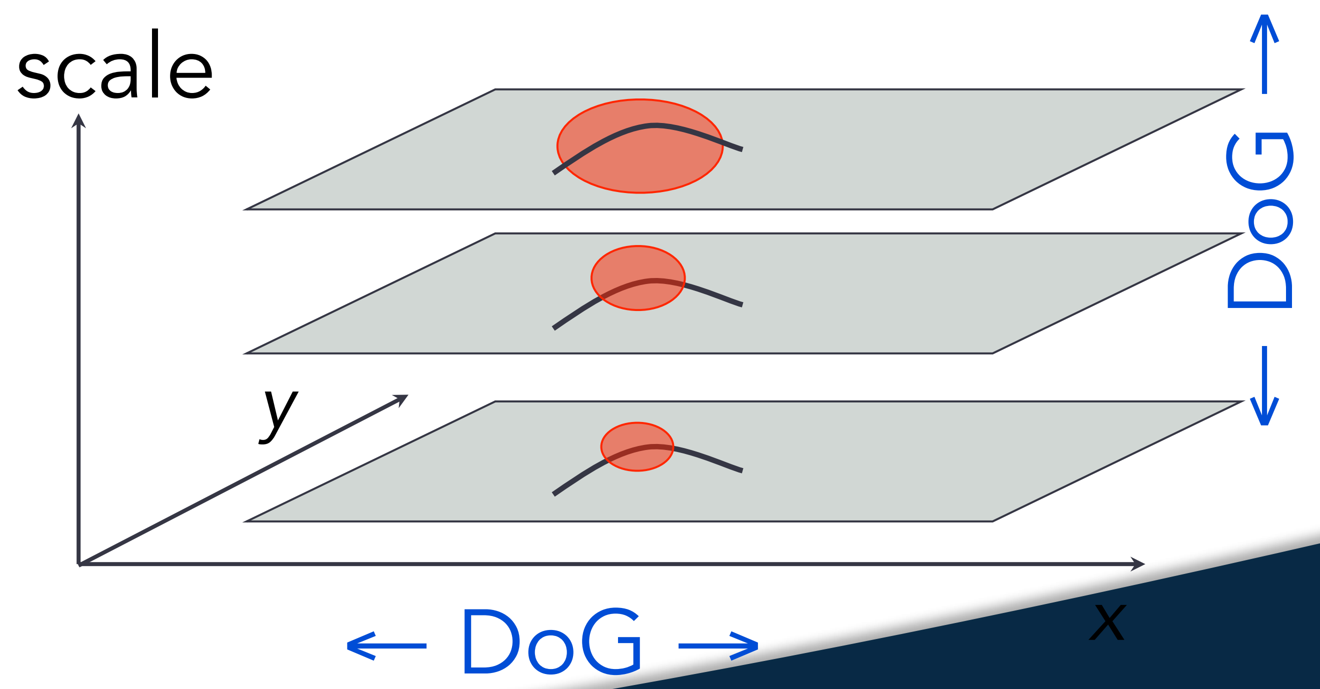




Scale Invariant Detectors

- Harris-Laplacian (Mikolajczyk and Schmid, 2001)
  Find local maximum of:
  - Harris corner detector in space (image coordinates)
  - Laplacian in scale
- SIFT (Lowe, 2004)
  Find local maximum of:
  - Difference of Gaussians (DoG) in space and scale
  - Where DoG is simply a pyramid of the difference of Gaussians within each octave





SIFT (Scale-Invariant Feature Transform)

http://en.wikipedia.org/wiki/File:Sift_keypoints_filtering.jpg

19

SIFT (Scale-Invariant Feature Transform)

- ★ Scale-space extrema detection
  - Search over multiple scales and image locations

http://en.wikipedia.org/wiki/File:Sift_keypoints_filtering.jpg

SIFT (Scale-Invariant Feature Transform)

- ★ Scale-space extrema detection
  - Search over multiple scales and image locations
- ★ Keypoint localization (not just corners!)
  - Define a model to determine location and scale. Select keypoints based on a measure of stability.

http://en.wikipedia.org/wiki/File:Sift_keypoints_filtering.jpg

SIFT (Scale-Invariant Feature Transform)

- ★ Scale-space extrema detection
  - Search over multiple scales and image locations
- ★ Keypoint localization (not just corners!)
  - Define a model to determine location and scale. Select keypoints based on a measure of stability.
- ★ Orientation assignment
  - Compute best orientation(s) for each keypoint region.

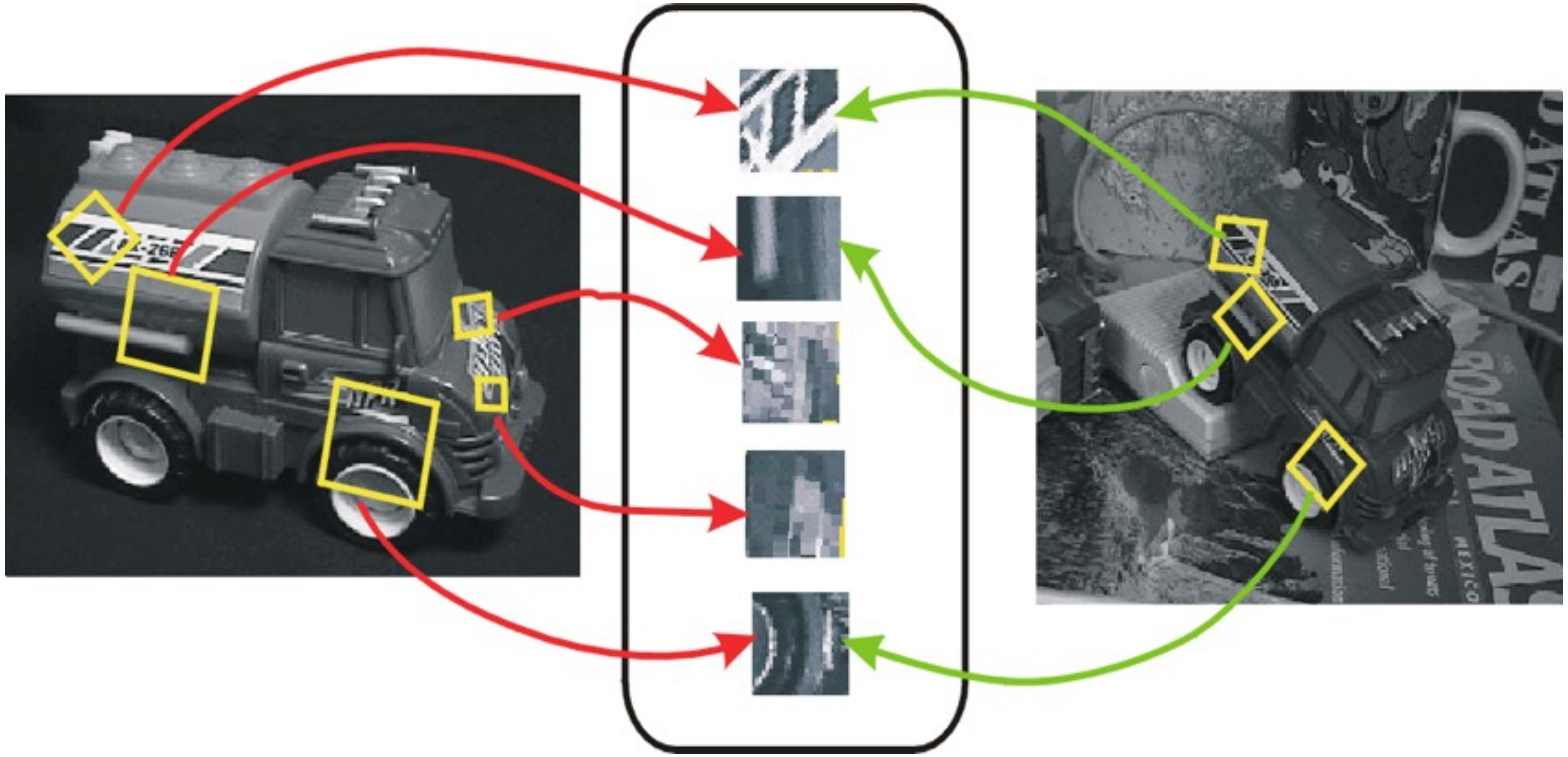
http://en.wikipedia.org/wiki/File:Sift_keypoints_filtering.jpg

SIFT (Scale-Invariant Feature Transform)

- ★ Scale-space extrema detection
  - Search over multiple scales and image locations
- ★ Keypoint localization (not just corners!)
  - Define a model to determine location and scale. Select keypoints based on a measure of stability.
- ★ Orientation assignment
  - Compute best orientation(s) for each keypoint region.
- ★ Keypoint description
  - Use local image gradients at selected scale and rotation to describe each keypoint region.

http://en.wikipedia.org/wiki/File:Sift_keypoints_filtering.jpg

19



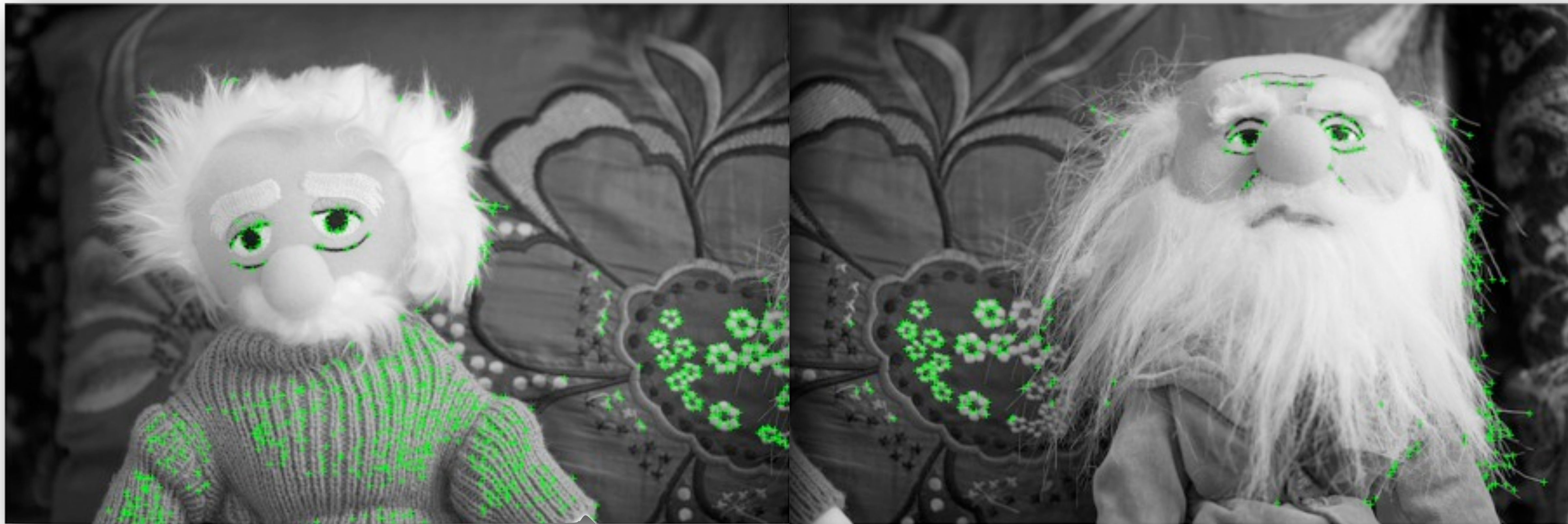
Invariant Local Features

- ★ Image content is transformed into local feature coordinates that are invariant to translation, rotation, scale, and other imaging parameters
- ★ (Lowe, 2004)

20

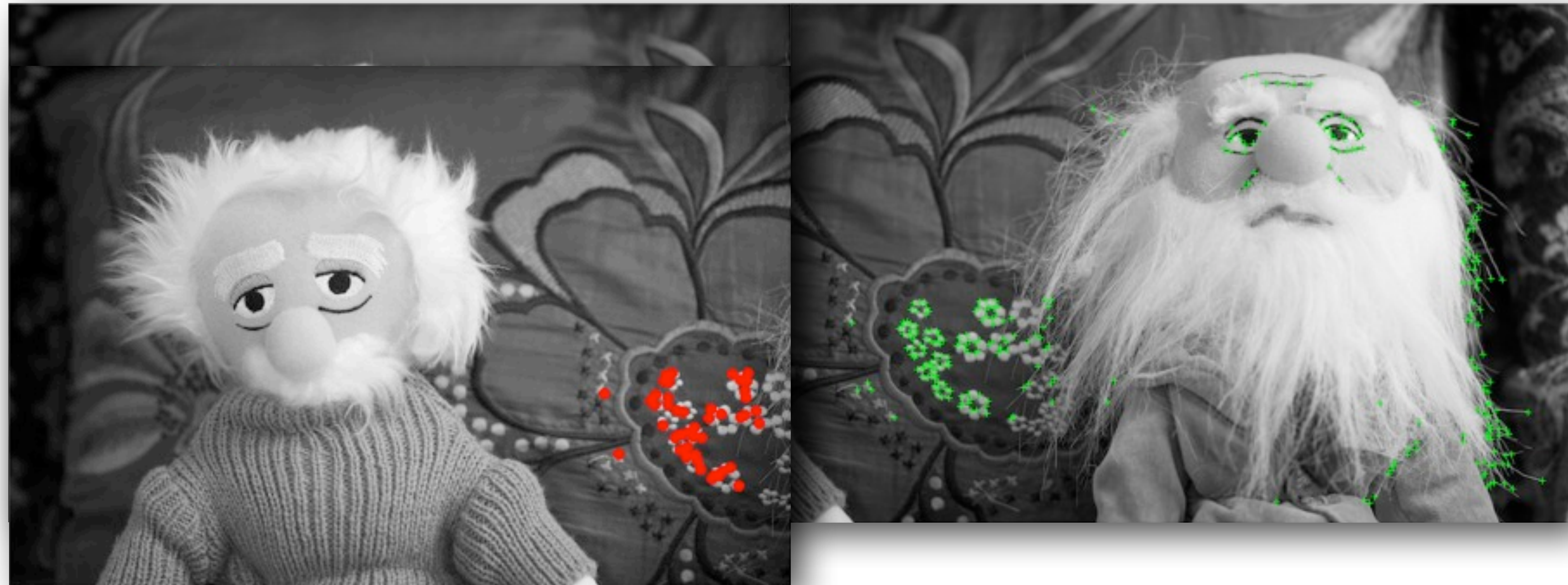
Results

Detect



[Results](#)

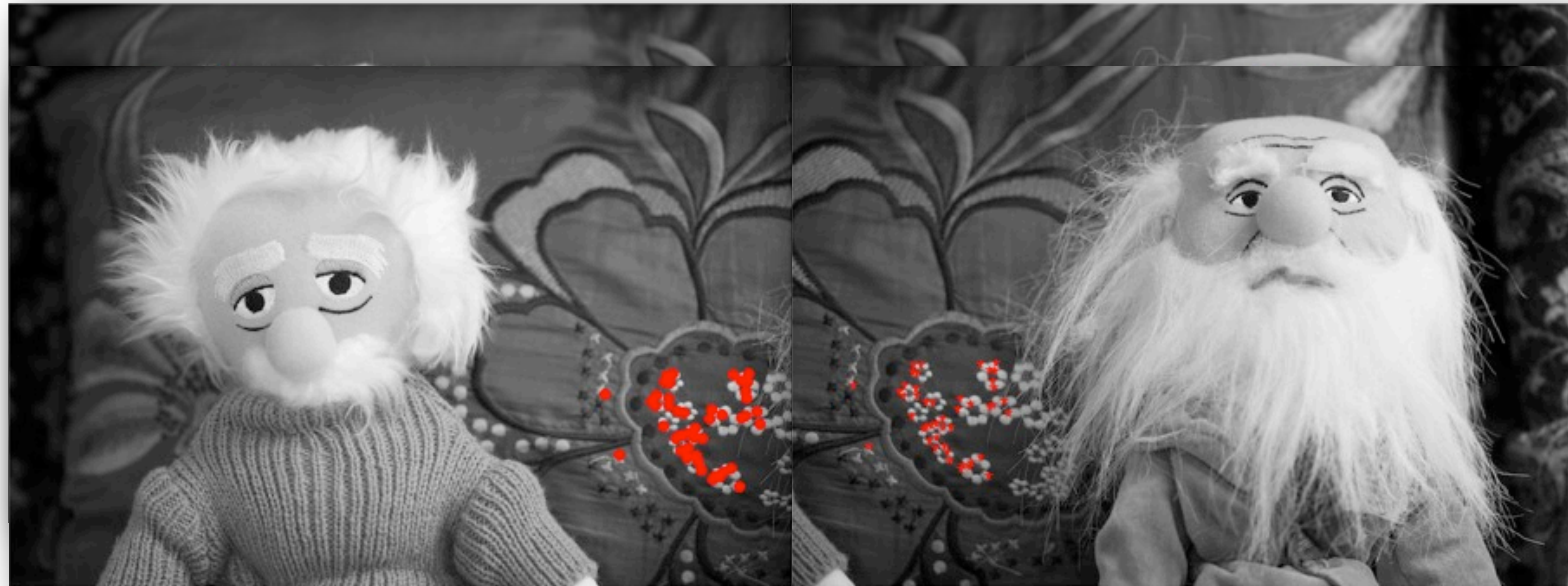
Detect



Results

Detect

Match

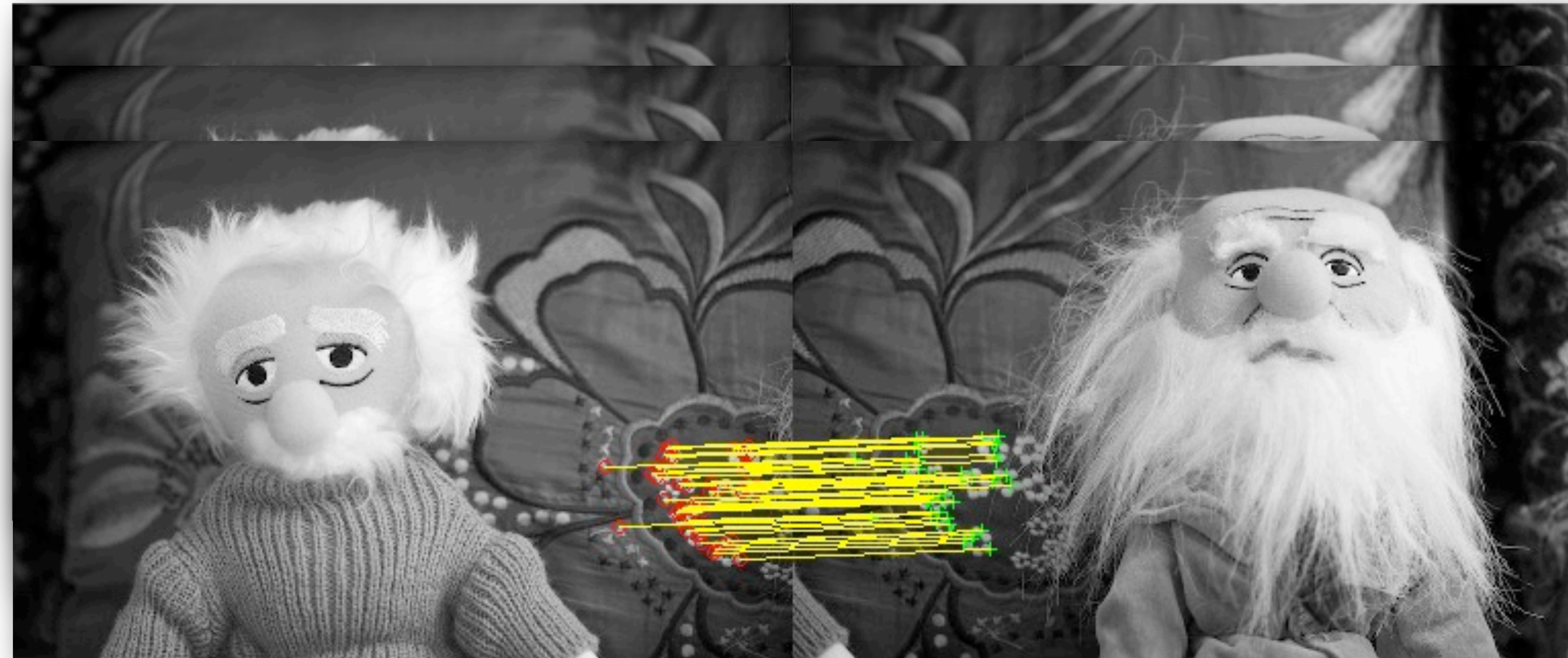


Results

Detect

Match

Show



Results

Summary

- ★ Introduced Feature Detection and Matching for images
- ★ Discussed the four (4) Characteristics of Good Features.
- ★ Introduced the Harris Corner Detector Framework
- ★ Introduced the SIFT detector

Further Reading

- ★ Harris and Stephens (1988) "A Combined Corner and Edge Detector." Proceedings of the 4th Alvey Vision Conference, 1988, [PDF][DOI]
- ★ Mikolajczyk and Schmid (2001). "Indexing Based on Scale Invariant Interest Points". ICCV 2001
- ★ Lowe (2004) "Distinctive Image Features from Scale-Invariant Keypoints". IJCV 2004
- ★ Search for "Features" on OpenCV site

commons.wikimedia.org/

Next Class

★ How to do Alignment for Panoramas

Credits

★ For more information, see

- Richard Szeliski (2010) Computer Vision: Algorithms and Applications, Springer.

★ Some concepts in slides motivated by similar slides by A. Efros and J. Hays.

★ Some images retrieved from

- http://commons.wikimedia.org/.
- List will be available on website.

www.flickr.com/photos/neneonline/231886965/

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.