INSTITUTO DE INFORMÁTICA
Universidade Federal de Goiás

Computer Vision

Corner detection

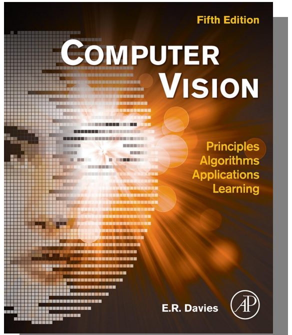
Prof. Dr. Gustavo T Laureano

INSTITUTO DE INFORMÁTICA
UFG

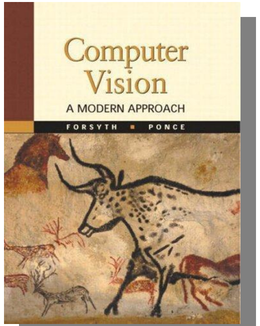
UFG
UNIVERSIDADE FEDERAL DE GOIÁS

Course References

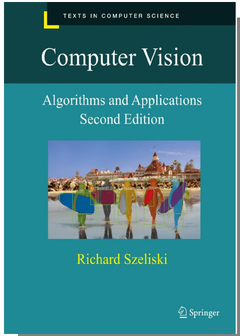
Many of the words of this course are taken from the books:



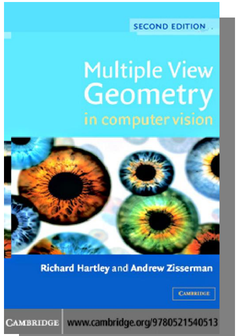
Computer & Machine Vision – Theory, Algorithms, Practicalities.
Fourth Edition, 2018. Elsevier Inc.
E. Roy Davies



Computer Vision – A modern Approach.
2003. Prentice Hall.
David Forsyth and Jean Ponce



Computer Vision – Algorithm and Applications.
2021. Springer.
Richard Szeliski



Multiple View Geometry.
Second Edition, 2012.
Cambridge.
Richard Hartley & Andrew Zisserman

Objectives

Corner detection

Harris/Plessey

Shi-Tomasi

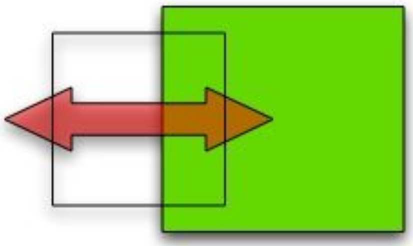
Susan

FAST

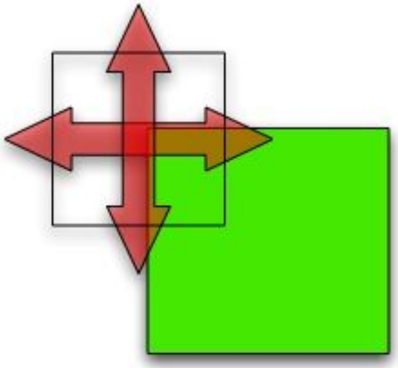
Blob Detection

Low Level Features (Corners)

What is a corner?



Edge

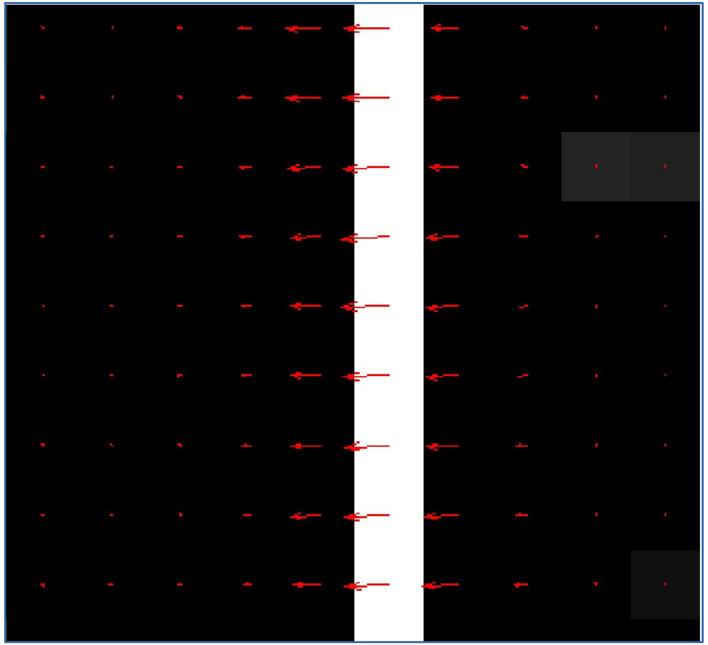


Corner

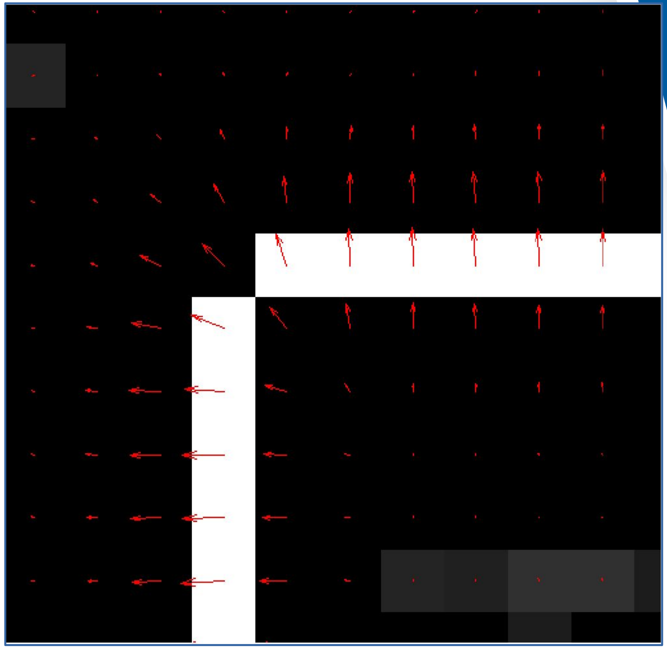
Low Level Features (Corners)

What is a corner?

Gradients:



Edge

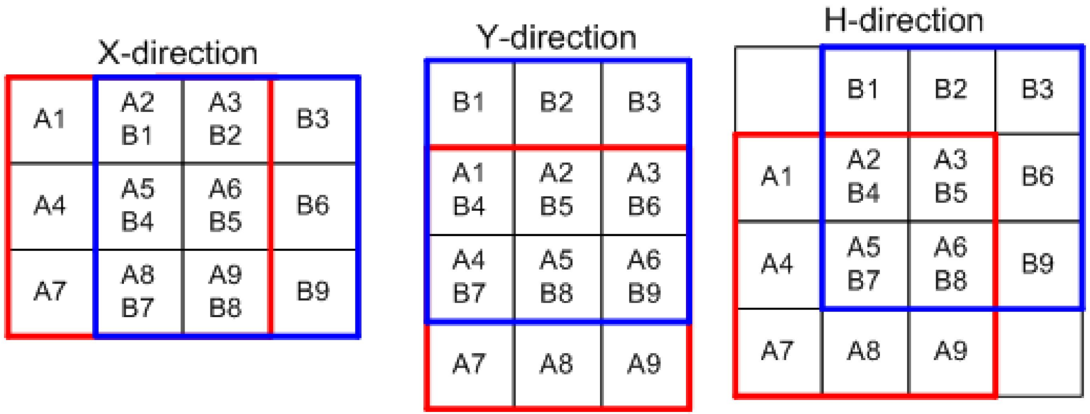


Corner

Low Level Features (Corners)

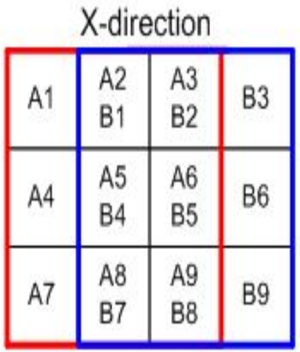
Moravec Operator:

- Transformation described by Hans P. Moravec (1977) to identify interest points:
- Points with great intensity variation in your neighborhood
- Principle:
    - Verifies intensity variation of shifted regions around central point



Low Level Features (Corners)

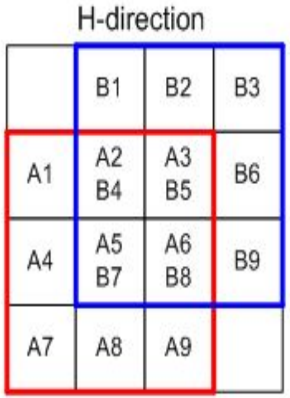
- Moravec Operator:



Horizontal Moravec Intensity Variation:

$$V_x = \sum_{i=1}^{9}(A_i - B_i)^2 = \sum_{i=1}^{9}(B_i - A_i)^2 \approx \sum_{i=1}^{9}\left(\frac{\partial I_i}{\partial x}\right)^2$$

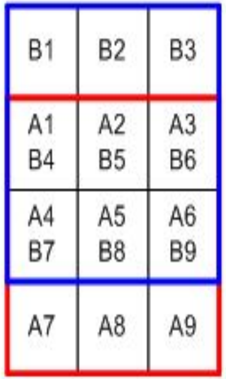
$$\text{where: } \frac{\partial I_i}{\partial x} \equiv I_i \otimes (-1,0,1) \approx B_i - A_i$$



Diagonal Moravec Intensity Variation:

$$V_h = \sum_{i=1}^{9}(A_i - B_i)^2 = \sum_{i=1}^{9}(B_i - A_i)^2 \approx \sum_{i=1}^{9}\left(\frac{\partial I_i}{\partial h}\right)^2$$

$$\text{where: } \frac{\partial I_i}{\partial h} \equiv I_i \otimes \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} \approx B_i - A_i$$



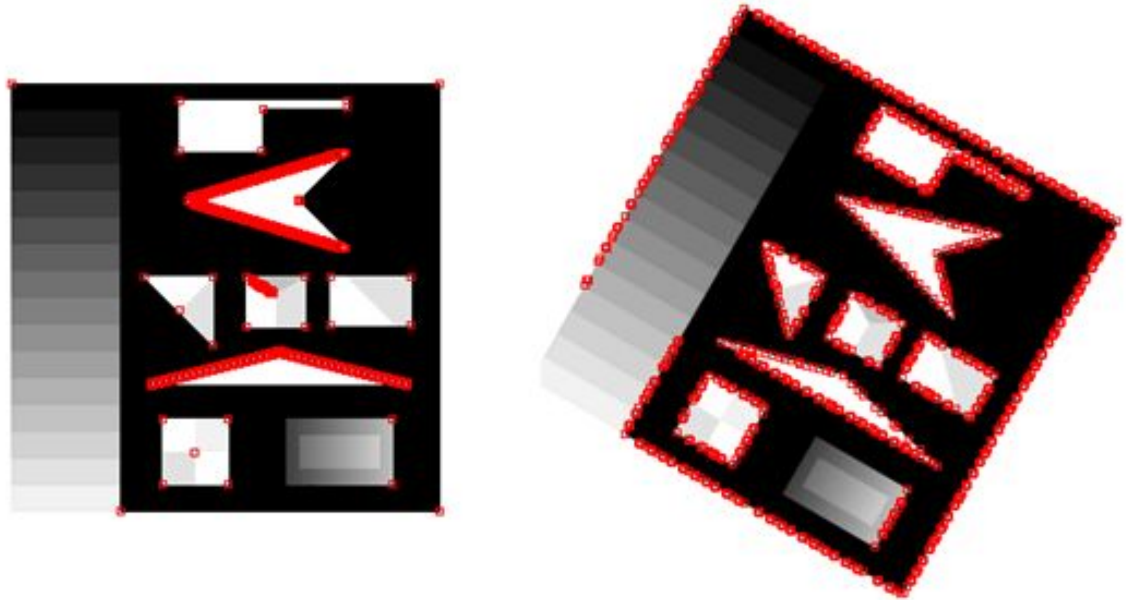
Vertical Moravec Intensity Variation:

$$V_y = \sum_{i=1}^{9}(A_i - B_i)^2 = \sum_{i=1}^{9}(B_i - A_i)^2 \approx \sum_{i=1}^{9}\left(\frac{\partial I_i}{\partial y}\right)^2$$

$$\text{where: } \frac{\partial I_i}{\partial y} \equiv I_i \otimes (-1,0,1)^T \approx B_i - A_i$$

Low Level Features (Corners)

- Moravec Operator:
  - Rotation sensitive



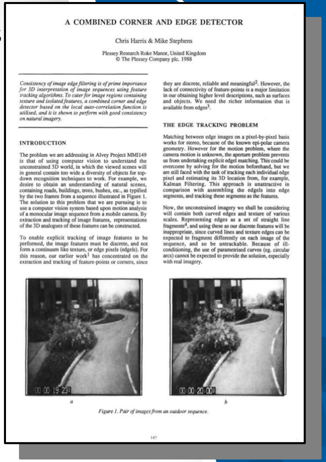
Original Image

Image Rotated 30°

Low Level Features (Corners)

- Harris and Plessey Operator
  - Described Chris Harris e Mike Stephens in 1988.
  - Based on Moravec Operator.
  - Less sensitive to rotation.

Chris Harris and Mike Stephens (1988).
A combined corner and edge detector.
Plessey Research Roke Manor.



A COMBINED CORNER AND EDGE DETECTOR

Chris Harris & Mike Stephens

Plessey Research Roke Manor, United Kingdom
© The Plessey Company plc. 1988

Consistency of image edge filtering is of prime importance for 3D interpretation of image sequences using feature tracking algorithms. To cater for image regions containing texture and isolated features, a combined corner and edge detector based on the local auto-correlation function is utilised, and it is shown to perform with good consistency on natural imagery.

INTRODUCTION

The problem we are addressing in Alvey Project MMI149 is that of using computer vision to understand the unconstrained 3D world, in which the viewed scenes will in general contain too wide a diversity of objects for top-down recognition techniques to work. For example, we desire to obtain an understanding of natural scenes, containing roads, buildings, trees, bushes, etc., as typified by the two frames from a sequence illustrated in Figure 1. The solution to this problem that we are pursuing is to use a computer vision system based upon motion analysis of a monocular image sequence from a mobile camera. By extraction and tracking of image features, representations of the 3D analogues of these features can be constructed.

To enable explicit tracking of image features to be performed, the image features must be discrete, and not form a continuum like texture, or edge pixels (edgels). For this reason, our earlier work[1] has concentrated on the extraction and tracking of feature-points or corners, since they are discrete, reliable and meaningful[2]. However, the lack of connectivity of feature-points is a major limitation in our obtaining higher level descriptions, such as surfaces and objects. We need the richer information that is available from edges[3].

THE EDGE TRACKING PROBLEM

Matching between edge images on a pixel-by-pixel basis works for stereo, because of the known epi-polar camera geometry. However for the motion problem, where the camera motion is unknown, the aperture problem prevents us from undertaking explicit edgel matching. This could be overcome by solving for the motion beforehand, but we are still faced with the task of tracking each individual edge pixel and estimating its 3D location from, for example, Kalman Filtering. This approach is unattractive in comparison with assembling the edgels into edge segments, and tracking these segments as the features.

Now, the unconstrained imagery we shall be considering will contain both curved edges and texture of various scales. Representing edges as a set of straight line fragments[4], and using these as our discrete features will be inappropriate, since curved lines and texture edges can be expected to fragment differently on each image of the sequence, and so be untrackable. Because of ill-conditioning, the use of parametrised curves (eg. circular arcs) cannot be expected to provide the solution, especially with real imagery.

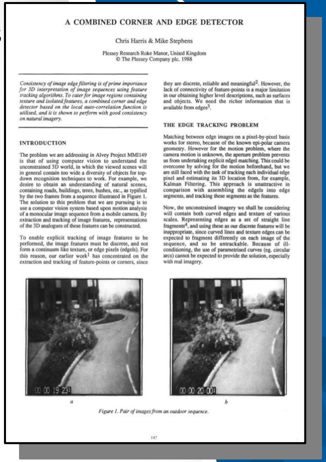
Figure 1. Pair of images from an outdoor sequence.

$$I(x+\Delta_x, y+\Delta_y) = I(x,y) + \overbrace{\frac{\partial I(x,y)}{\partial x}}^{G_x} \cdot \Delta_x + \overbrace{\frac{\partial I(x,y)}{\partial y}}^{G_y} \cdot \Delta_y + HOT(x,y)$$

Low Level Features (Corners)

- Harris and Plessey Operator
  - Described Chris Harris e Mike Stephens in 1988.
  - Based on Moravec Operator.
  - Less sensitive to rotation.

Chris Harris and Mike Stephens (1988). *A combined corner and edge detector*. Plessey Research Roke Manor.



A COMBINED CORNER AND EDGE DETECTOR

Chris Harris & Mike Stephens

Plessey Research Roke Manor, United Kingdom
© The Plessey Company plc. 1988

Consistency of image edge filtering is of prime importance for 3D interpretation of image sequences using feature tracking algorithms. To cater for image regions containing texture and isolated features, a combined corner and edge detector based on the local auto-correlation function is utilised, and it is shown to perform with good consistency on natural imagery.

INTRODUCTION

The problem we are addressing in Alvey Project MMI149 is that of using computer vision to understand the unconstrained 3D world, in which the viewed scenes will in general contain too wide a diversity of objects for top-down recognition techniques to work. For example, we desire to obtain an understanding of natural scenes, containing roads, buildings, trees, bushes, etc., as typified by the two frames from a sequence illustrated in Figure 1. The solution to this problem that we are pursuing is to use a computer vision system based upon motion analysis of a monocular image sequence from a mobile camera. By extraction and tracking of image features, representations of the 3D analogues of these features can be constructed.

To enable explicit tracking of image features to be performed, the image features must be discrete, and not form a continuum like texture, or edge pixels (edgels). For this reason, our earlier work[1] has concentrated on the extraction and tracking of feature-points or corners, since they are discrete, reliable and meaningful[2]. However, the lack of connectivity of feature-points is a major limitation in our obtaining higher level descriptions, such as surfaces and objects. We need the richer information that is available from edges[3].

THE EDGE TRACKING PROBLEM

Matching between edge images on a pixel-by-pixel basis works for stereo, because of the known epi-polar camera geometry. However for the motion problem, where the camera motion is unknown, the aperture problem prevents us from undertaking explicit edgel matching. This could be overcome by solving for the motion beforehand, but we are still faced with the task of tracking each individual edge pixel and estimating its 3D location from, for example, Kalman Filtering. This approach is unattractive in comparison with assembling the edgels into edge segments, and tracking these segments as the features.

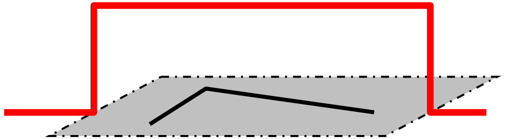
Now, the unconstrained imagery we shall be considering will contain both curved edges and texture of various scales. Representing edges as a set of straight line fragments[4], and using these as our discrete features will be inappropriate, since curved lines and texture edges can be expected to fragment differently on each image of the sequence, and so be untrackable. Because of ill-conditioning, the use of parametrised curves (eg. circular arcs) cannot be expected to provide the solution, especially with real imagery.

Figure 1. Pair of images from an outdoor sequence.

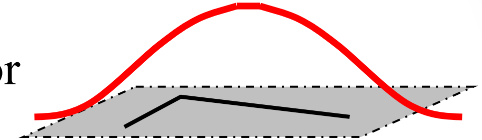
147

$$c(x,y,\Delta x,\Delta y) = \sum_{(x,y)\in W} w(x,y)\cdot\Bigg(\overbrace{I(x+\Delta x, y+\Delta y)}^{\text{Shiffted Image}} - \overbrace{I(x,y)}^{\text{Original Image}}\Bigg)^2$$

Window function $w(x,y) =$



or



1 in window, 0 outside

Gaussian

Low Level Features (Corners)

- Harris and Plessey Operator
  - Described Chris Harris e Mike Stephens in 1988.
  - Based on Moravec Operator.
  - Less sensitive to rotation.

$$c(x, y, \Delta x, \Delta y) = \sum_{(x,y)\in W} w(x, y) \cdot \left(\overbrace{I(x + \Delta x, y + \Delta y)}^{\text{Shiffted Image}} - \overbrace{I(x, y)}^{\text{Original Image}} \right)^2$$

  - The shift function can be approximated by Taylor expansion.

$$I(x + \Delta x, y + \Delta y) = I(x, y) + \overbrace{\frac{\partial I(x, y)}{\partial x}}^{G_x} \cdot \Delta x + \overbrace{\frac{\partial I(x, y)}{\partial y}}^{G_y} \cdot \Delta y + \overbrace{H.O.T.}^{\text{High order terms}}$$

$$\approx I(x, y) + \begin{bmatrix} G_x & G_y \end{bmatrix} \cdot \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}$$

Chris Harris and Mike Stephens (1988). *A combined corner and edge detector*. Plessey Research Roke Manor.

Low Level Features (Corners)

- Harris and Plessey Operator
  - Given: $$c(x,y,\Delta x,\Delta y) = \sum_{(x,y)\in W} w(x,y)\cdot\left(\overbrace{I(x+\Delta x, y+\Delta y)}^{\text{Shiffted Image}} - \overbrace{I(x,y)}^{\text{Original Image}}\right)^2$$
  - $$I(x+\Delta x, y+\Delta y) \approx I(x,y) + \begin{bmatrix} G_x & G_y \end{bmatrix}\cdot\begin{bmatrix}\Delta x \\ \Delta y\end{bmatrix}$$
  - Simplifying the equations, we have:

$$c(x,y,\Delta x,\Delta y) \approx \sum_{(x,y)\in W} w(x,y)\cdot\left(I(x,y) + \begin{bmatrix} G_x & G_y \end{bmatrix}\cdot\begin{bmatrix}\Delta x \\ \Delta y\end{bmatrix} - I(x,y)\right)^2$$

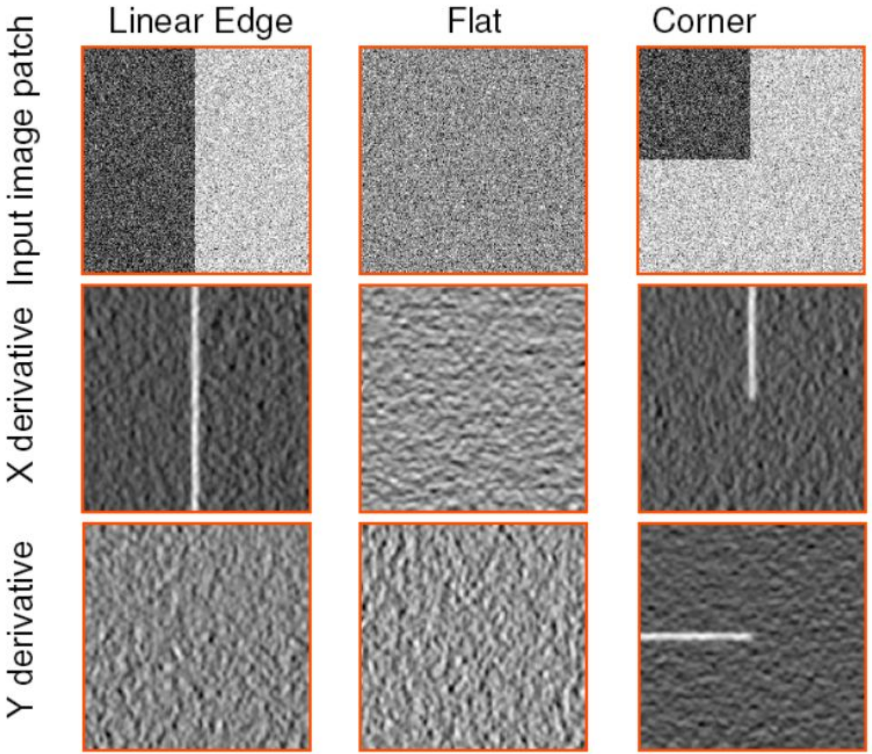
$$c(x,y,\Delta x,\Delta y) \approx \begin{bmatrix}\Delta x & \Delta y\end{bmatrix}\cdot\left(\sum_{(x,y)\in W} w(x,y)\cdot\begin{bmatrix} G_x^2 & G_xG_y \\ G_yG_x & G_y^2\end{bmatrix}\right)\cdot\begin{bmatrix}\Delta x \\ \Delta y\end{bmatrix}$$

Low Level Features (Corners)

- Harris and Plessey Operator
  - Given:
  -
  - Simplifying the equations, we have:

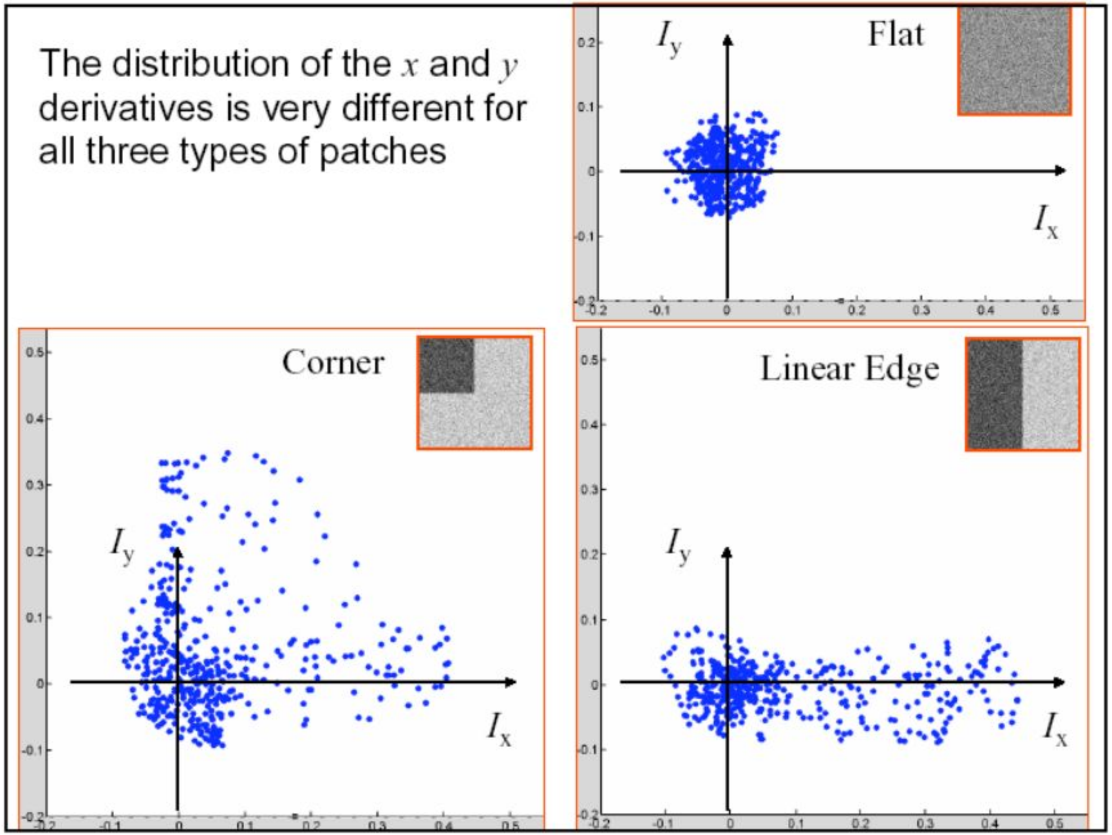
Low Level Features (Corners)

- Harris and Plessey Operator
  - Image gradients



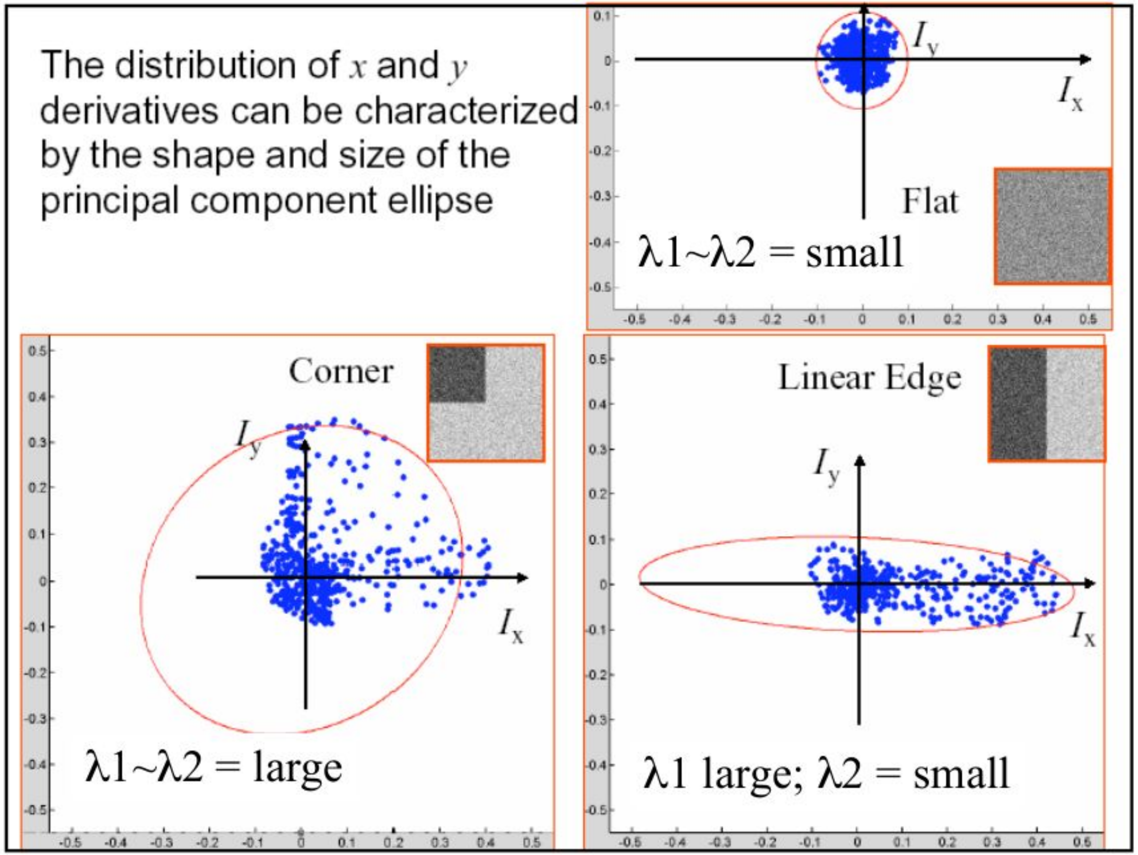
Low Level Features (Corners)

- Harris and Plessey Operator
  - Image gradients
  - Plotting derivatives



Low Level Features (Corners)

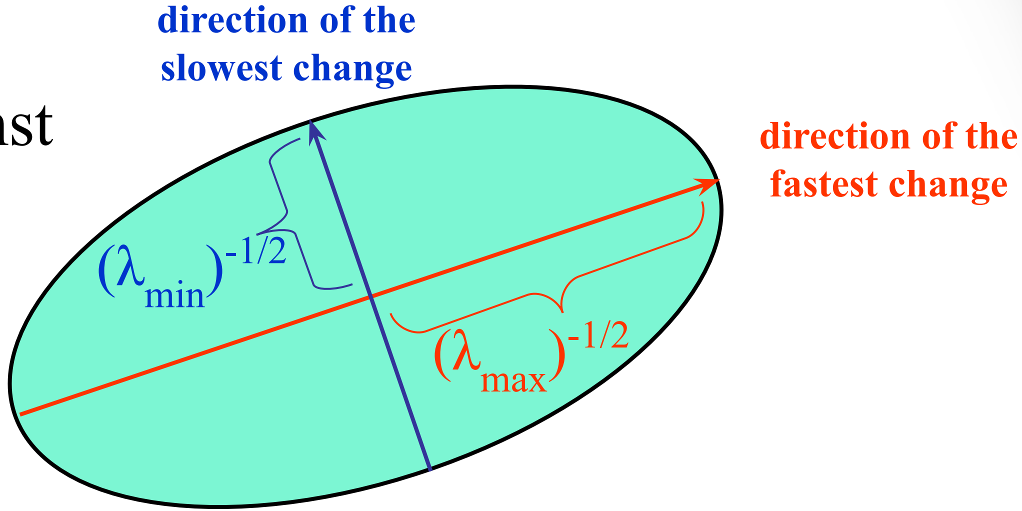
- Harris and Plessey Operator
  - Image gradients
  - Plotting derivatives
  - Fitting ellipses



Low Level Features (Corners)

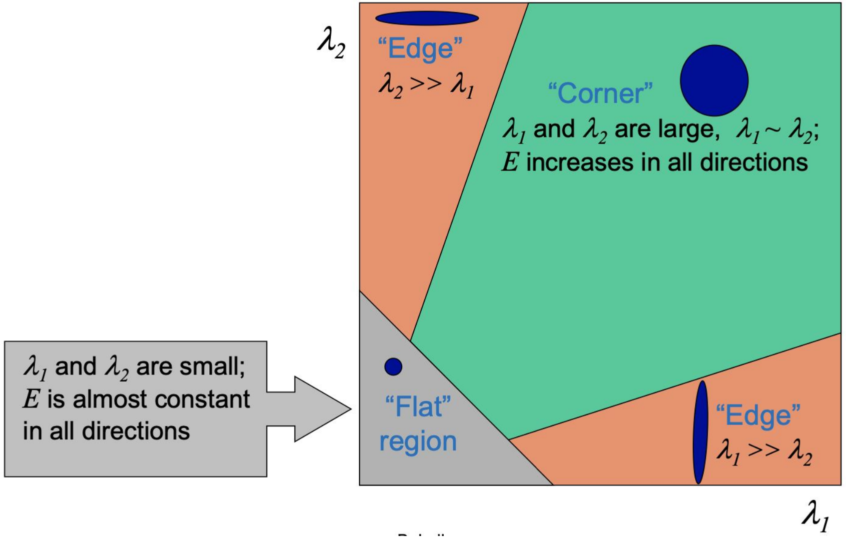
- Harris and Plessey Operator
  - Image gradients
  - Plotting derivatives
  - Fitting ellipses

Ellipse $E(u,v) = \text{const}$



Low Level Features (Corners)

- Harris and Plessey Operator
  - The eigenvalues determine the elongation of covariance ellipse.
  - The eigenvectors determine the slope of covariance ellipse.
  - k is empirically defined in interval: $$0.04 \leq k \leq 0.06$$



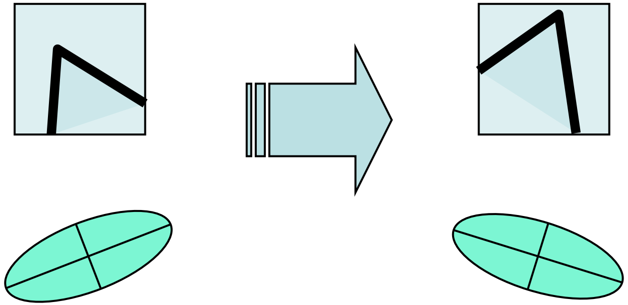
Harris detector: Steps

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

C.Harris and M.Stephens. "A Combined Corner and Edge Detector.“ *Proceedings of the 4th Alvey Vision Conference*: pages 147—151, 1988.

Low Level Features (Corners)

- Harris and Plessey Operator



- Ellipse rotates but its shape (i.e. eigenvalues) remains the same

Corner response R is invariant to image rotation

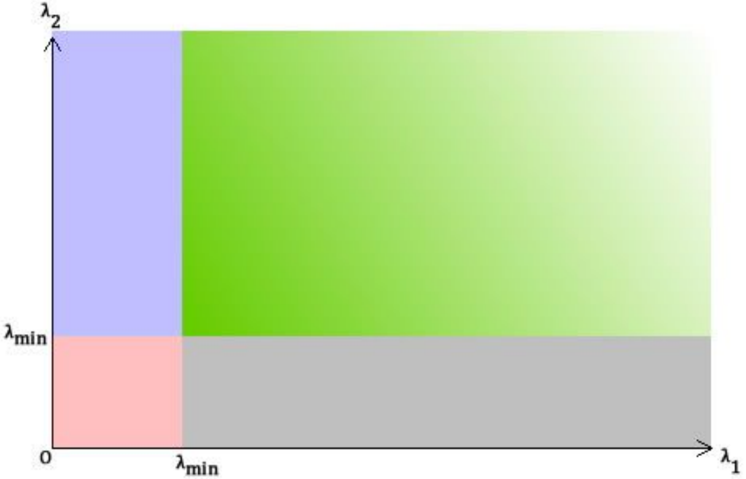
Low Level Features (Corners)

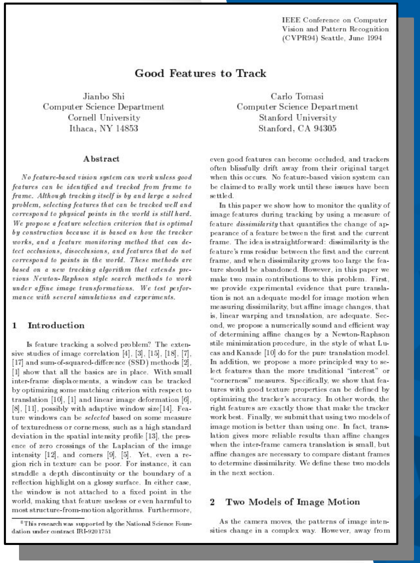
- Harris and Plessey Operator

Demonstration (MATLAB)

Low Level Features (Corners)

- Shi-Tomasi Operator
  - They propose an alternative computation of corner response
  - $$R(x, y) = \min(\lambda_1, \lambda_2)$$
  - The eigenvalues have to be greater than λ_{min}





IEEE Conference on Computer Vision and Pattern Recognition (CVPR94) Seattle, June 1994

Good Features to Track

Jianbo Shi
Computer Science Department
Cornell University
Ithaca, NY 14853

Carlo Tomasi
Computer Science Department
Stanford University
Stanford, CA 94305

Abstract

No feature-based vision system can work unless good features can be identified and tracked from frame to frame. Although tracking itself is by and large a solved problem, selecting features that can be tracked well and correspond to physical points in the world is still hard. We propose a feature selection criterion that is optimal by construction because it is based on how the tracker works, and a feature monitoring method that can detect occlusions, disocclusions, and features that do not correspond to points in the world. These methods are based on a new tracking algorithm that extends previous Newton-Raphson style search methods to work under affine image transformations. We test performance with several simulations and experiments.

1 Introduction

Is feature tracking a solved problem? The extensive studies of image correlation [4], [3], [15], [18], [7], [17] and sum-of-squared-difference (SSD) methods [2], [1] show that all the basics are in place. With small inter-frame displacements, a window can be tracked by optimizing some matching criterion with respect to translation [10], [1] and linear image deformation [6], [8], [11], possibly with adaptive window size[14]. Feature windows can be *selected* based on some measure of texturedness or cornerness, such as a high standard deviation in the spatial intensity profile [13], the presence of zero crossings of the Laplacian of the image intensity [12], and corners [9], [5]. Yet, even a region rich in texture can be poor. For instance, it can straddle a depth discontinuity or the boundary of a reflection highlight on a glossy surface. In either case, the window is not attached to a fixed point in the world, making that feature useless or even harmful to most structure-from-motion algorithms. Furthermore, even good features can become occluded, and trackers often blissfully drift away from their original target when this occurs. No feature-based vision system can be claimed to really work until these issues have been settled.

In this paper we show how to monitor the quality of image features during tracking by using a measure of feature *dissimilarity* that quantifies the change of appearance of a feature between the first and the current frame. The idea is straightforward: dissimilarity is the feature's rms residue between the first and the current frame, and when dissimilarity grows too large the feature should be abandoned. However, in this paper we make two main contributions to this problem. First, we provide experimental evidence that pure translation is not an adequate model for image motion when measuring dissimilarity, but affine image changes, that is, linear warping and translation, are adequate. Second, we propose a numerically sound and efficient way of determining affine changes by a Newton-Raphson stile minimization procedure, in the style of what Lucas and Kanade [10] do for the pure translation model. In addition, we propose a more principled way to select features than the more traditional "interest" or "cornerness" measures. Specifically, we show that features with good texture properties can be defined by optimizing the tracker's accuracy. In other words, the right features are exactly those that make the tracker work best. Finally, we submit that using two models of image motion is better than using one. In fact, translation gives more reliable results than affine changes when the inter-frame camera translation is small, but affine changes are necessary to compare distant frames to determine dissimilarity. We define these two models in the next section.

2 Two Models of Image Motion

As the camera moves, the patterns of image intensities change in a complex way. However, away from

[0] This research was supported by the National Science Foundation under contract IRI-9201751

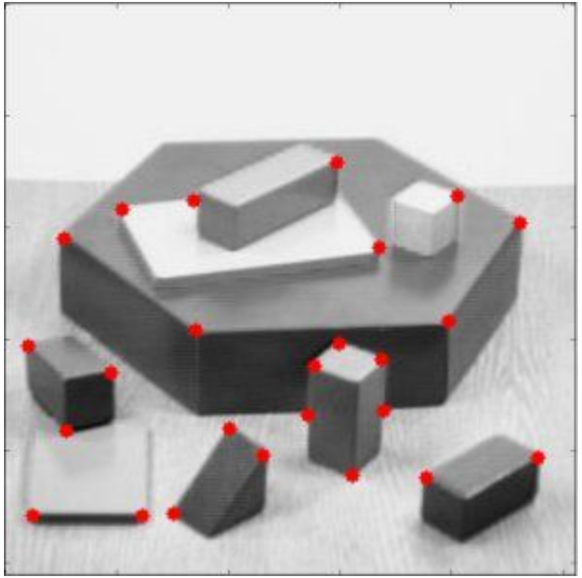
Jianbo Shi and Carlo Tomasi (1994). *Goog Features*
To Track. Computer Vision and Pattern Recognition
(CVPR'94)

Low Level Features (Corners)

- OpenCV function: goodFeaturesToTrack()
  - void goodFeaturesToTrack(
    - InputArray image,
    - OutputArray corners,
    - int maxCorners = 100,
    - double qualityLevel = 0.01,
    - double minDistance = 10,
    - InputArray mask=noArray(),
    - int blockSize=3,
    - bool useHarrisDetector = false,
    - double k=0.04)

Low Level Features (Corners)

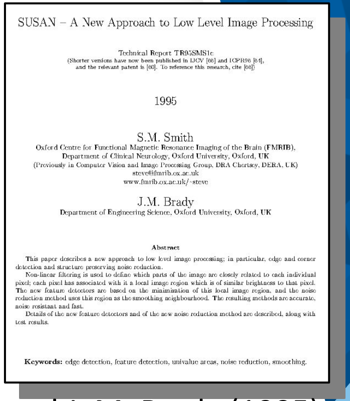
- OpenCV function: goodFeaturesToTrack()
  - void goodFeaturesToTrack(
    - InputArray image,
    - OutputArray corners,
    - int maxCorners = 100,
    - double qualityLevel = 0.01,
    - double minDistance = 10,
    - InputArray mask=noArray(),
    - int blockSize=3,
    - bool useHarrisDetector = false,
    - double k=0.04)



Low Level Features (Corners)

- Others corner detectors
- Smallest Univalue Segment Assimilating Nucleus (SUSAN)
  - Edge detector
  - Corner detector
  - Noise filter

Technical Report TR95SMS1c
(Shorter versions have now been published in IJCV [66] and ICPR96 [64], and the relevant patent is [60]. To reference this research, cite [66])


SUSAN – A New Approach to Low Level Image Processing

Technical Report TR95SMS1c
(Shorter versions have now been published in IJCV [66] and ICPR96 [64], and the relevant patent is [60]. To reference this research, cite [66])

1995

S.M. Smith
Oxford Centre for Functional Magnetic Resonance Imaging of the Brain (FMRIB),
Department of Clinical Neurology, Oxford University, Oxford, UK
(Previously in Computer Vision and Image Processing Group, DRA Chertsey, DERA, UK)
steve@fmrib.ox.ac.uk
www.fmrib.ox.ac.uk/~steve

J.M. Brady
Department of Engineering Science, Oxford University, Oxford, UK

Abstract

This paper describes a new approach to low level image processing; in particular, edge and corner detection and structure preserving noise reduction.

Non-linear filtering is used to define which parts of the image are closely related to each individual pixel; each pixel has associated with it a local image region which is of similar brightness to that pixel. The new feature detectors are based on the minimization of this local image region, and the noise reduction method uses this region as the smoothing neighbourhood. The resulting methods are accurate, noise resistant and fast.

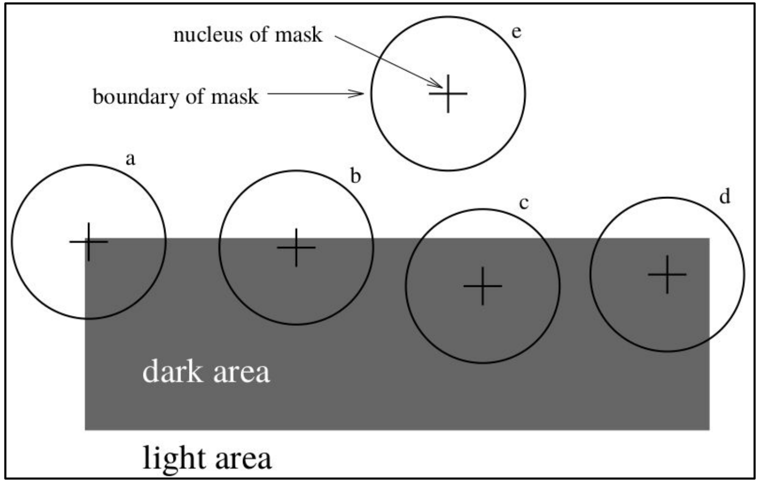
Details of the new feature detectors and of the new noise reduction method are described, along with test results.

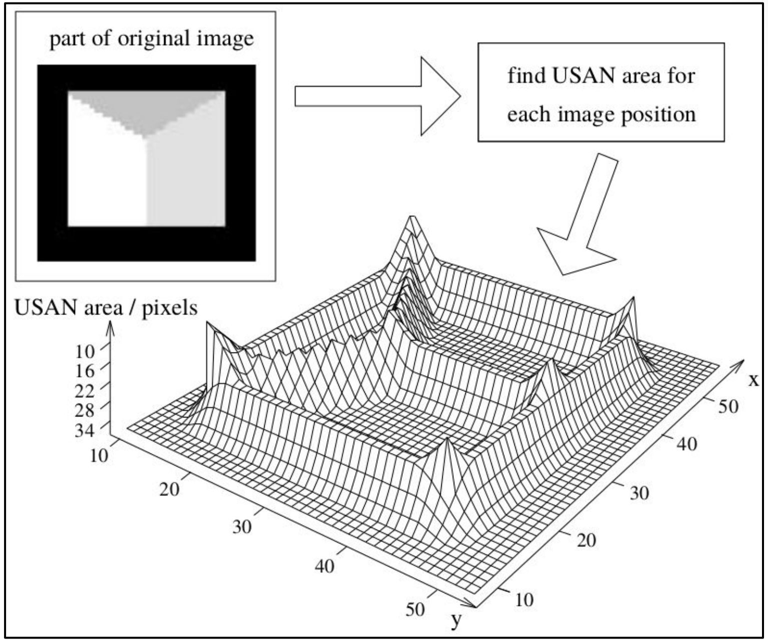
Keywords: edge detection, feature detection, univalue areas, noise reduction, smoothing.

S. M. Smith and J. M. Brady (1995)
SUSAN - A new Approach to Low Level Image Processing
Technical Report Oxford University

Low Level Features (Corners)

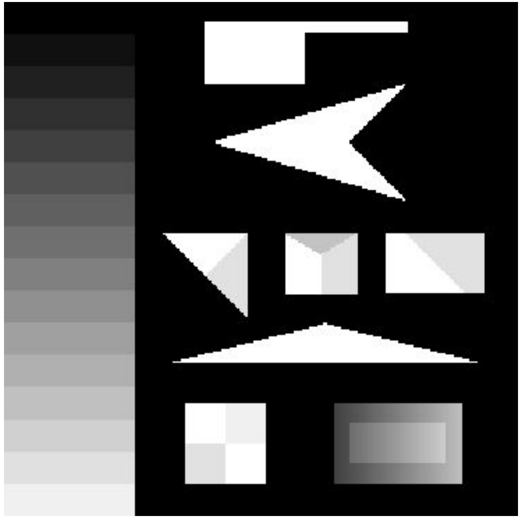
- Others corner detectors
- Smallest Univalue Segment Assimilating Nucleus (SUSAN)



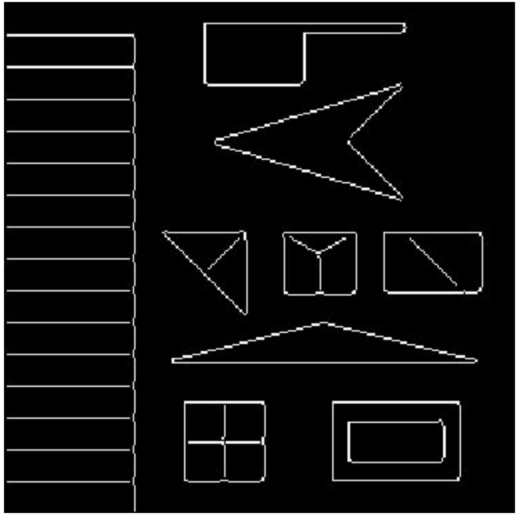


Low Level Features (Corners)

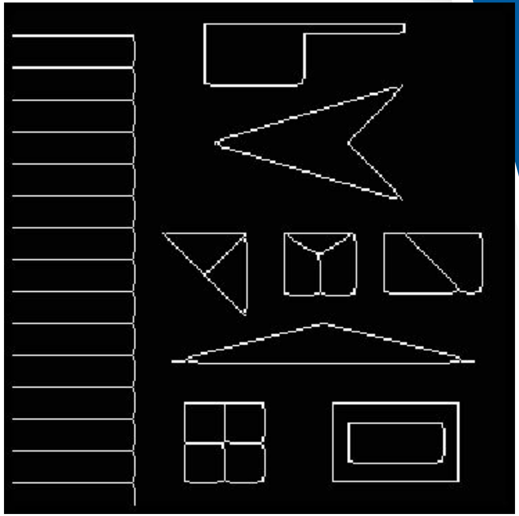
- Others corner detectors
- Smallest Univalue Segment Assimilating Nucleus (SUSAN)



Test
Image



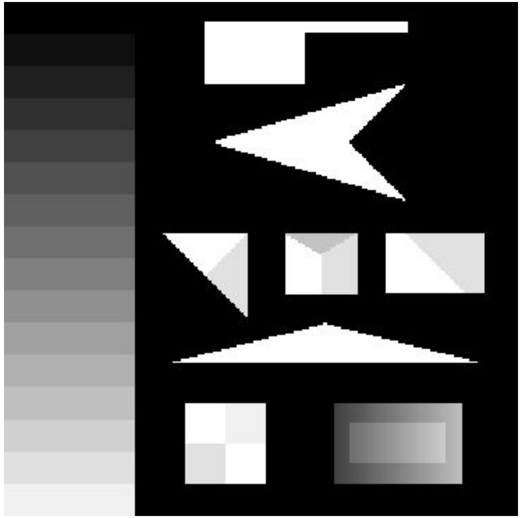
Canny edge
detector



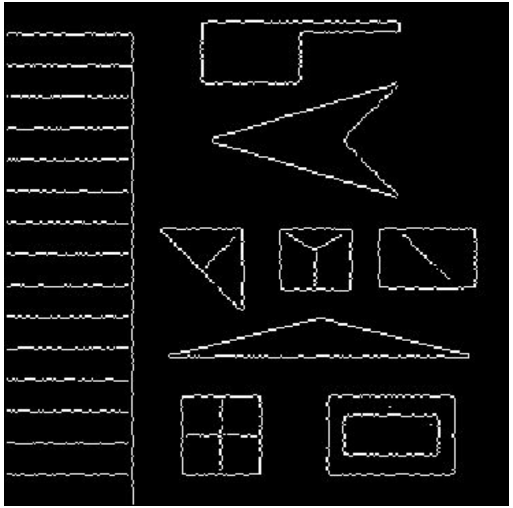
SUSAN edge
detector

Low Level Features (Corners)

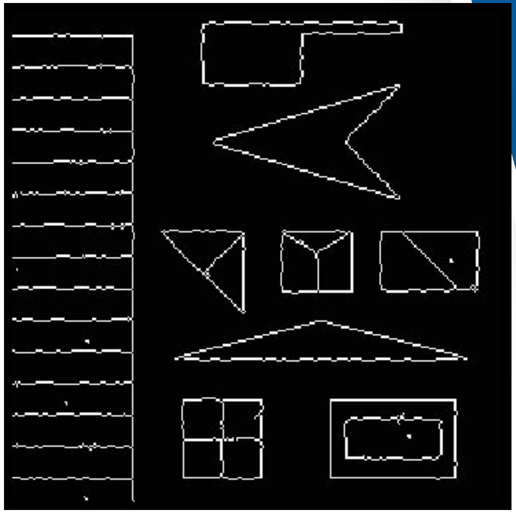
- Others corner detectors
- Smallest Univalue Segment Assimilating Nucleus (SUSAN)



Test
Image



Canny edge
detector
AWGN (5db)

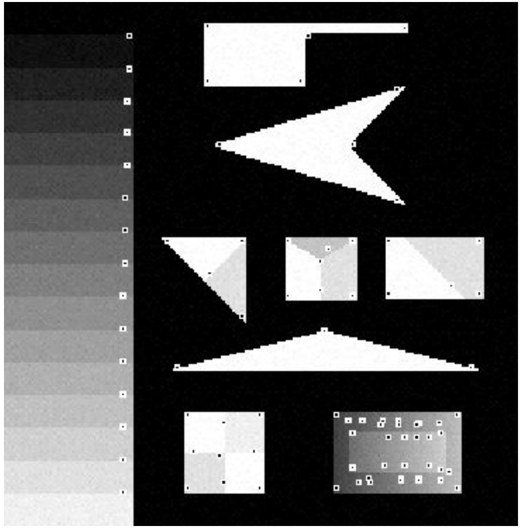


SUSAN edge
detector
AWGN (5db)

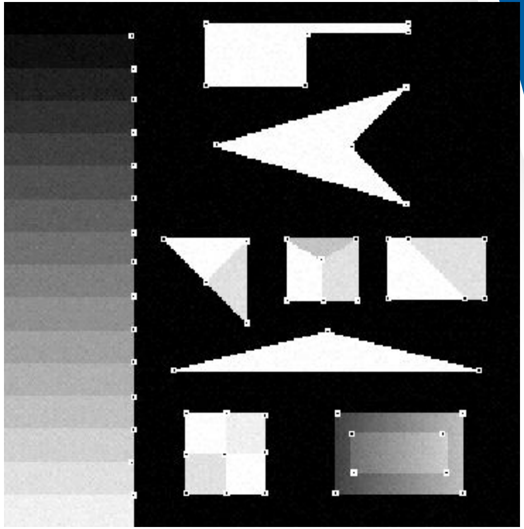
Low Level Features (Corners)

- Others corner detectors
- Smallest Univalue Segment Assimilating Nucleus (SUSAN)

Test
Image



Harris-Plessey corner
detector
AWGN (5db)



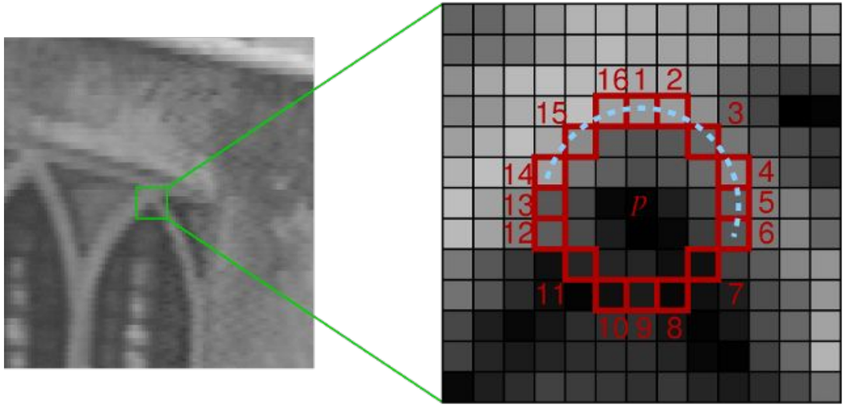
SUSAN corner
detector
AWGN (5db)

Low Level Features (Corners)

- Others corner detectors
- FASTER and better: A machine learning approach to corner detection
- (FAST Corner Detector)


Faster and better: a machine learning approach to corner detection

Edward Rosten, Reid Porter, and Tom Drummond



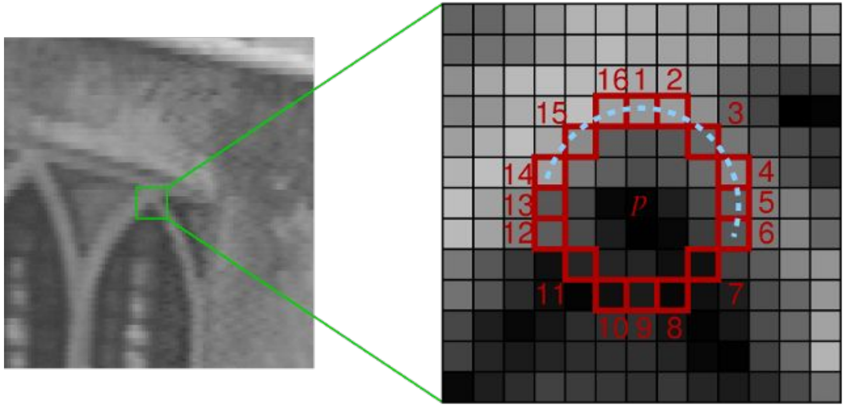
Eduard Rosten and Tom Drummond (2010)

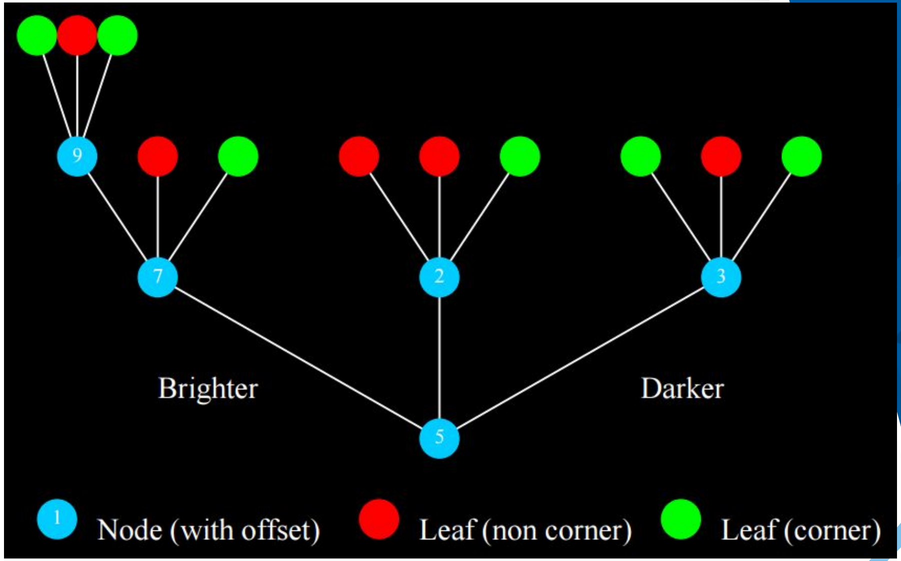
FASTER and better: A machine learning approach to corner detection

IEEE Trans. Pattern Analysis and Machine Intelligence

Low Level Features (Corners)

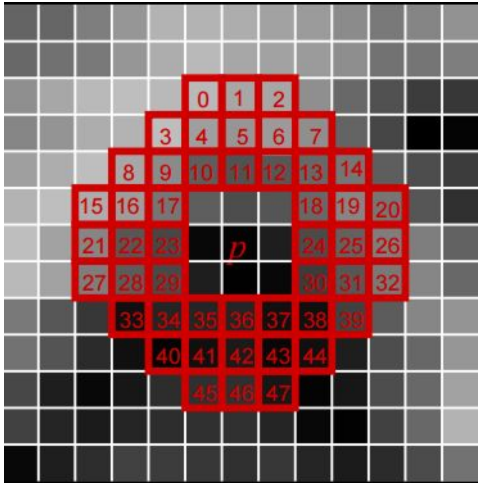
- Others corner detectors
- FASTER and better: A machine learning approach to corner detection
- (FAST Corner Detector)





Low Level Features (Corners)

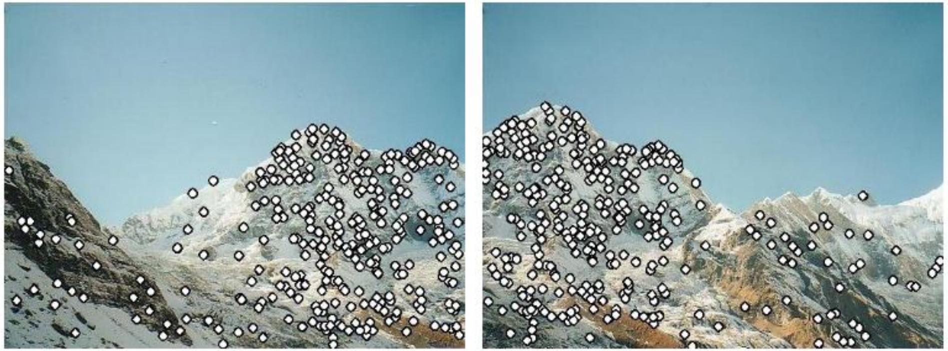
- Others corner detectors
- FASTER and better: A machine learning approach to corner detection
- (FAST-ER: Enhanced Repeatability)



Detector	Opteron 2.6GHz		Pentium III 850MHz	
	ms	%	ms	%
Fast $n = 9$ (non-max suppression)	1.33	6.65	5.29	26.5
Fast $n = 9$ (raw)	1.08	5.40	4.34	21.7
Fast $n = 12$ (non-max suppression)	1.34	6.70	4.60	23.0
Fast $n = 12$ (raw)	1.17	5.85	4.31	21.5
Original FAST $n = 12$ (non-max suppression)	1.59	7.95	9.60	48.0
Original FAST $n = 12$ (raw)	1.49	7.45	9.25	48.5
Harris	24.0	120	166	830
DoG	60.1	301	345	1280
SUSAN	7.58	37.9	27.5	137.5

Table 1. Timing results for a selection of feature detectors run on fields (768×288) of a PAL video sequence in milliseconds, and as a percentage of the processing budget per frame. Note that since PAL and NTSC, DV and 30Hz VGA (common for web-cams) have approximately the same pixel rate, the percentages are widely applicable. Approximately 500 features per field are detected.

Characteristics of good features

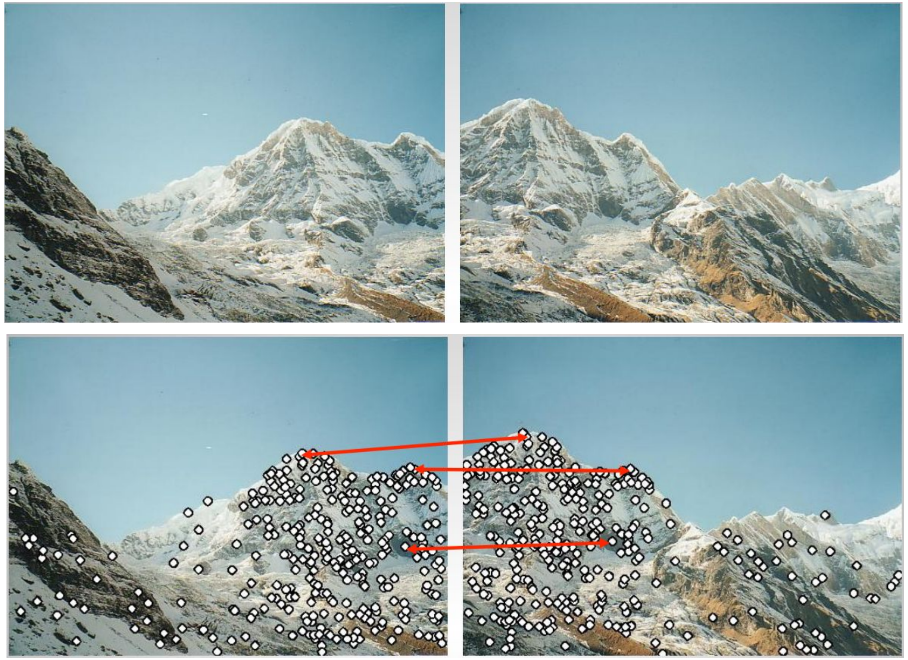


- Repeatability
  - The same feature can be found in several images despite geometric and photometric transformations
- Saliency
  - Each feature has a distinctive description
- Compactness and efficiency
  - Many fewer features than image pixels
- Locality
  - A feature occupies a relatively small area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

Why to use corners or features of images?

Why to use corners or features of images?

- **Example of motivation:**
  - Image registration



Why to use corners or features of images?

- **Example of motivation:**
  - Automate object tracking
  - Motion estimation
  - Optical flow

H. Badino and T. Kanade:
"A Head-Wearable Short-Baseline Stereo System for the Simultaneous Estimation of Structure and Motion".
IAPR Conference on Machine Vision Applications (MVA), Nara, Japan, June 2011.

Why to use corners or features of images?

- **Example of motivation:**
  - 3D Reconstruction
  - Augmented Reality

Richard Newcombe, Steven Lovegrove and Andrew Davison
DTAM: Dense Tracking and Mapping in Real-Time"
ICCV 2011.

Why to use corners or features of images?

- **List of motivations:**
  - Image registration
  - Automate object tracking
  - Point matching for computing disparity
  - Stereo calibration
    - Estimation of fundamental matrix
  - Motion based segmentation
  - Recognition
  - 3D object reconstruction
  - Robot navigation
  - Image retrieval and indexing

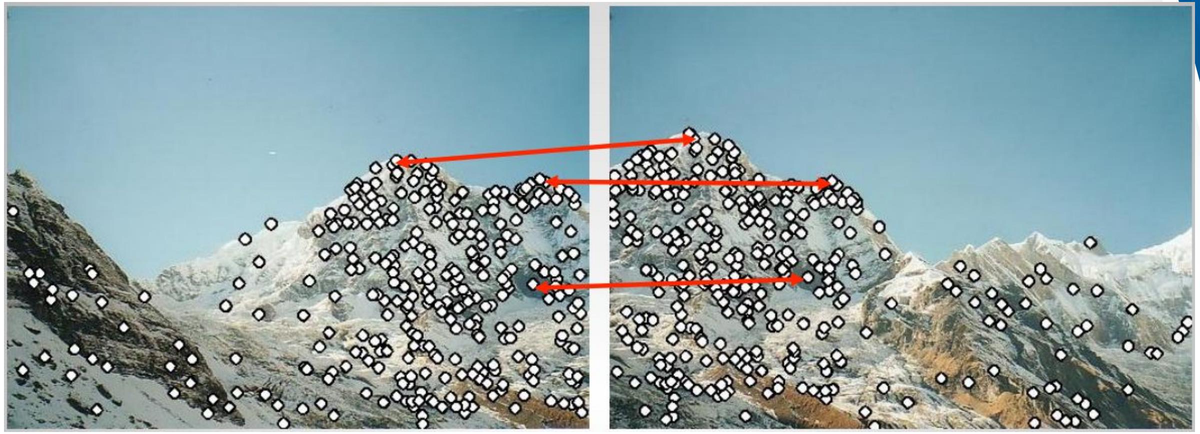
Main Concepts

- **Goal 1: interest operator repeatability**
  - We want to detect, at least some of, the same points in the images.
  - Yet we have to be able to run the detection procedure independently per image.
  - Ideal interest point operators must be invariant to:
    - Noise
    - Illumination
    - Translation
    - Rotation
    - Scale

No chance to find true matches!

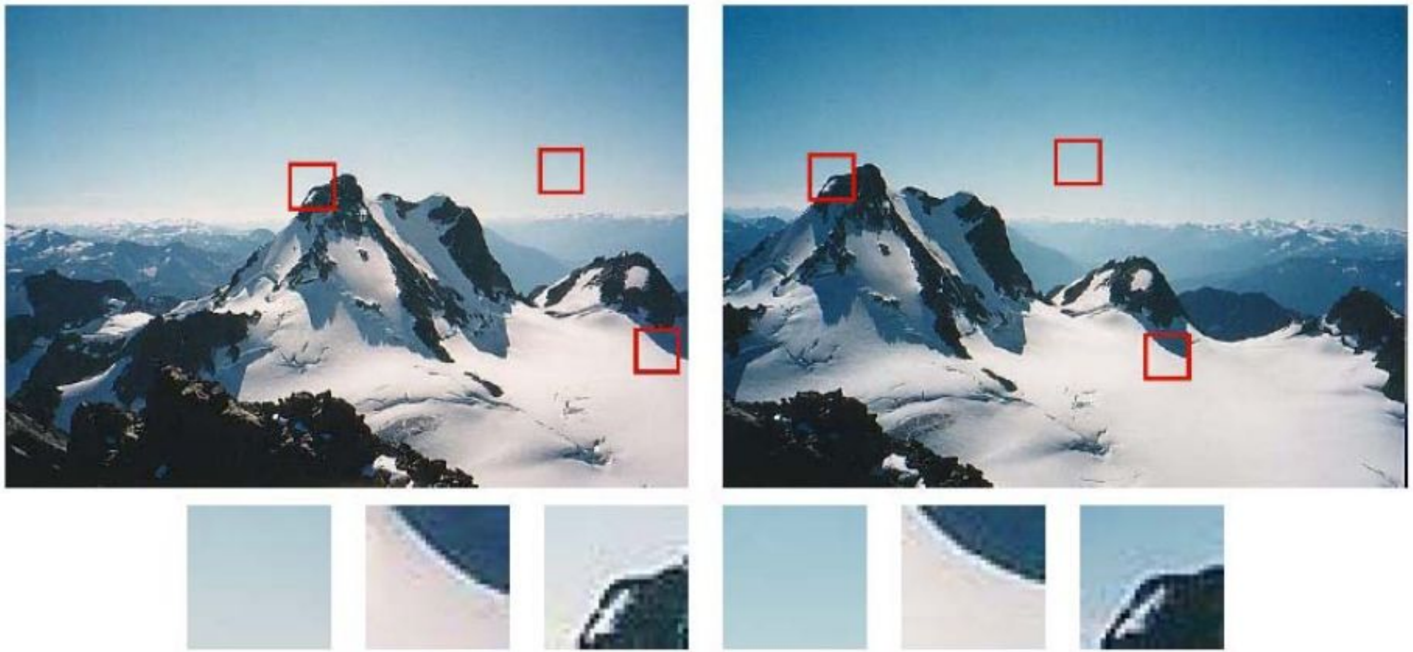
Main Concepts

- **Goal 2: descriptor distinctiveness**
  - We want to be able to reliably determine which point goes with which (Matches).
  - Must provide some invariance to geometric and photometric differences between the two views.
  - Ideal descriptors must be robust to:
    - Noise
    - Illumination
    - Translation
    - Rotation
    - Scale
    - Perspective
    - Partial occlusion



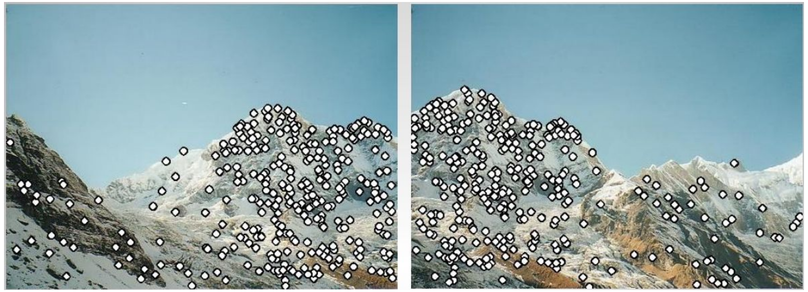
Main Concepts

- **Goal 2: descriptor distinctiveness**
  - Some patches can be matched with high accuracy than others.

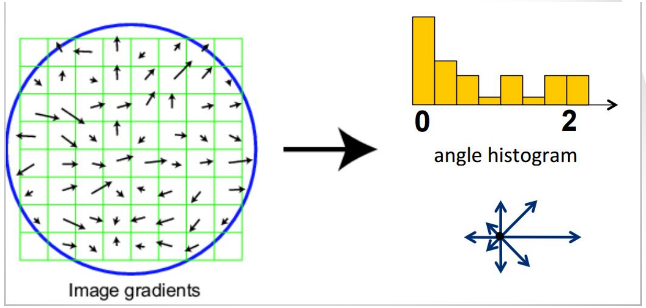


How to use interest points?

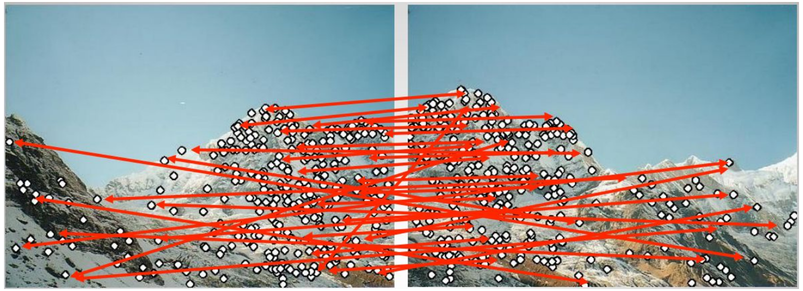
1) Detection



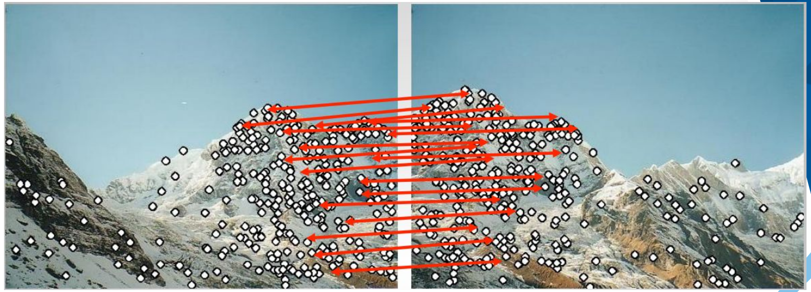
2) Description



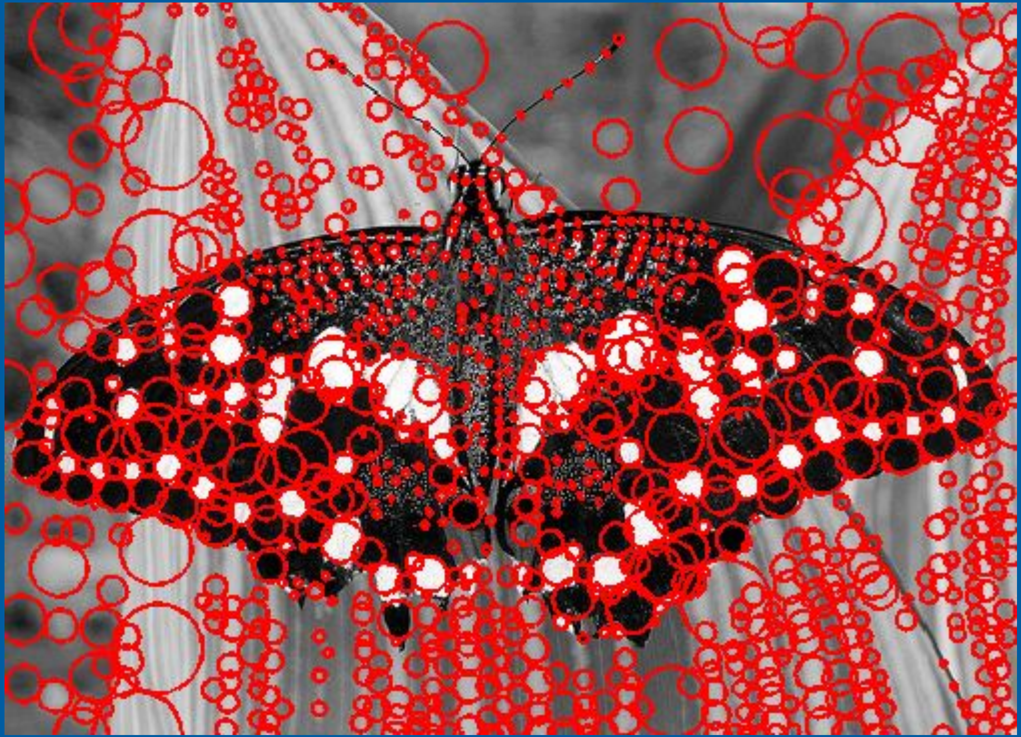
3) Matching



4) Filtering

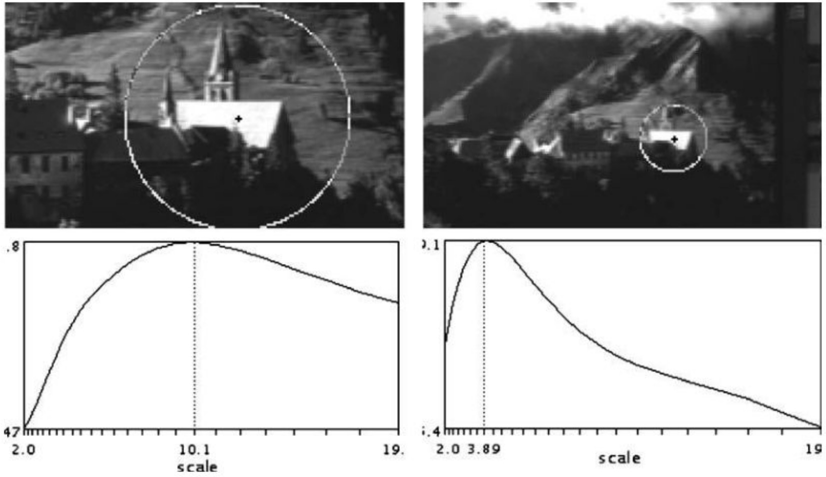


Blob Detection



Blob detection with scale selection

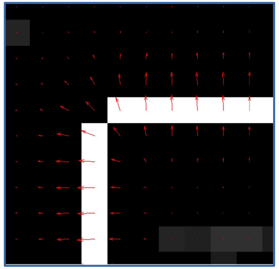
- Goal: independently detect corresponding regions in scaled versions of the same image
- Need *scale selection* mechanism for finding characteristic region size that is *covariant* with the image transformation

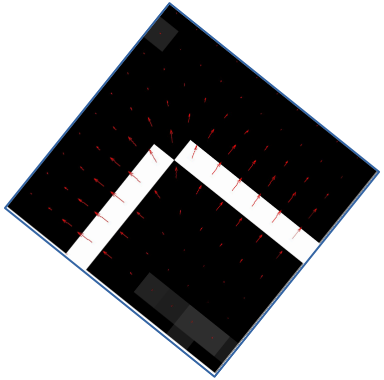


Recap Harris

Harris & Stephens (1988)

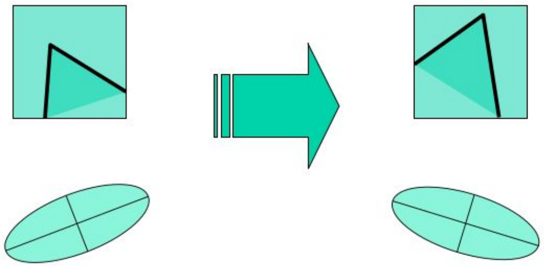
$$\det(C) - \kappa \mathrm{trace}^2(C)$$





Recap Harris

Properties: Rotational Invariance



Ellipse rotates but its shape
(**eigenvalues**) remains the same

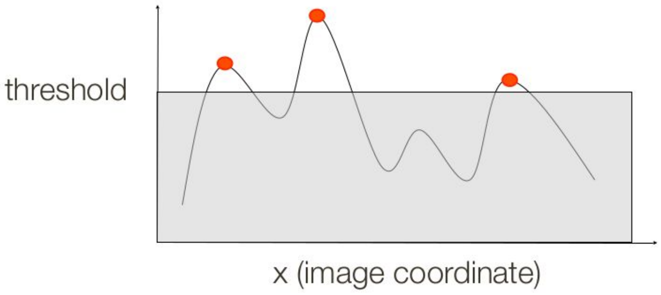
Corner response is **invariant** to image rotation

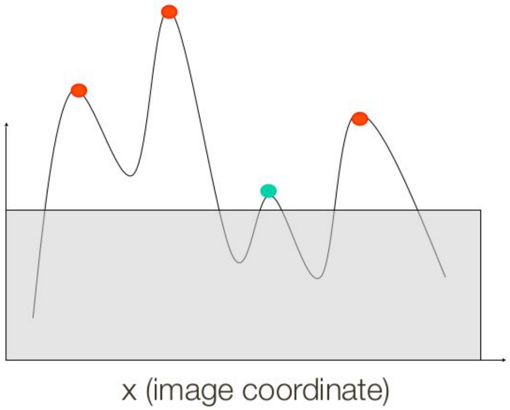
Recap Harris

Properties: (partial) Invariance to Intensity Shifts and Scaling

Only derivatives are used -> Invariance to intensity shifts

Intensity scale could effect performance



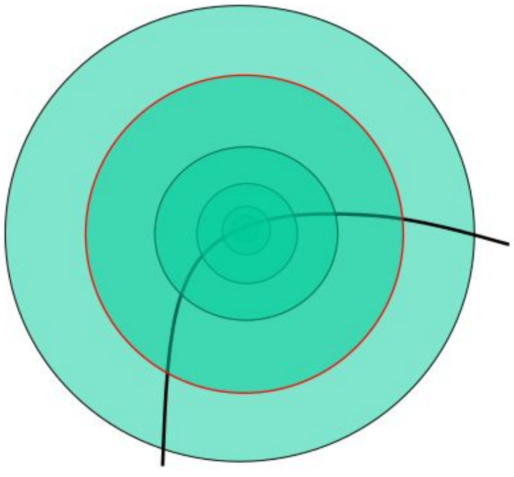
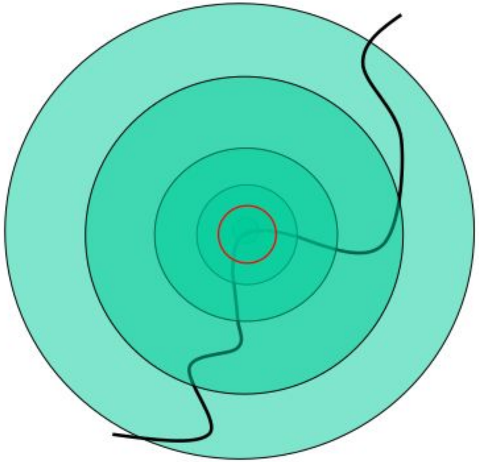


Recap Harris

Properties: NOT Invariant to Scale Changes

How to deal with scale?

Find local maxima in both, **position** and **scale**.

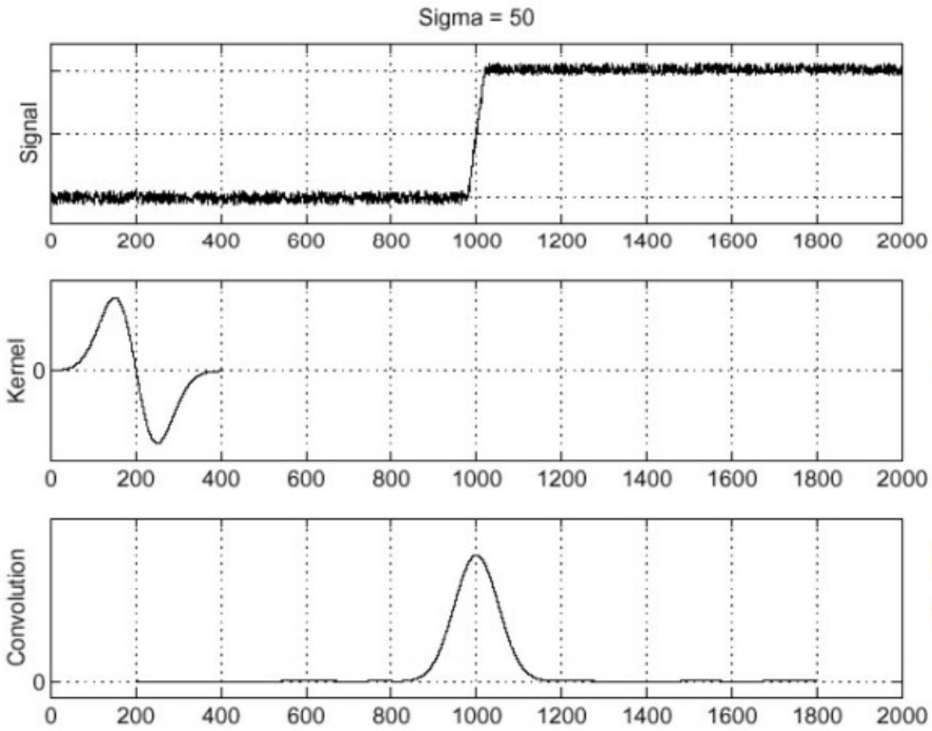
 

How to deal with scale?

f

$\frac{d}{dx} g$

$f * \frac{d}{dx} g$

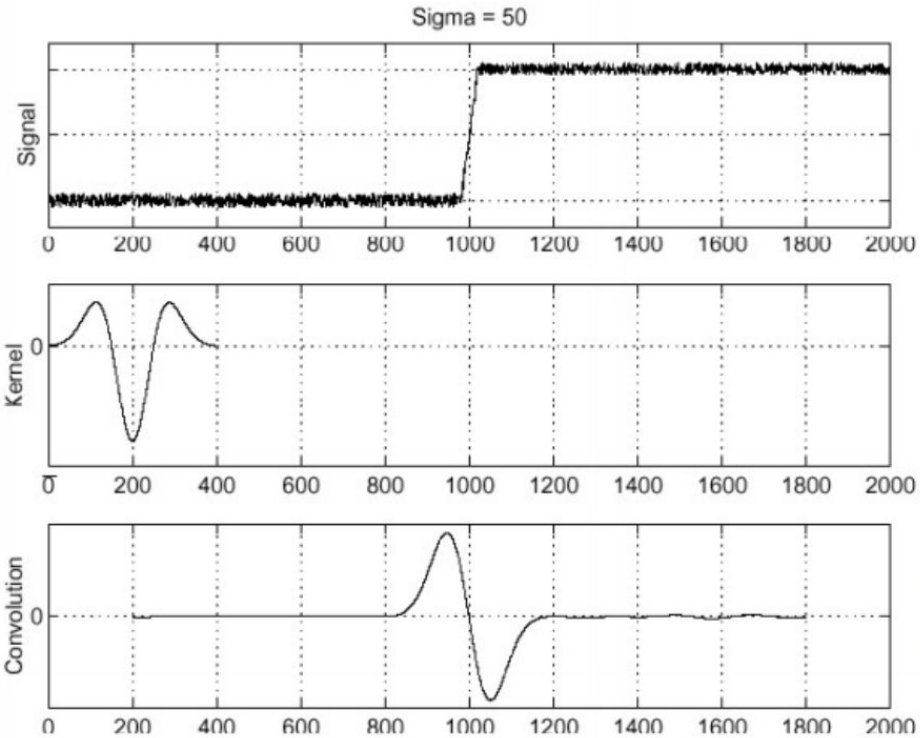


Edge

Derivative of Gaussian

Edge = maximum of derivative

How to deal with scale?



f

Edge

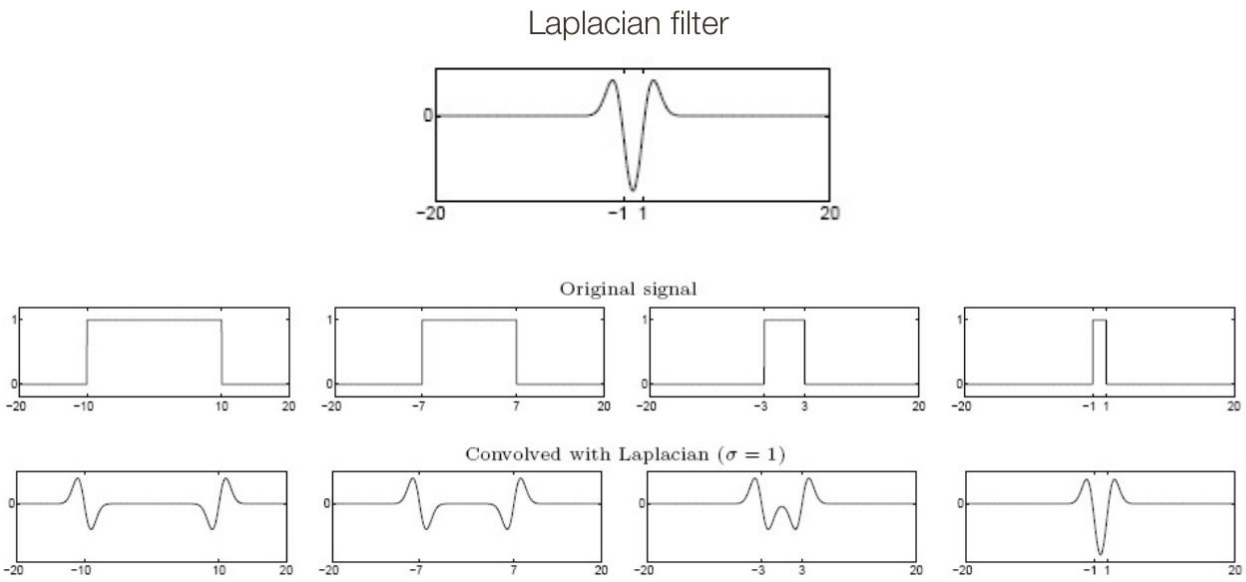
$\frac{d^2}{dx^2} g$

Second derivative of Gaussian (Laplacian)

$f * \frac{d^2}{dx^2} g$

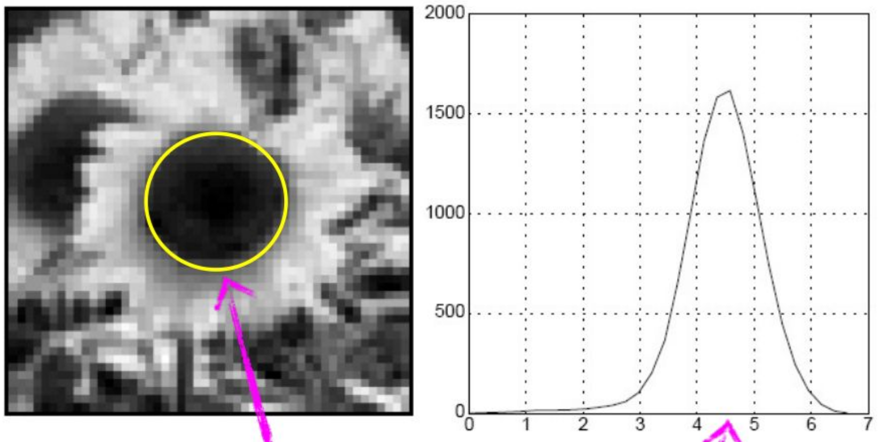
Edge = zero crossing of second derivative

How to deal with scale?



Highest response when the signal has the same **characteristic scale** as the filter

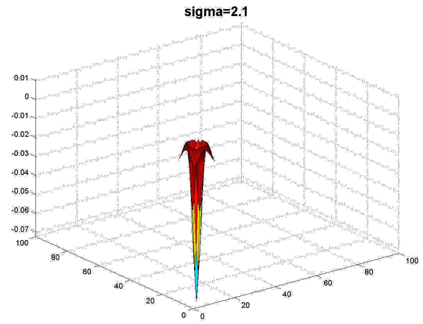
characteristic scale - the scale that produces peak filter response



characteristic scale

we need to search over characteristic scales

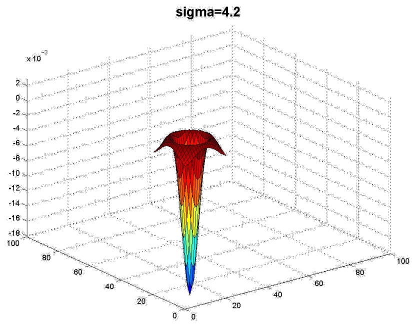
Applying **Laplacian** Filter at Different **Scales**



jet color scale
blue: low, red: high

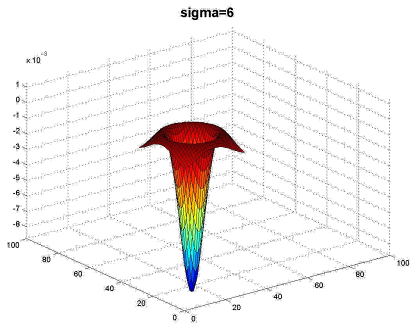
Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

Applying **Laplacian** Filter at Different **Scales**



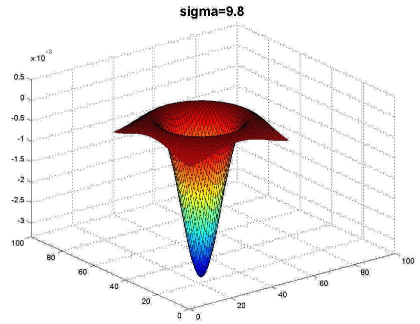
Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

Applying **Laplacian** Filter at Different **Scales**



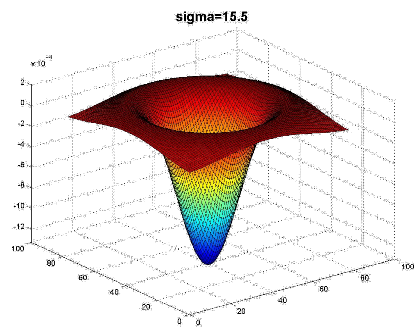
Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

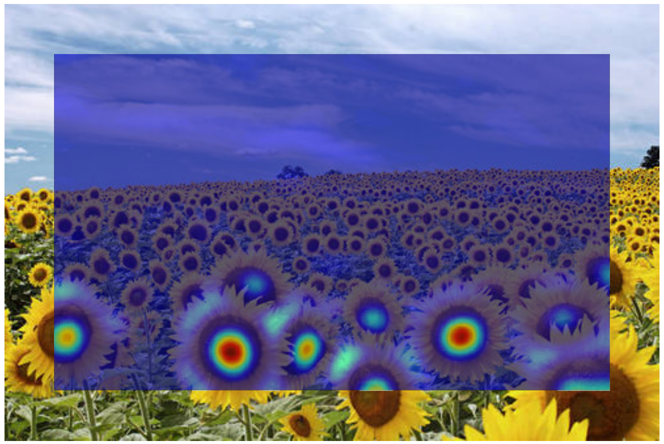
Applying **Laplacian** Filter at Different **Scales**



Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

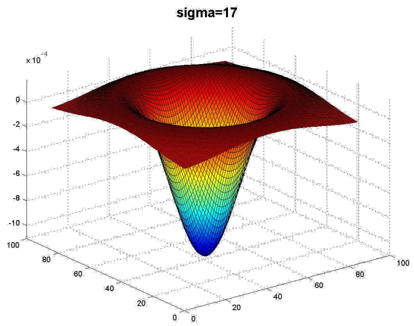
Applying **Laplacian** Filter at Different **Scales**

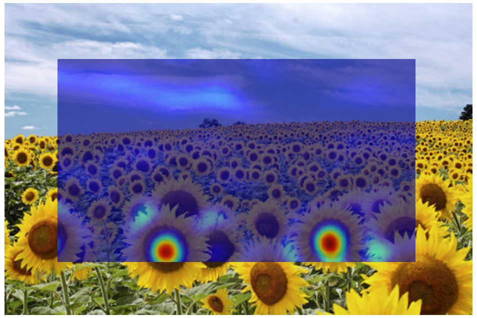




Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

Applying **Laplacian** Filter at Different **Scales**





Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

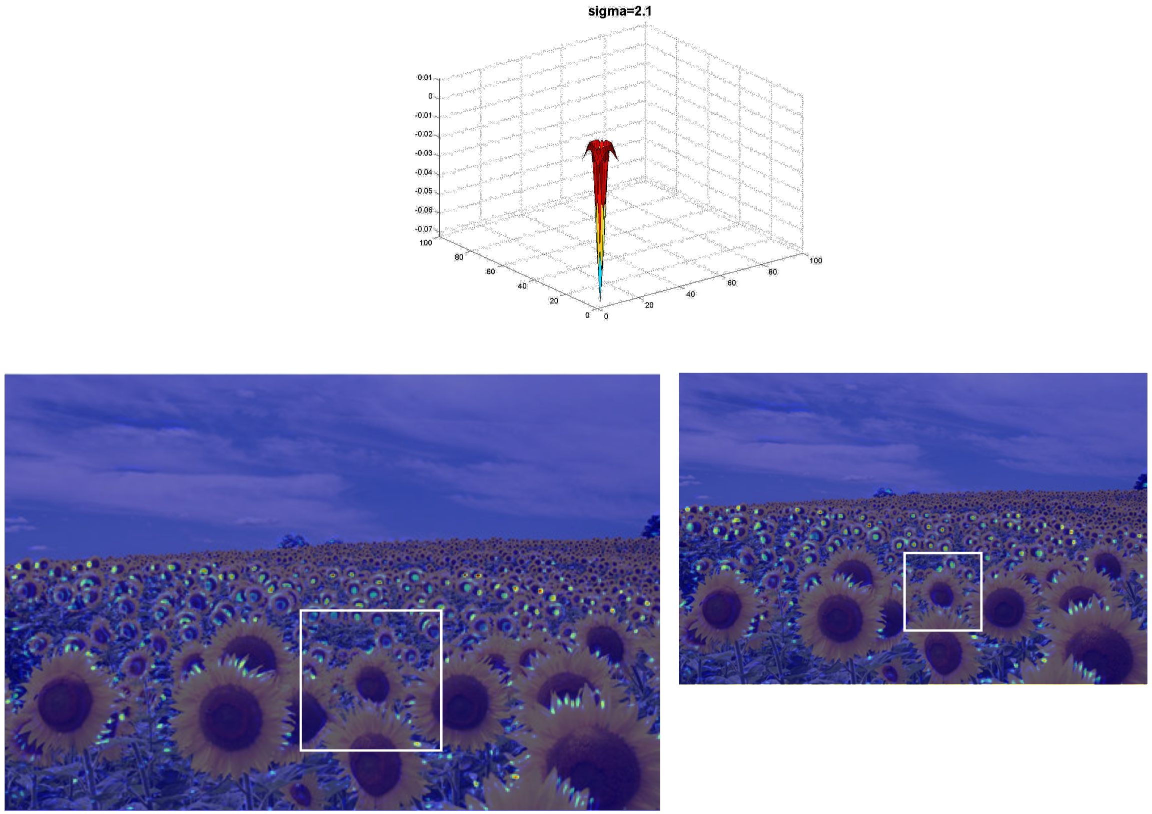
Applying **Laplacian** Filter at Different **Scales**

Full size

3/4 size

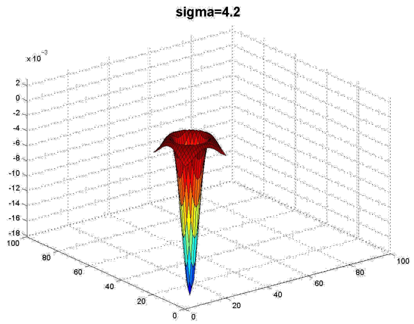
Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

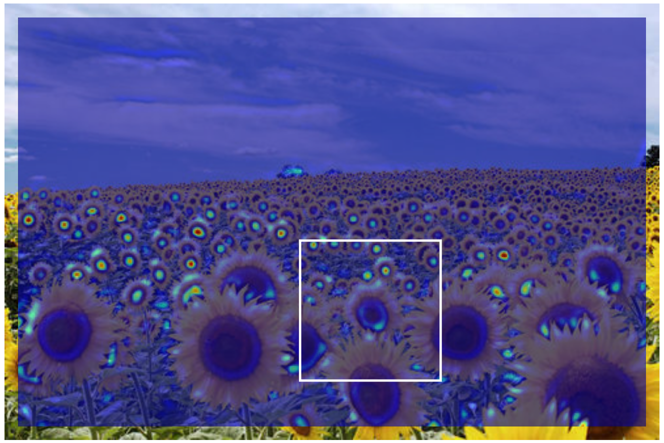
Applying **Laplacian** Filter at Different **Scales**

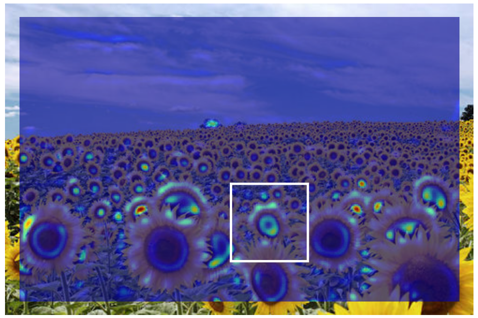


Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

Applying **Laplacian** Filter at Different **Scales**

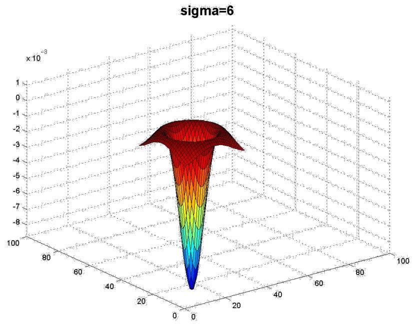


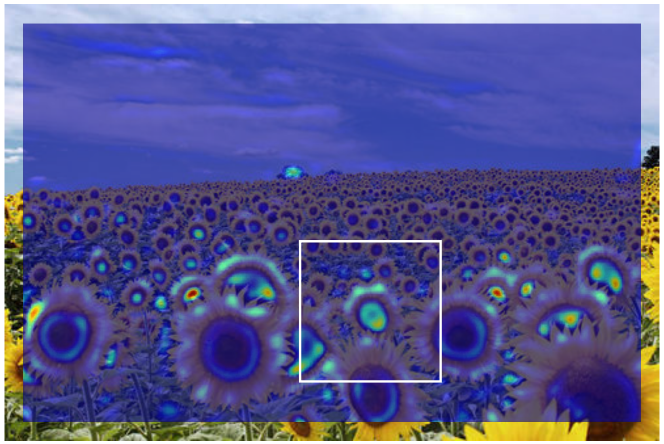


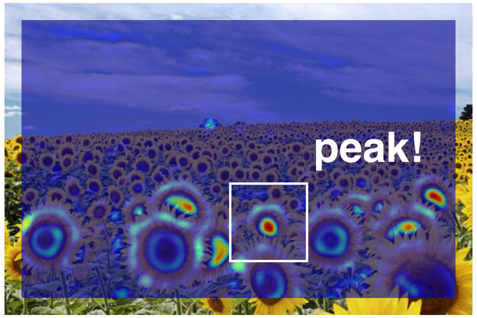


Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

Applying **Laplacian** Filter at Different **Scales**

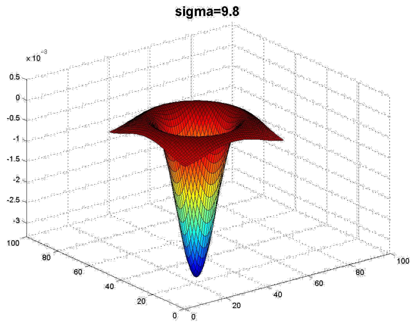


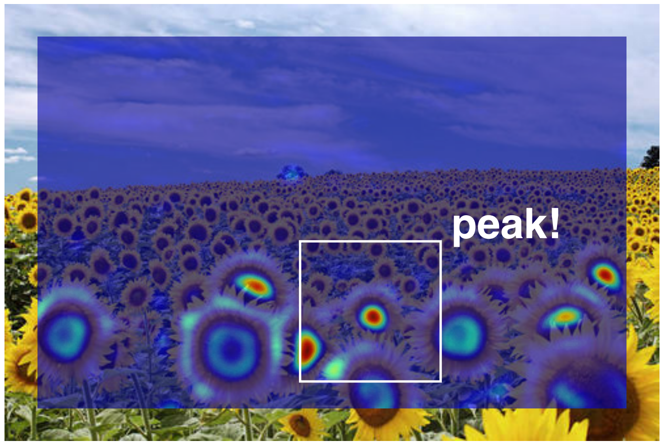


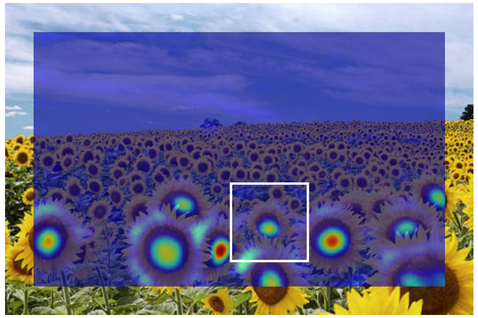


Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

Applying **Laplacian** Filter at Different **Scales**

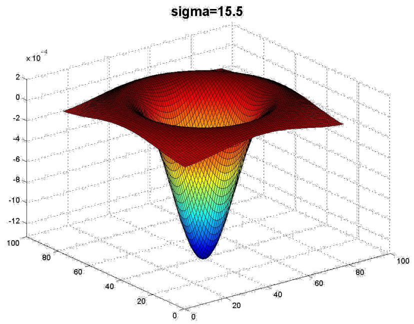


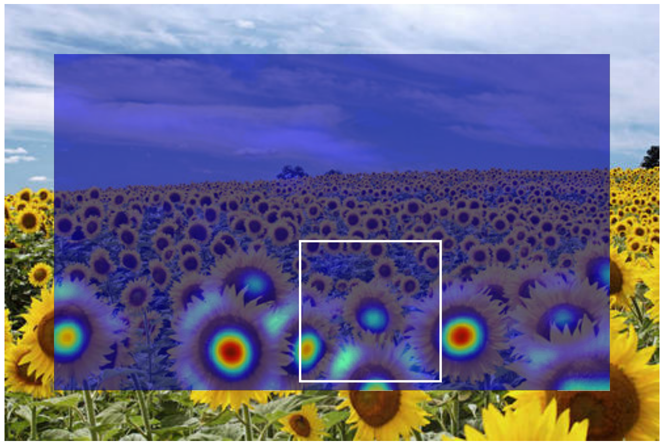


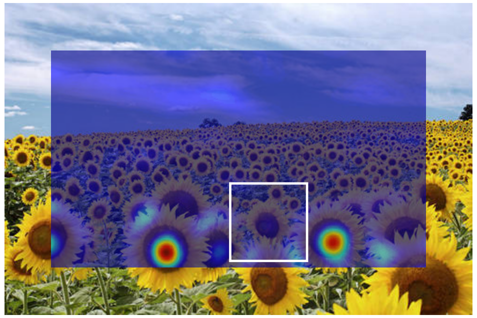


Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

Applying **Laplacian** Filter at Different **Scales**

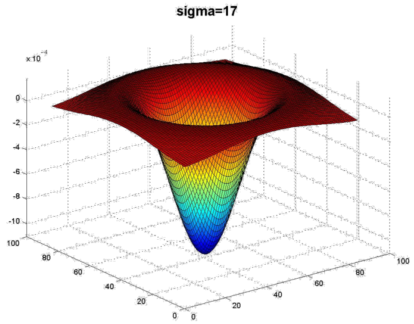


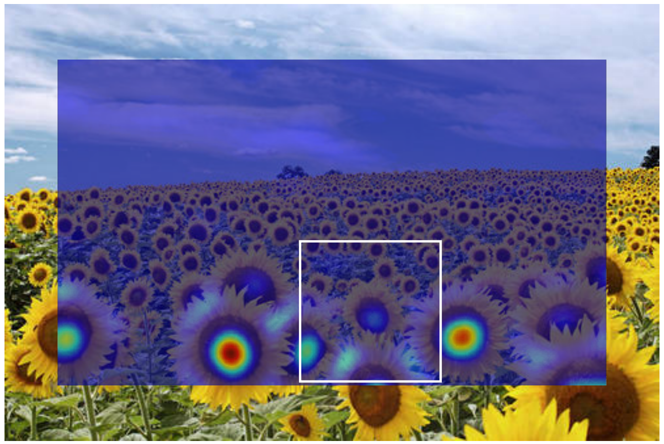


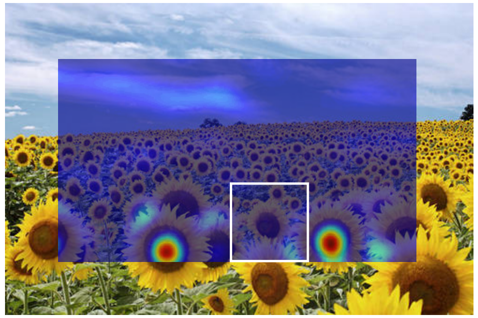


Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

Applying **Laplacian** Filter at Different **Scales**







Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

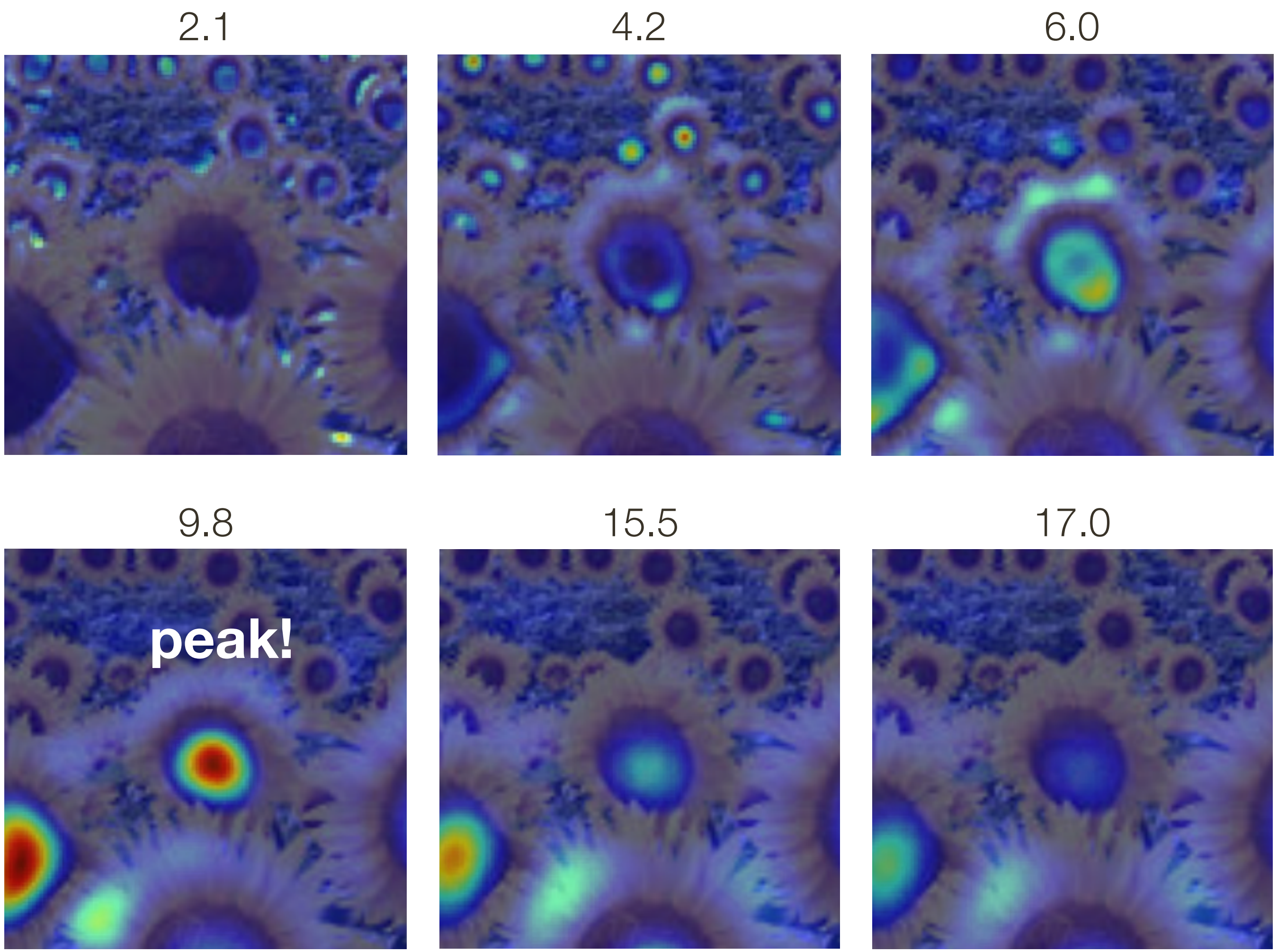
Applying **Laplacian** Filter at Different **Scales**

Full size

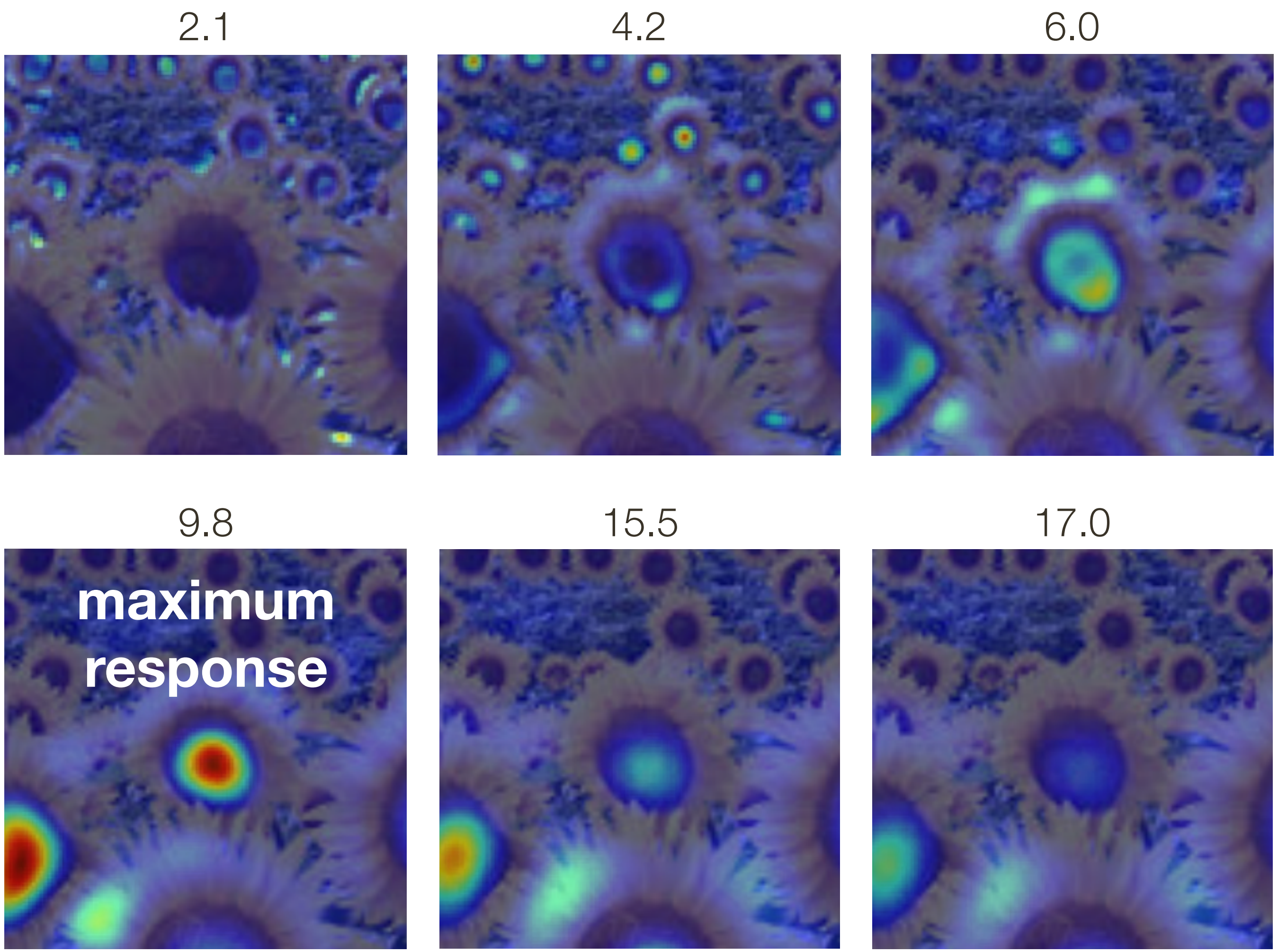
3/4 size

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

Applying **Laplacian** Filter at Different **Scales**

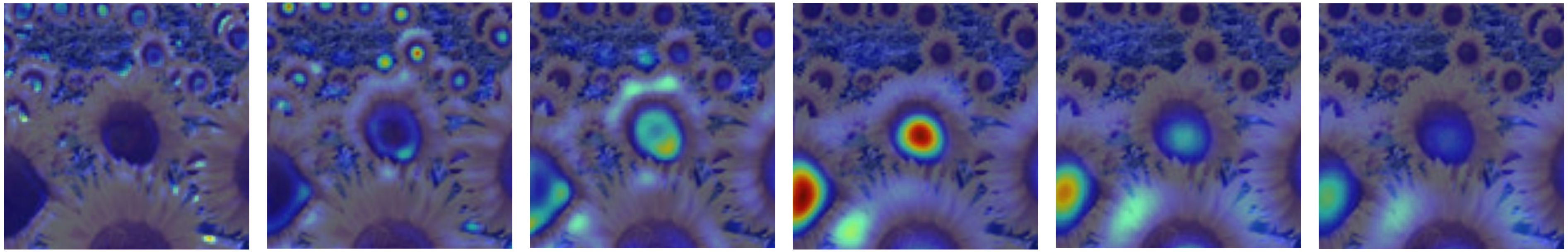


Applying **Laplacian** Filter at Different **Scales**



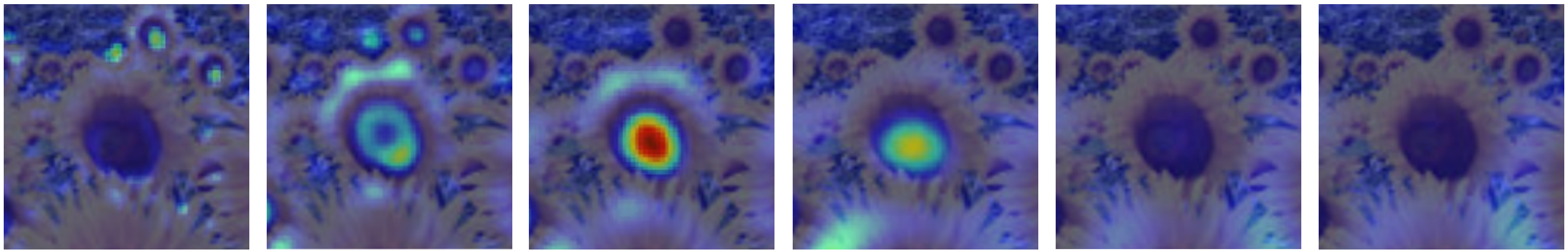
Optimal **Scale**

2.1 4.2 6.0 9.8 15.5 17.0



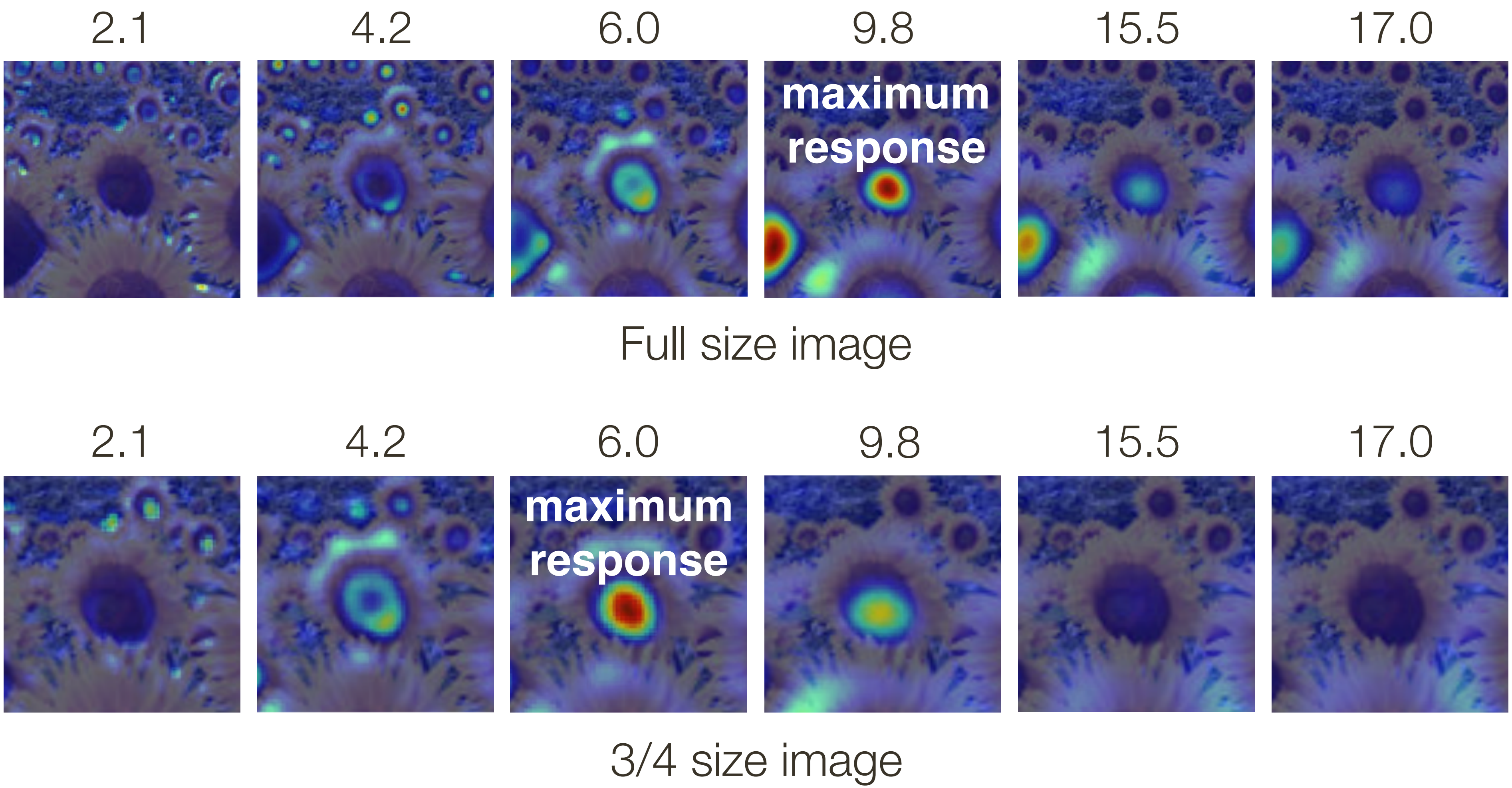
Full size image

2.1 4.2 6.0 9.8 15.5 17.0

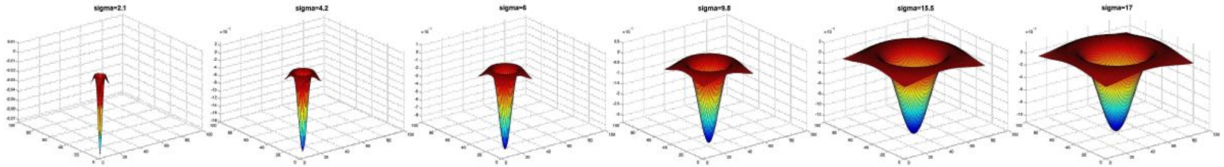


3/4 size image

Optimal **Scale**



Applying **Laplacian** Filter at Different **Scales**



Full size

3/4 size