

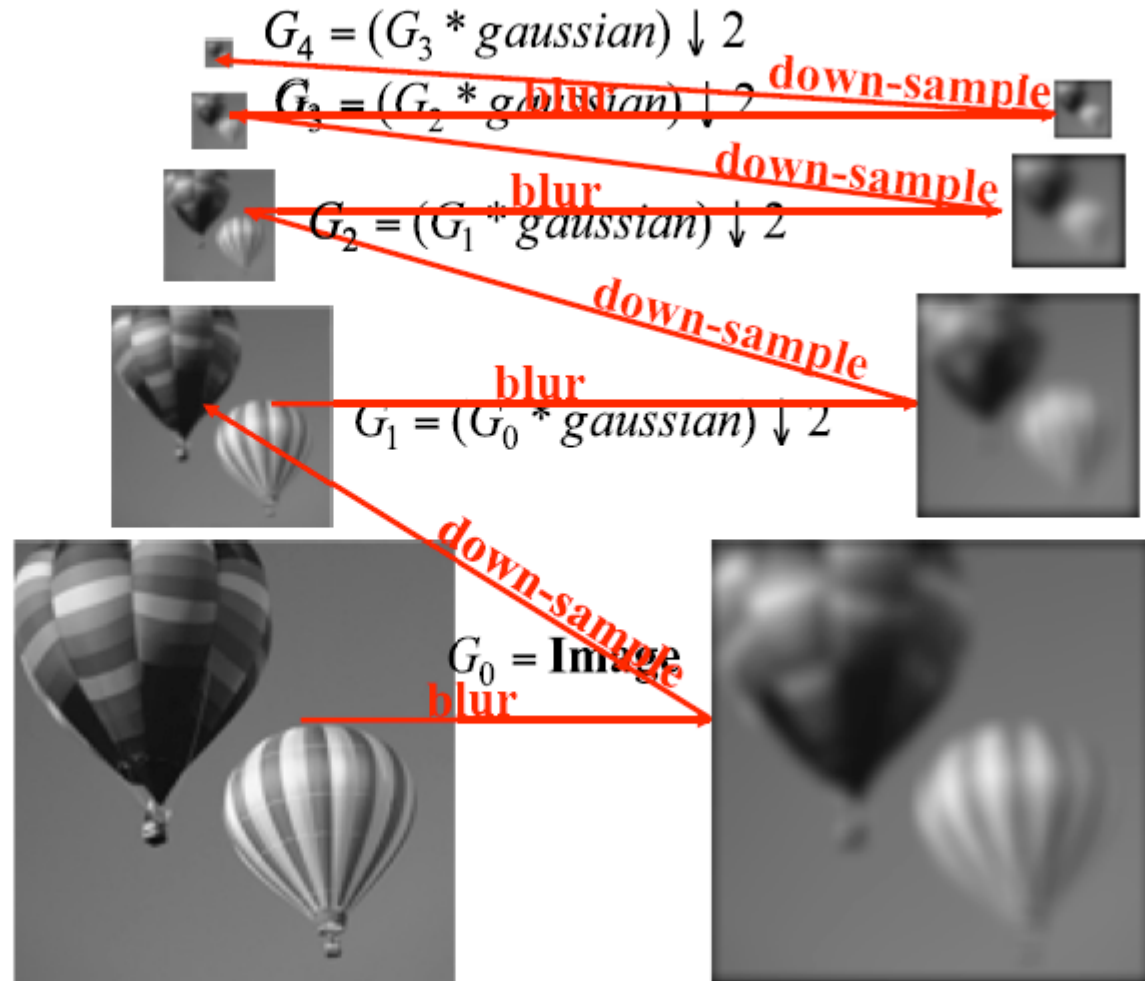
Corner Detection

Last Week

The Gaussian Pyramid

Low resolution

High resolution



Source: Irani

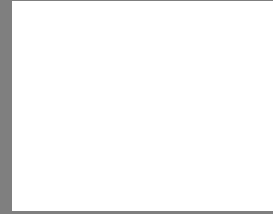
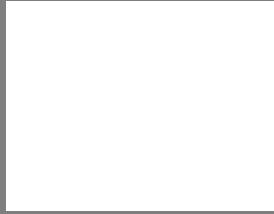
Outline

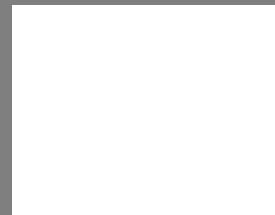
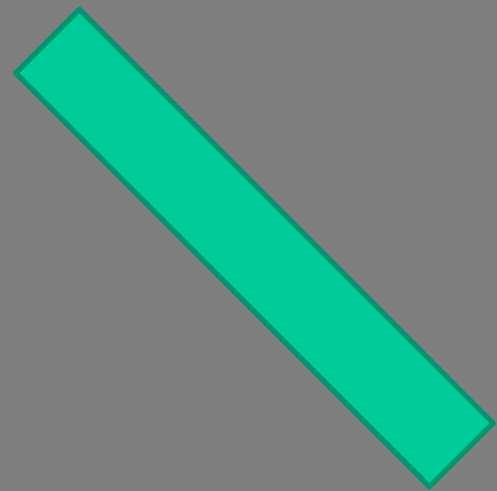
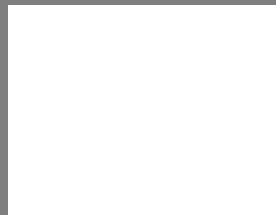
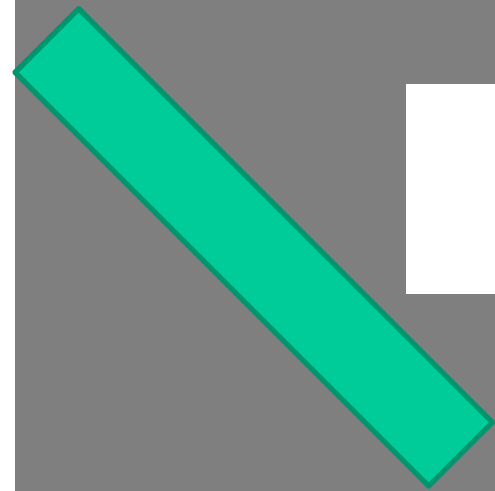
- Corners and point features
- Moravec operator
- Image structure tensor
- Harris corner detector
- Sub-pixel accuracy
- SUSAN
- FAST
- Example descriptor: SIFT

Point Features

(a.k.a. Corners or Interest Points)

- Many applications need to match corresponding points in images.
 - Stereo matching and 3D reconstruction
 - Tracking
 - Localization
 - Recognition
- Points along lines are poorly defined
 - See Aperture Problem phenomenon (next)



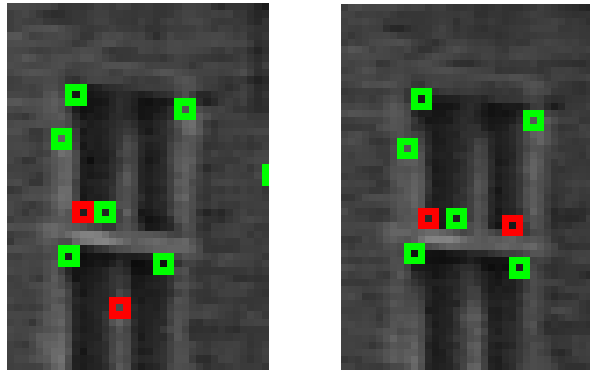


Tracking & Reconstruction

- 2D3 Commercial application: Boujou
 - Match-moving for special effects
 - Computes a point-cloud + camera poses
- UNC city-scanning (video) (Pollefeys et al.)

Feature matching vs. tracking

Image-to-image correspondences are key to passive triangulation-based 3D reconstruction



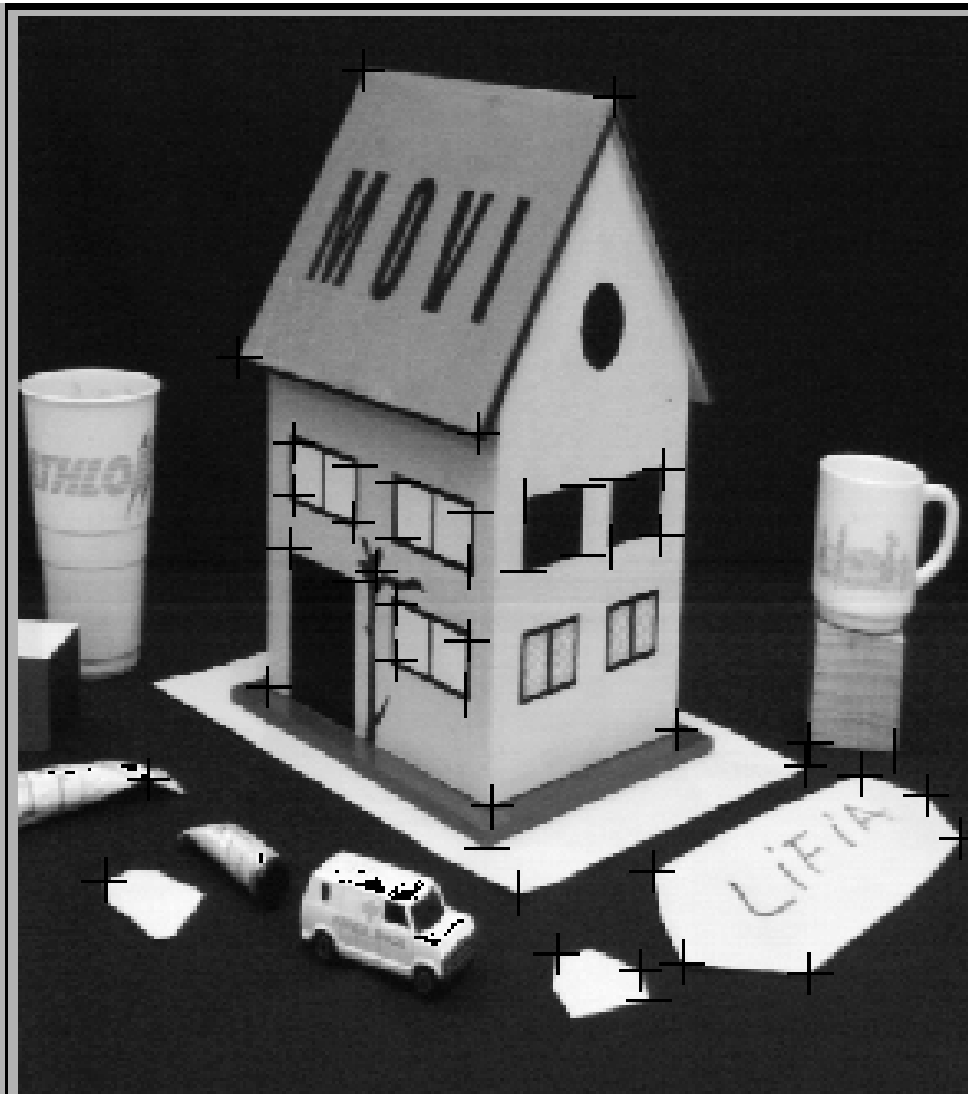
Extract features independently and then match by comparing descriptors



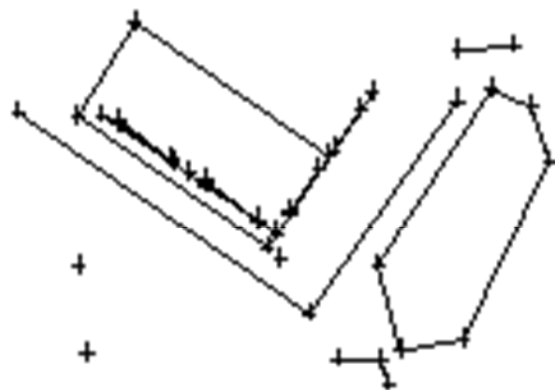
Extract features in first images and then try to find same feature back in next view

What is a good feature?

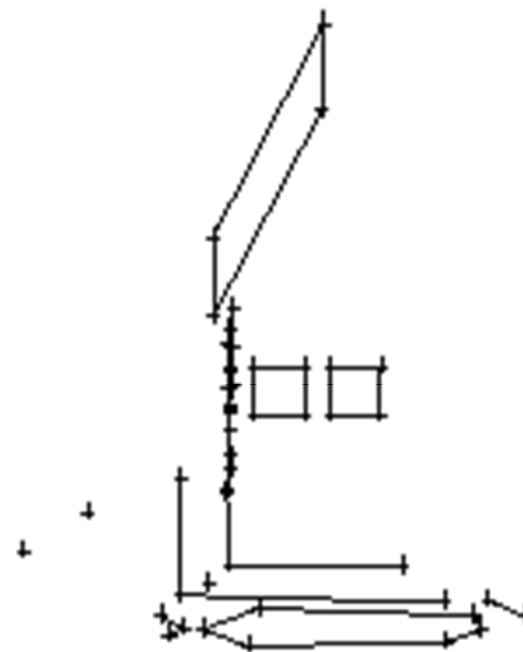
Stereo Example



Stereo Example



top view



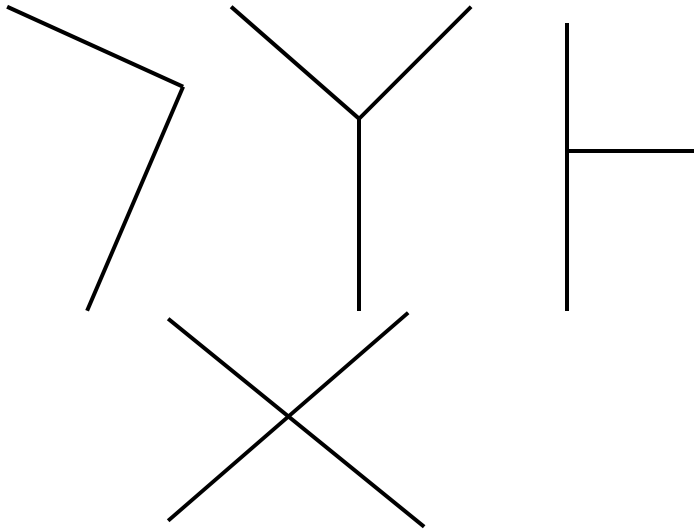
lateral view

“Corner” Interest Points

- We want corner points that are:
 - Distinctive
 - Stable from image to image
 - (Sparse) ?
- And are invariant to:
 - View point (scale, orientation, translation)
 - Lighting conditions
 - Object deformations
 - Partial occlusion
- And are, if possible,
 - Geometrically meaningful, though not necessarily scene-object corners.

Intersections

- Edge and line intersections provide recognizable features

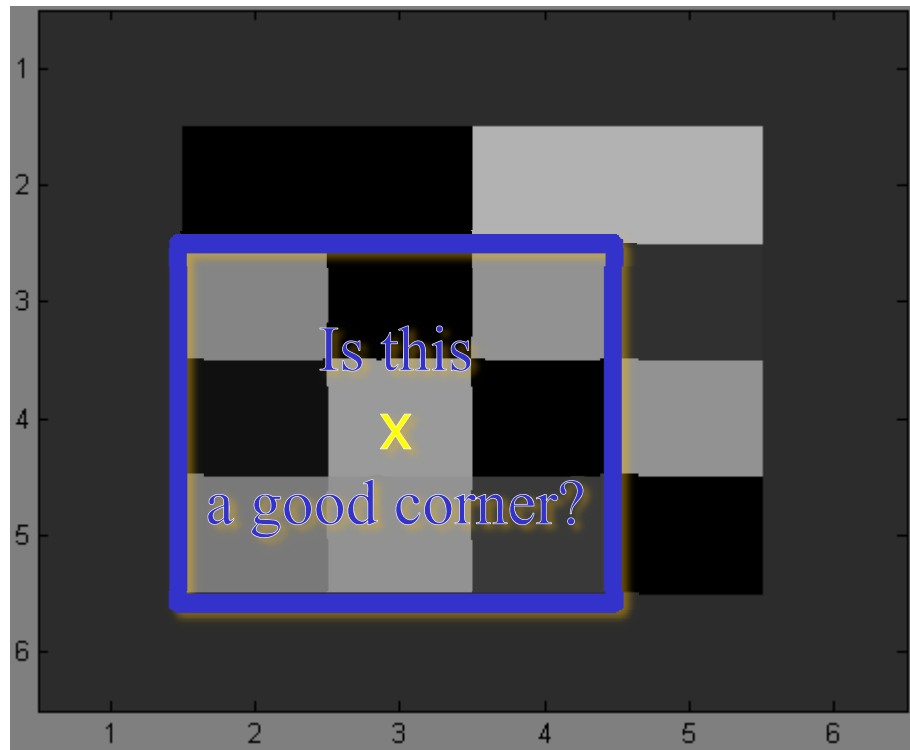


Intersections can be stable or **unstable**



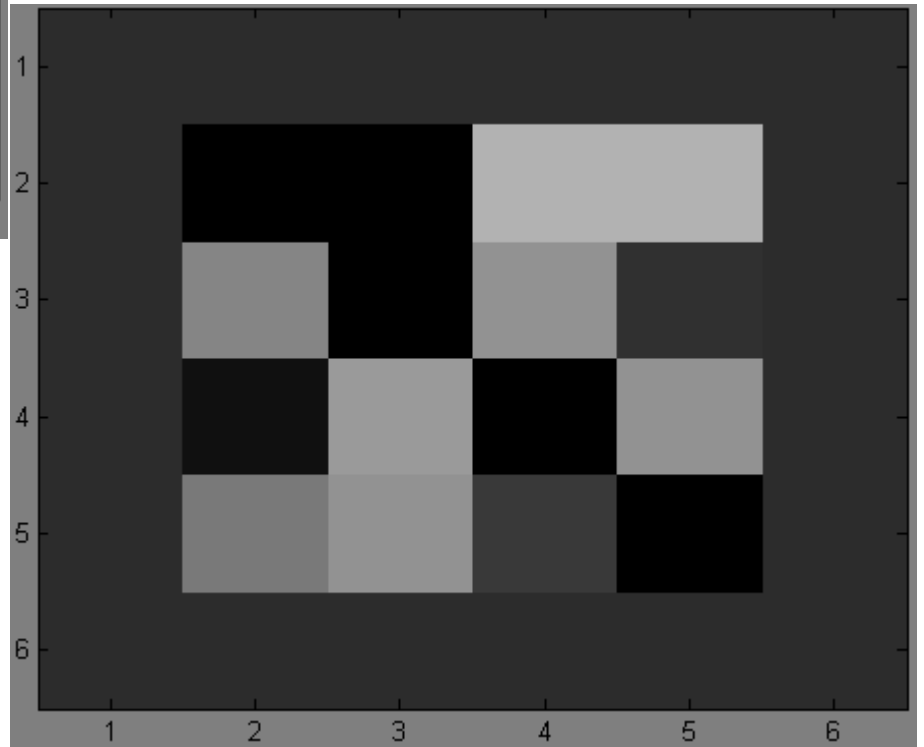
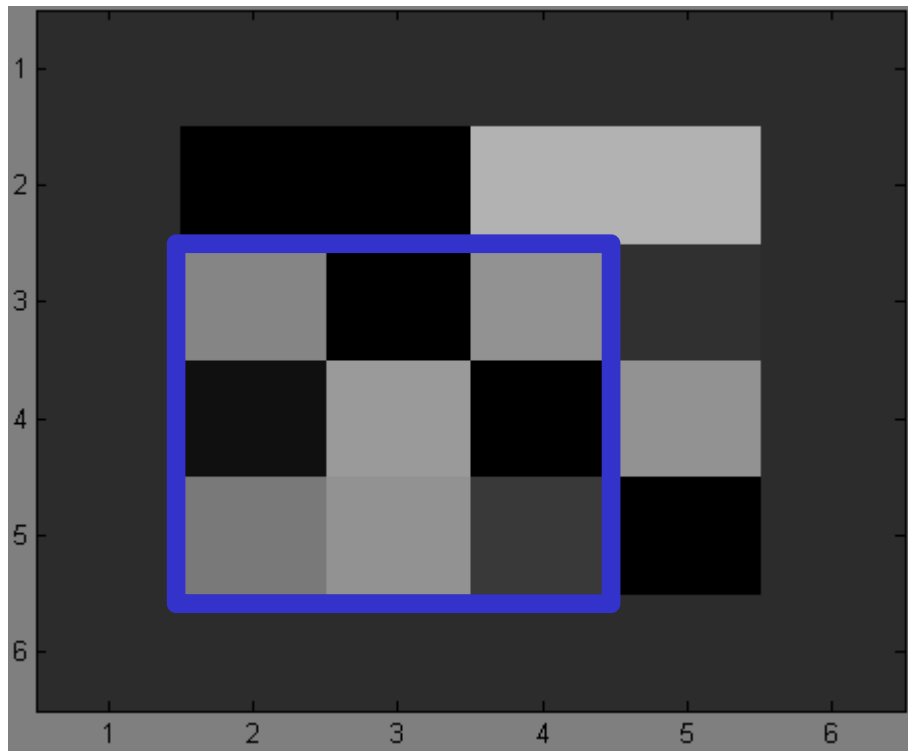
Moravec “Interest” Operator

- Use a window surrounding each pixel as its own matching template
- Tests local *autocorrelation* of the image:
 - SSD = Sum of Squared Differences
- Good matches in any direction
 - Flat image region
- Good matches in only one direction
 - Linear feature or edge
- No good matches in any direction
 - Distinctive point feature
 - Corner point



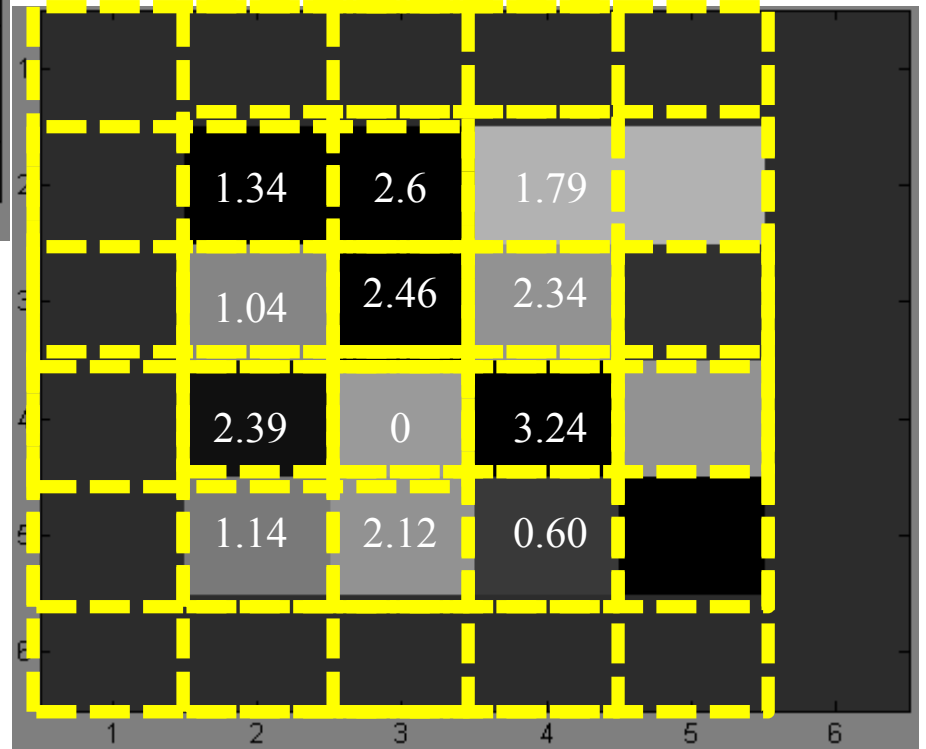
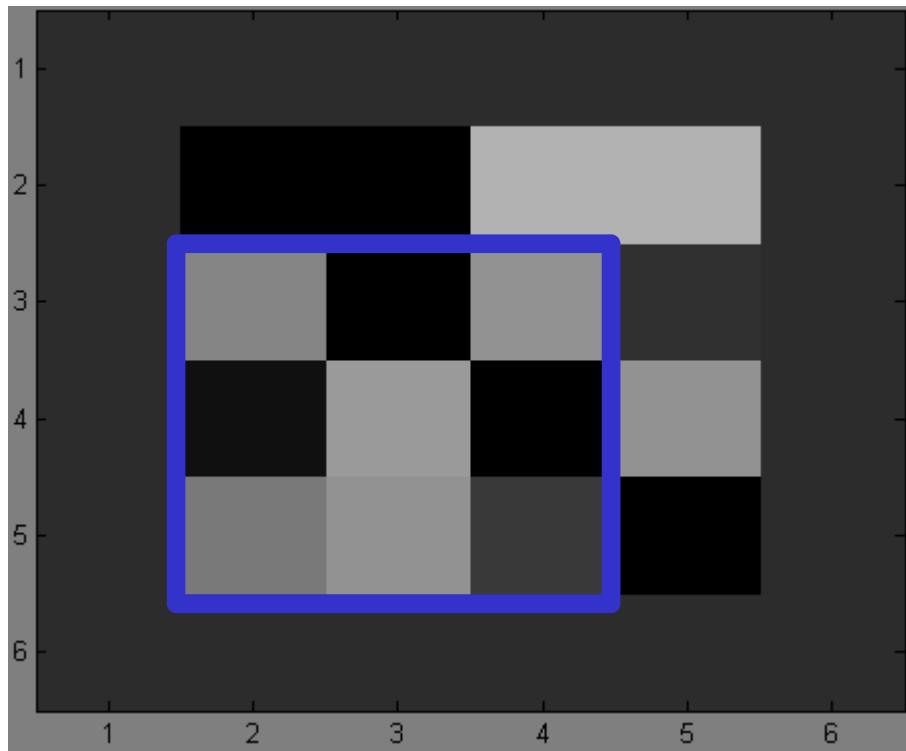
Neighborhood
stands out?

Check SSDs



```
sum(sum( (patch - Ipad(rowStart:rowStart+2, colStart:colStart+2) ).^2 ));
```

Check SSDs



Implementation

- Match quality for displacements $(\Delta x, \Delta y)$ uses L_2 norm:

$$\varepsilon(\Delta x, \Delta y) = \sum_{(x, y) \in N(3,3) \text{ of } I, \text{ centered at } p_0} w(x, y) (I(x + \Delta x, y + \Delta y) - I(x, y))^2$$

- w is a window function
 - e.g. constant or Gaussian
- Cornerness is $\min \varepsilon$ over the eight-neighbourhood N , excluding $(0, 0)$

Moravec “Interest” Operator

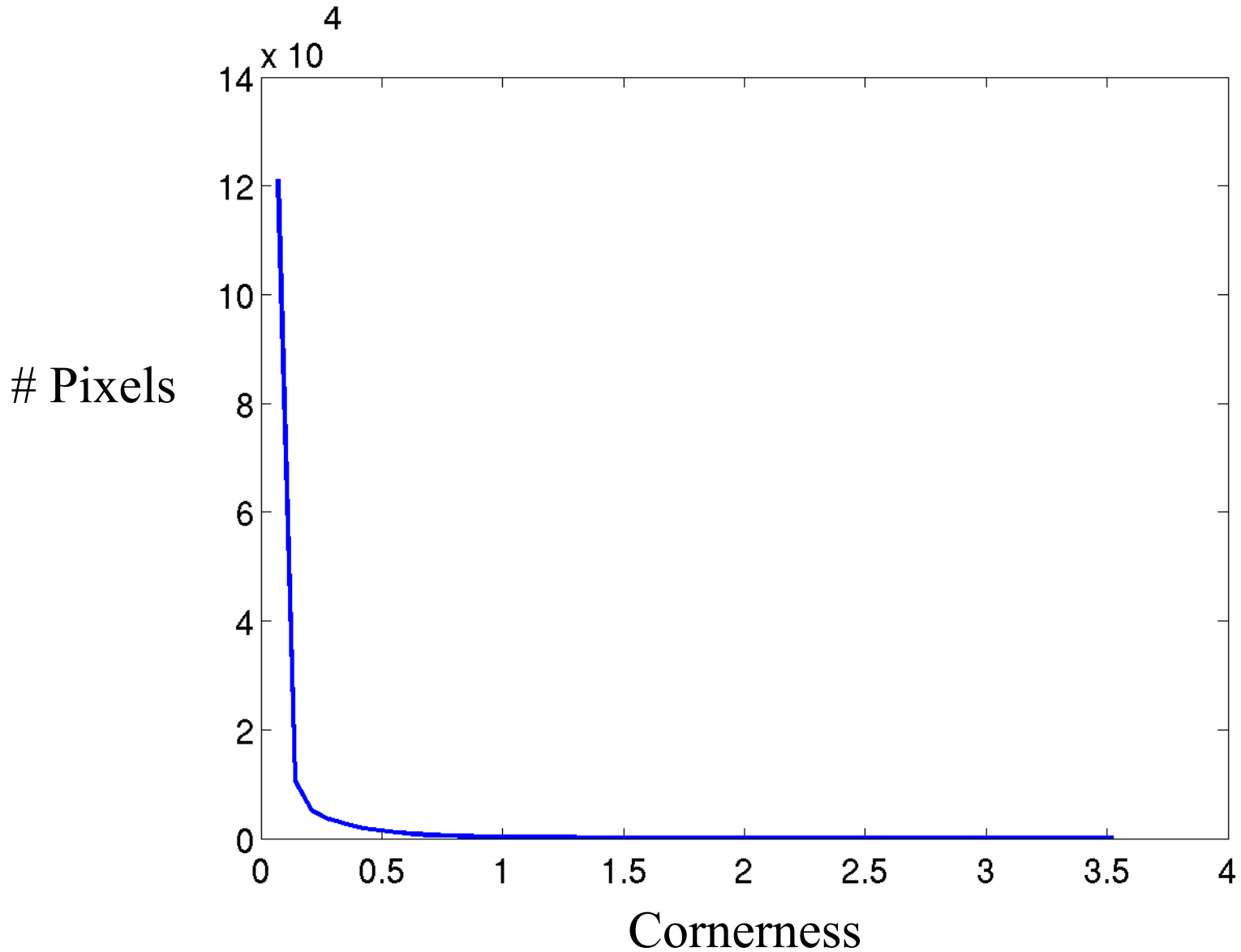
- Use a window surrounding each pixel as its own matching template T .
- Tests local *autocorrelation* of the image: SSD
- Good matches in any direction
 - Flat image region
- Good matches in only one direction
 - Linear feature or edge
- No good matches in any direction
 - Distinctive point feature
 - Corner point

Cornerness

- Using 7x7 matching window.

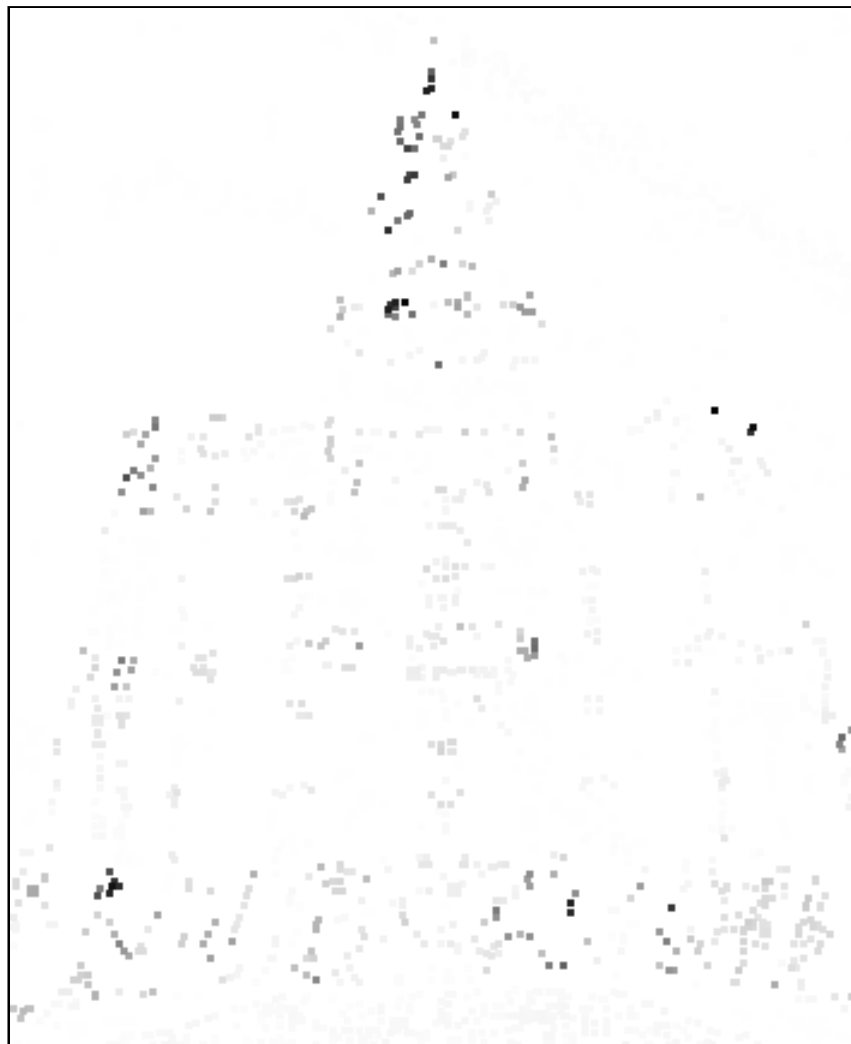


Cornerness distribution

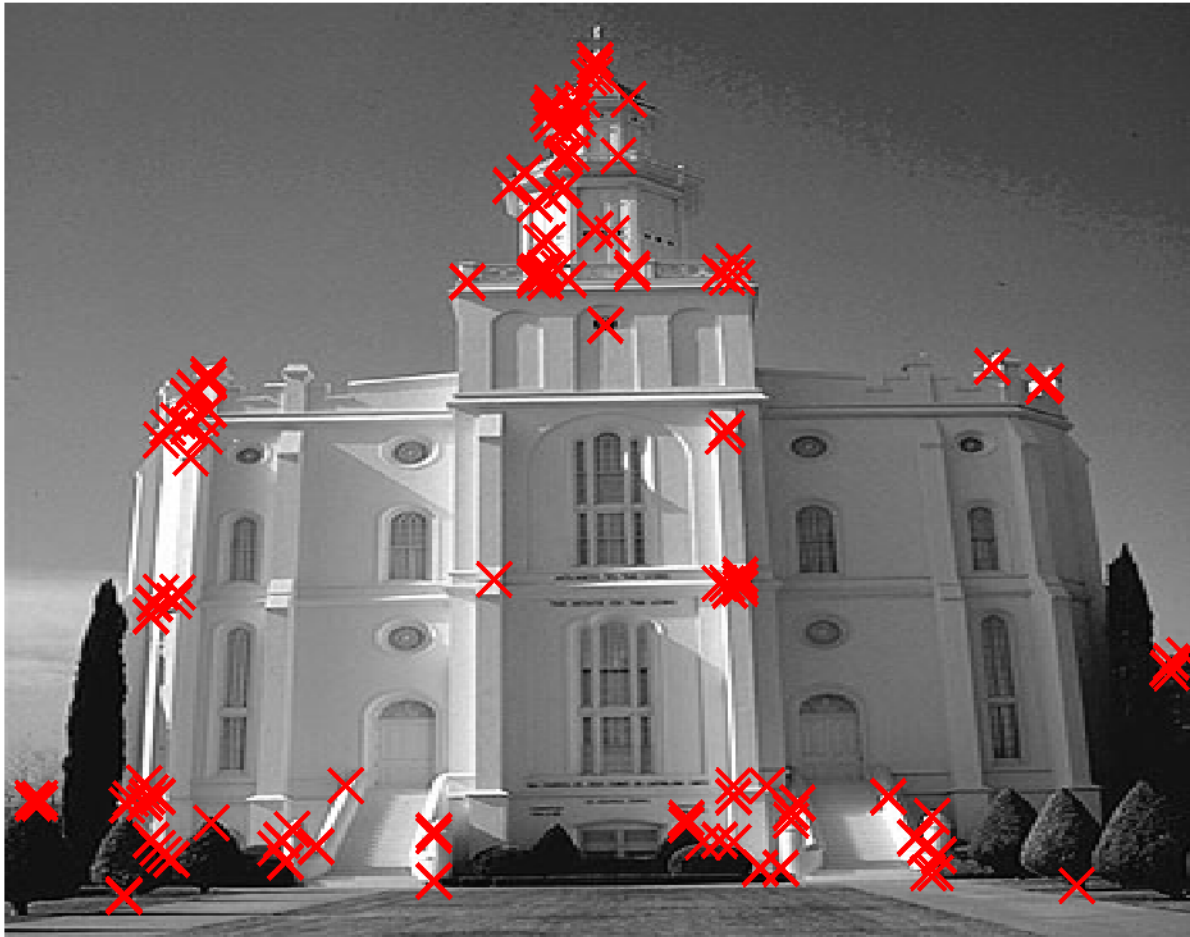


Non-maximal suppression

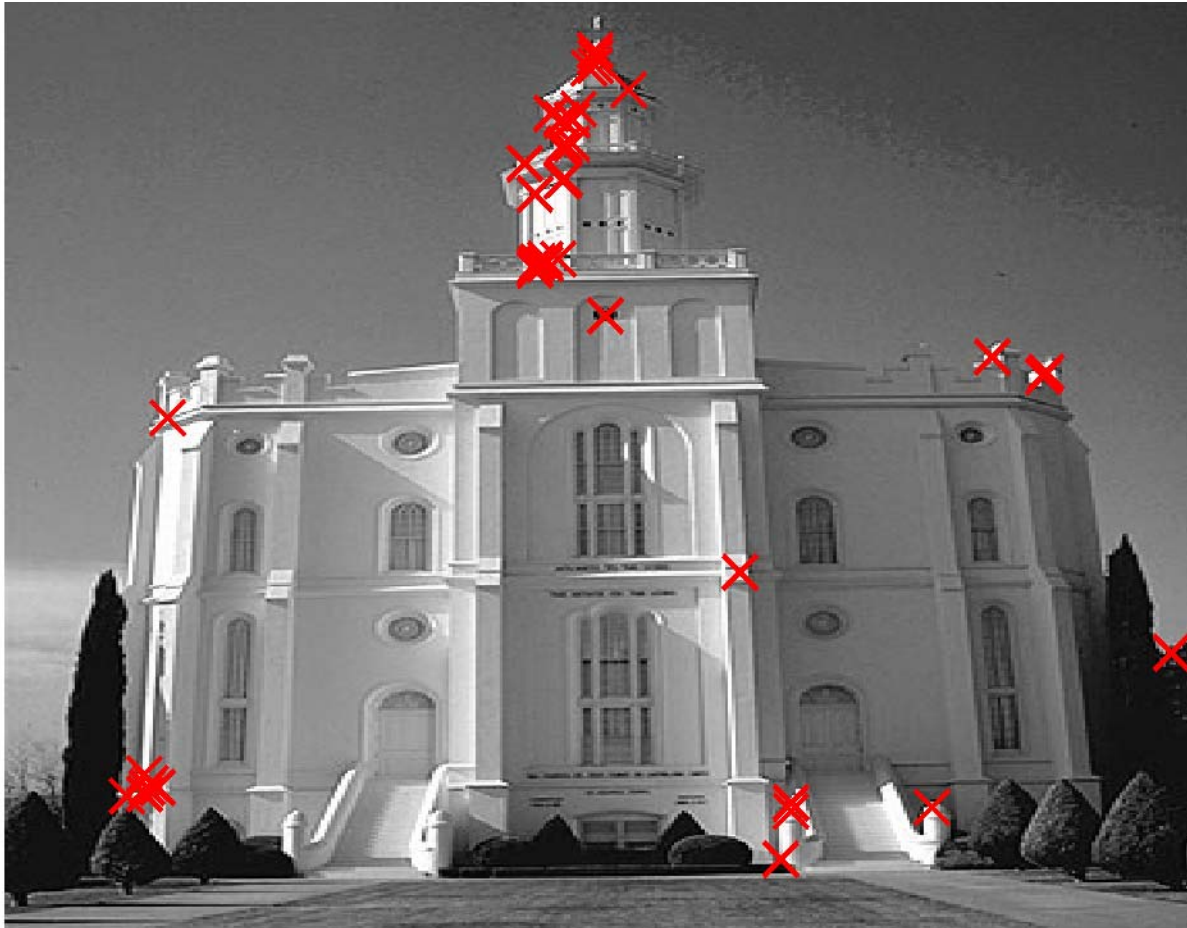
- Set pixels that have an 8 neighbour with higher cornerness to zero.



Threshold $T = 1$



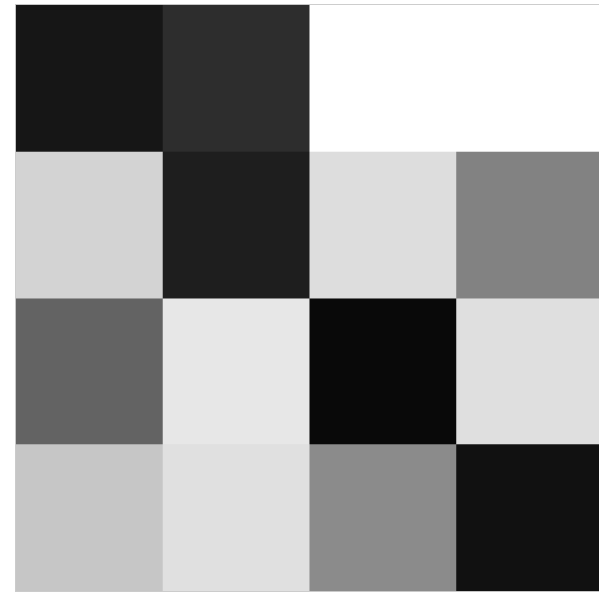
Threshold $T = 2$



Threshold $T = 3$

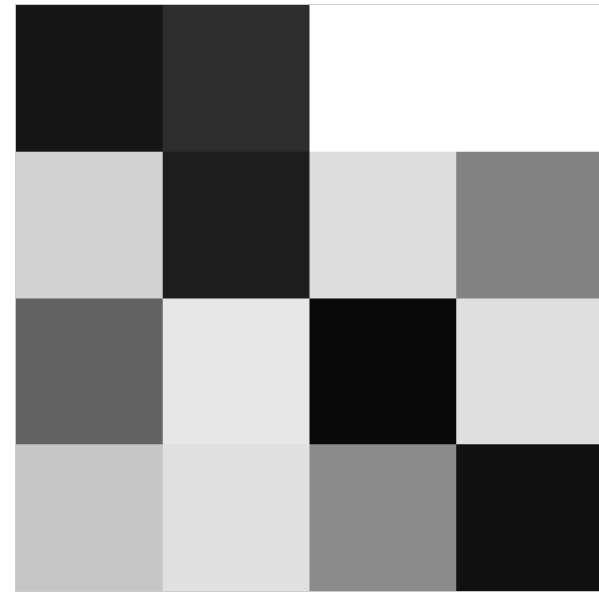


Non-max Suppression Efficiently?



Input, I
(this time, pretend these
are cornerness values)

Non-max Suppression by Dilation?



Input, I
(this time, pretend these
are cornerness values)

```
imdilate(I, ones(3,3))
```



Moravec Algorithm Summary

- Enhance corner features
- Non-maximal suppression
- Threshold

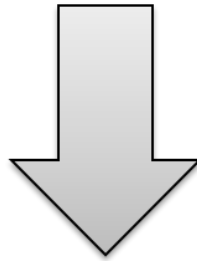
Problems

- Multiple responses at high interest points
 - Extend non-maximal suppression to windows
- Weak response to “blurred” corners
- Slow

General Form

Can we do better? Derive general theory of
“cornerness”

$$\varepsilon(\Delta x, \Delta y) = \sum_{(x, y) \in N(3,3) \text{ of } I, \text{ centered at } p_0} w(x, y) (I(x + \Delta x, y + \Delta y) - I(x, y))^2$$



$$\varepsilon(x, y) = (x, y) \begin{pmatrix} \langle I_x^2 \rangle & \langle I_x I_y \rangle \\ \langle I_x I_y \rangle & \langle I_y^2 \rangle \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \mathbf{x}^T \mathbf{S} \mathbf{x}$$

(See online Taylor series expansion/approximation)

Analytic Expansion

- Taylor expansion of $I(x + u, y + v)$ and substitution into ε gives:

$$\varepsilon(x, y) = (x, y) \begin{pmatrix} \langle I_x^2 \rangle & \langle I_x I_y \rangle \\ \langle I_x I_y \rangle & \langle I_y^2 \rangle \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \mathbf{x}^T \mathbf{S} \mathbf{x}$$

- $\langle \dots \rangle$ denotes average over the window N .
- S is the image structure tensor

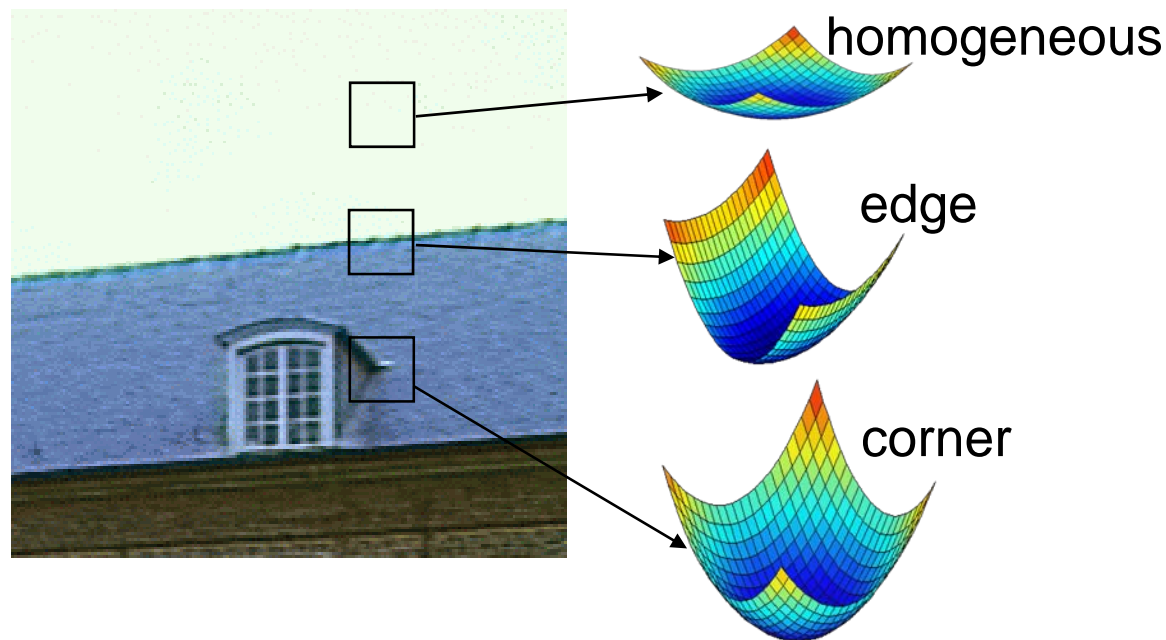
Structure tensor

- S captures the curvature of the local autocorrelation surface
- The eigenvalues are the principal curvatures
- They are the solutions to

$$\lambda^2 - \lambda \left(\langle I_x^2 \rangle + \langle I_y^2 \rangle \right) + \langle I_x^2 \rangle \langle I_y^2 \rangle - \langle I_x I_y \rangle^2 = 0$$

Principal Axes (i.e. Eigenvectors)

(Note: see Simon Prince's SVD slides online)



i.e. maximize smallest eigenvalue of S

Harris Corner Detector

(Example Video)

- Defines cornerness as size of smallest eigenvalue, or

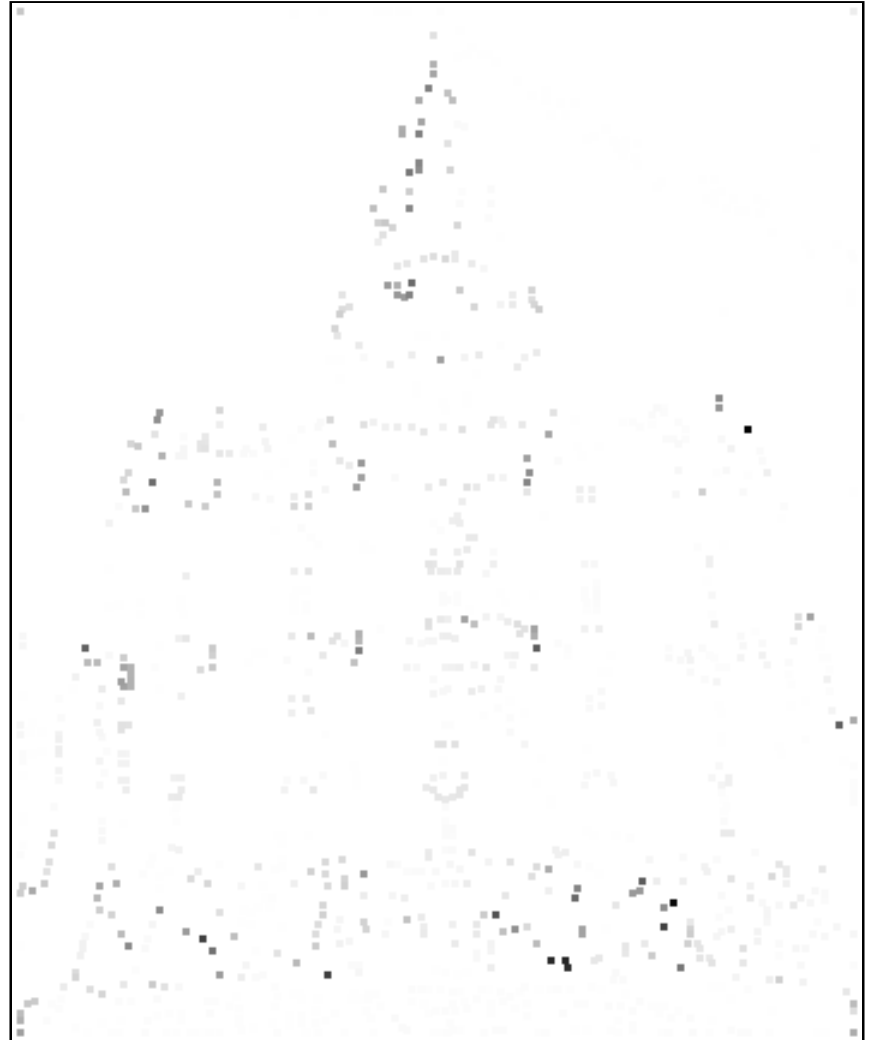
$$C = \det(S) / \text{Tr}(S)$$

$$C = \left(\langle I_x^2 \rangle \langle I_y^2 \rangle - \langle I_x I_y \rangle^2 \right) / \left(\langle I_x^2 \rangle + \langle I_y^2 \rangle \right)$$

$$C = \lambda_1 \lambda_2 / (\lambda_1 + \lambda_2)$$

- Non-maximal suppression and thresholding as before

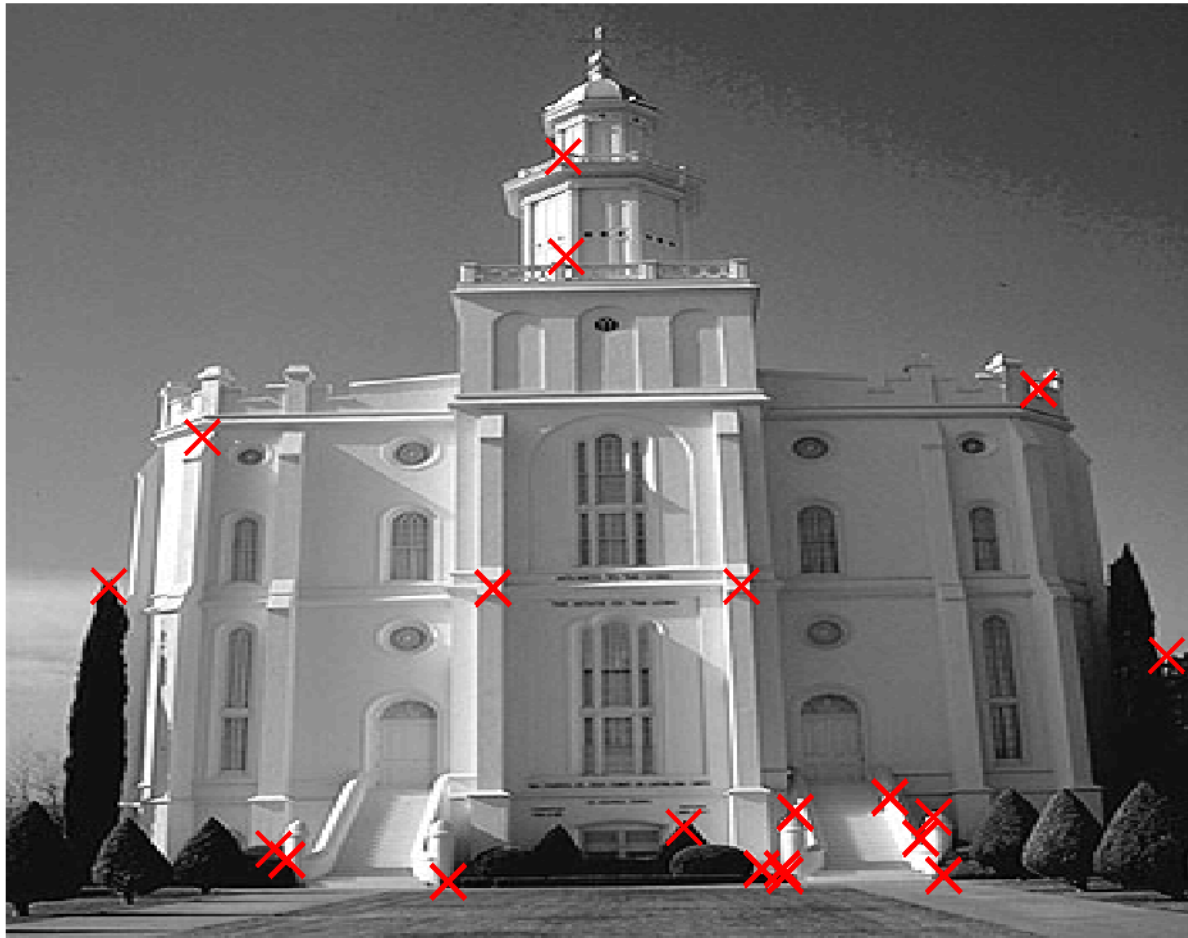
Corneriness and NMS



$\Sigma = 1$, Threshold $T=0.1$



$\Sigma = 1$, Threshold $T=0.15$



$\Sigma = 1$, Threshold $T=0.2$



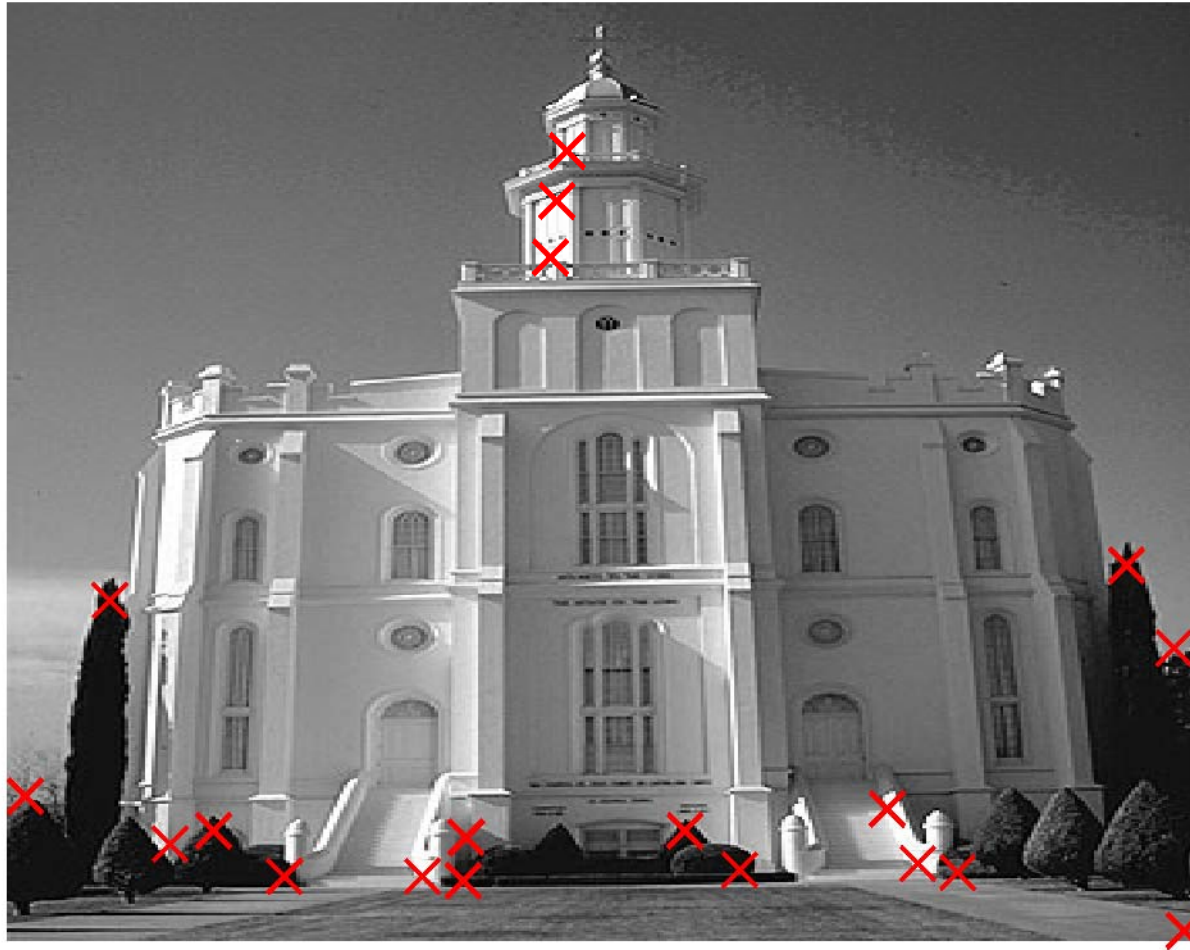
Variants

- Can include Gaussian smoothing when computing I_x and I_y .
 - Less important than for edge detection.
 - Helps eliminate multiple responses to the same corner
 - Similar effect using larger regions in non-maximal suppression
- Harris and Stephens combined edge and corner detector
 - $R = \det(S) - k\text{Tr}^2(S)$
- Various other corner measures, thresholding schemes, non-max suppression techniques
- $0 < k < 0.25$ to get the desired behaviour from R:
 - positive at corners and negative at edges

Sigma = 5, Threshold
 $T=0.001$



$\Sigma = 5$, Threshold
 $T=0.004$

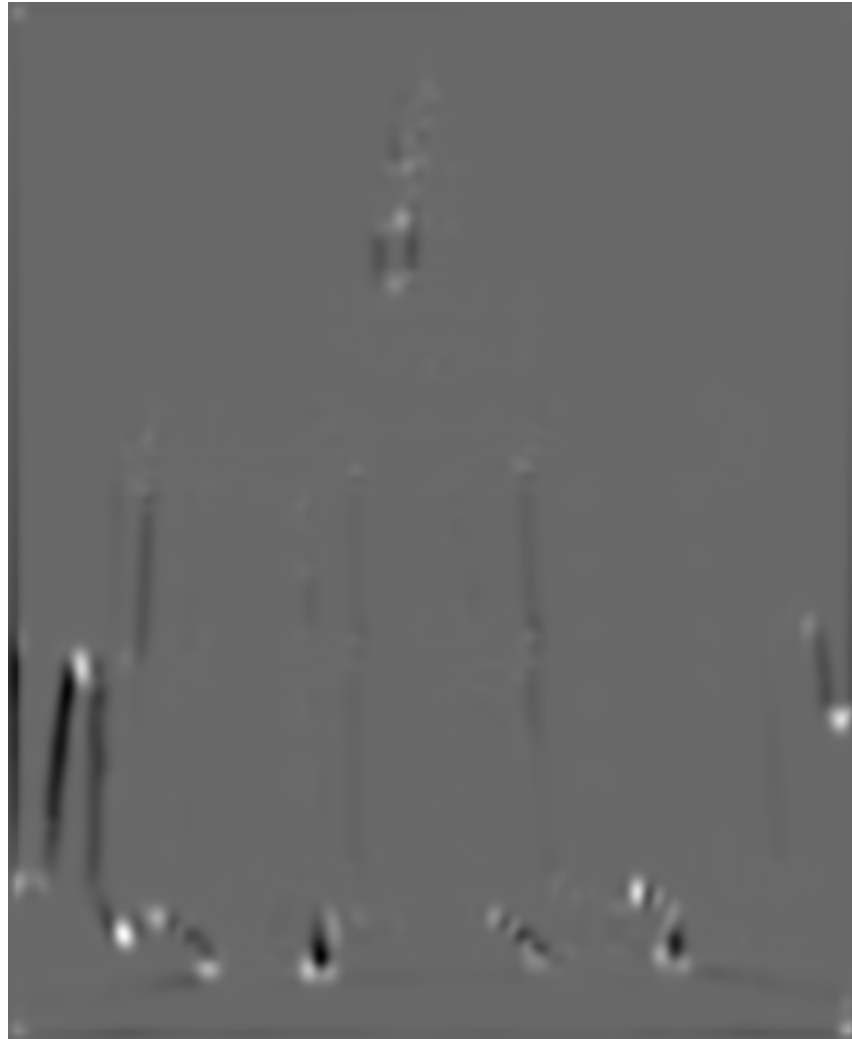


$\Sigma = 5$, Threshold
 $T=0.007$



Harris and Stephens

$k = 0.02$



Harris and Stephens,
Proc. 4th Alvey Vision
Conference, 147-151,
1988.

SUSAN

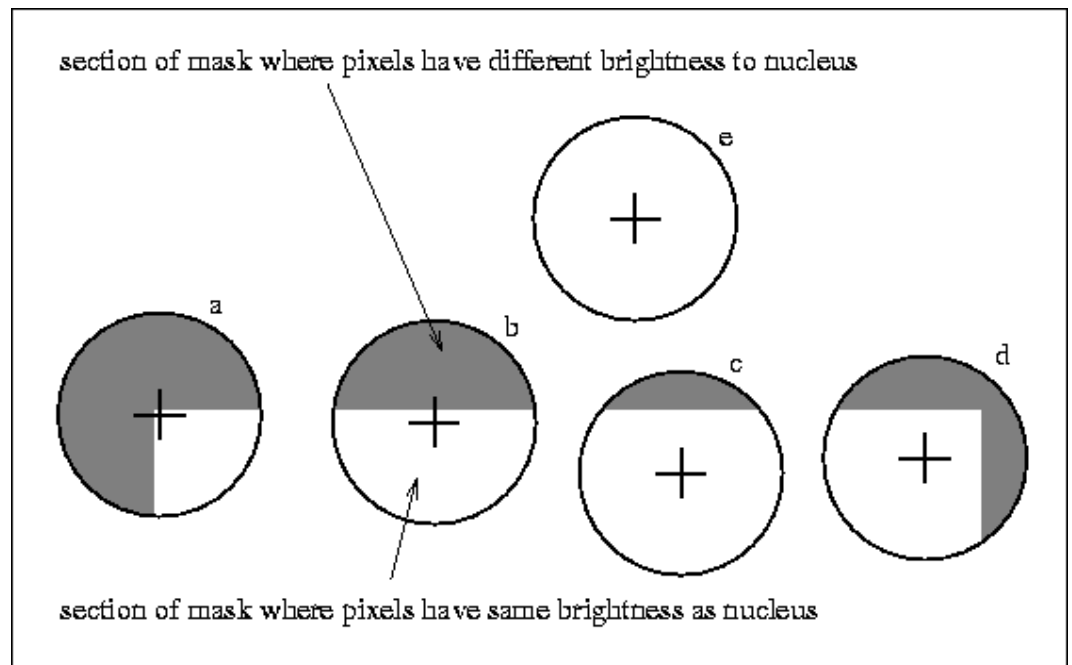
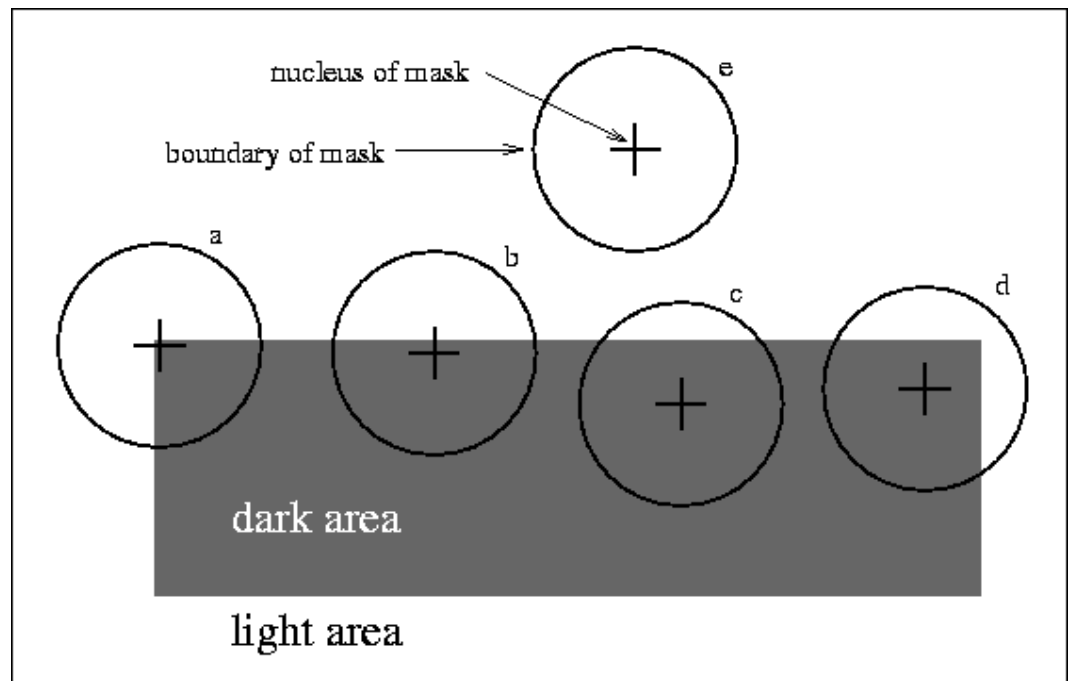
- Method for edge and corner detection
- No image derivatives
- Insensitive to noise

- Smith et al, IJCV, 23(1):45-78, 1997

USAN

“Univalue Segment
Assimilating
Nucleus”

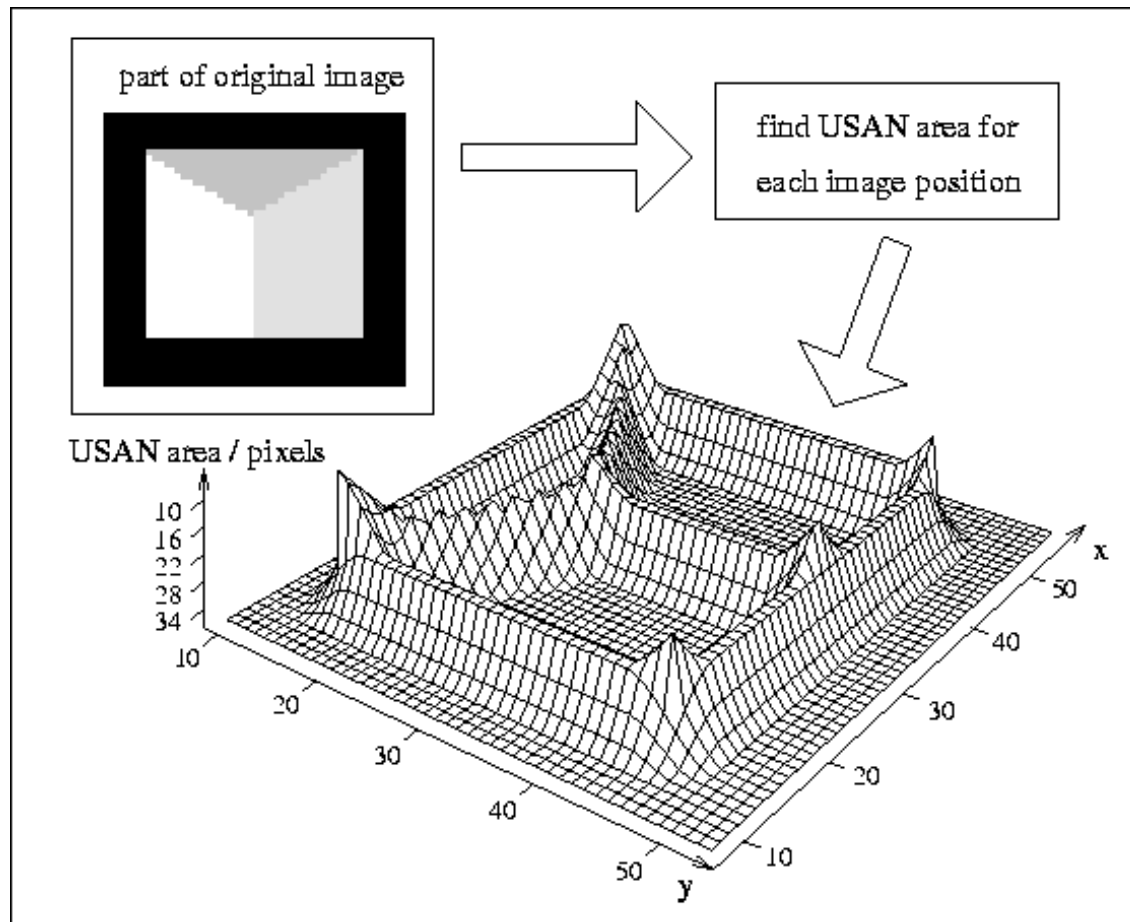
It is the portion of the
template with intensity
within a threshold of the
“nucleus”.



Edges and Corners

- In flat regions the USAN has similar area to the template
- At edges the USAN area is about half the template area
- At corners the USAN area is smaller than half the template area.
- “SUSAN” = Smallest USAN.

Example



Implementation

```

00111100
01111110
11111111
C=11111111
11111111
01111110
00111100

```

- Circular mask C with radius 3.4 ($|C| = 37$ pixels).
- The nucleus is the centre pixel r_0 .

$$u(r, r_0) = \begin{cases} 1 & |I(r) - I(r_0)| < t \\ 0 & \text{otherwise} \end{cases}$$

$$n = \sum_{r \in C(r_0)} u(r, r_0)$$

$$A(r_0) = \begin{cases} |C| - n & n < T \\ 0 & \text{otherwise} \end{cases}$$

$T=3|C|/4$ for edge detection

$T=|C|/2$ for corner detection

Select t by considering image noise level.

Refinements

- ‘Band’ edge orientation from USAN moments:
 $\tan^{-1}(\mu_{20} / \mu_{02})$
- Step edge orientation from centre of gravity.
- Eliminate false positive corners using
 - Distance of USAN centre of gravity from r_0 .
 - Connectivity of USAN.
- Non-maximal suppression

FAST

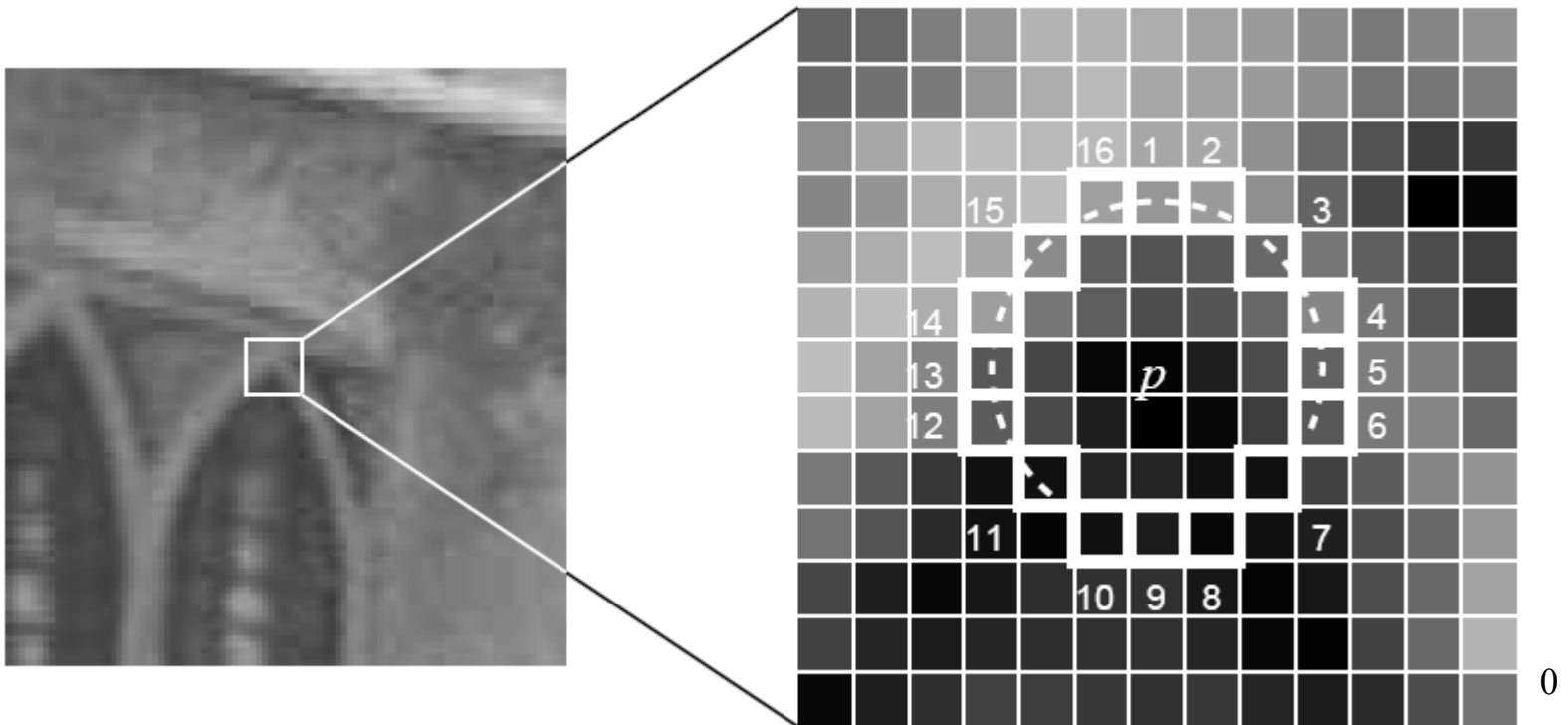
Machine learning for high-speed corner detection
Rosten & Drummond, ECCV 2006

- Reuse nucleus concept
 - From Trajkovic & Hedley'98
- Machine learning to train it to be fast
- Retain performance

Trajkovic & Hedley

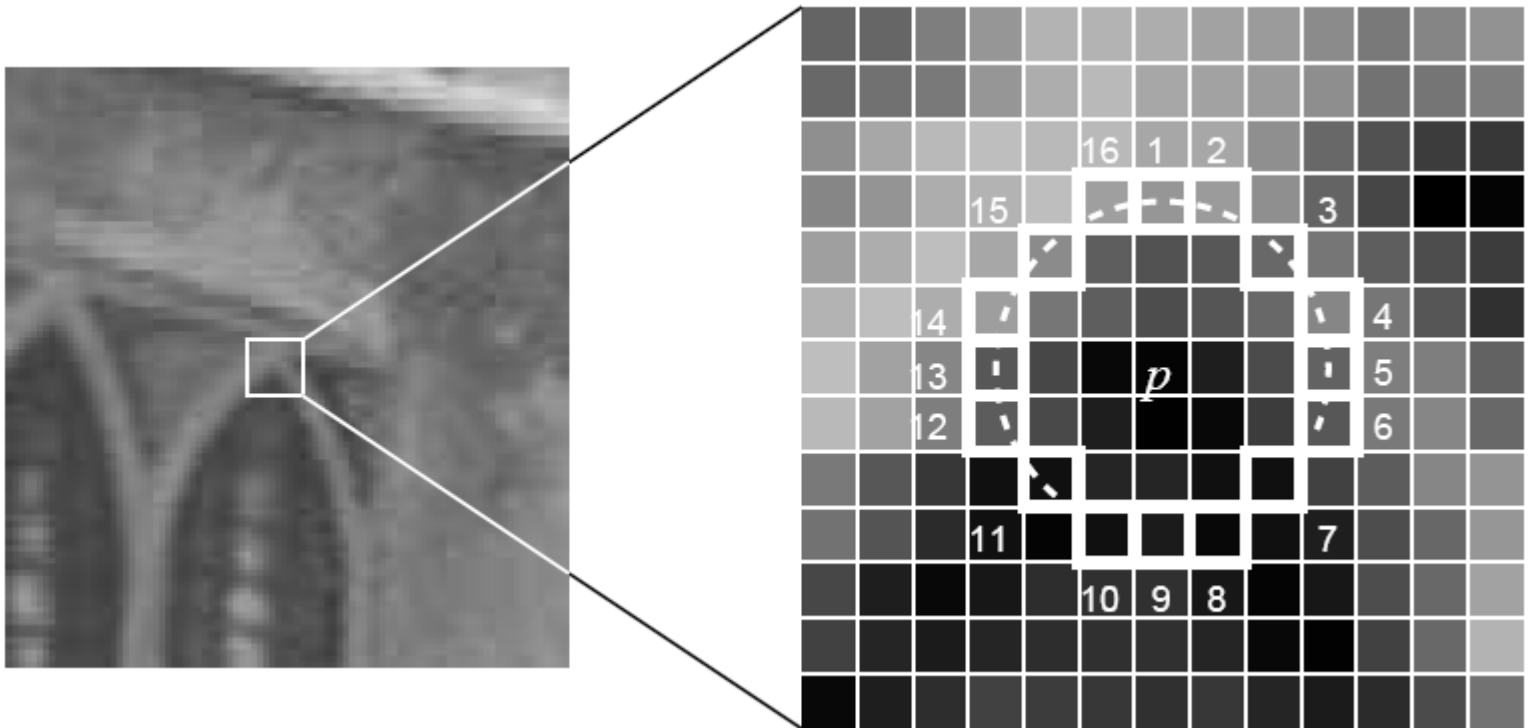
- P and P' are opposite points, diameter D apart

$$C = \min_P (f_P - f_C)^2 + (f_{P'} - f_C)^2$$



FAST

- Set of n contiguous pixels in the circle which are all brighter / darker by some T ?
 - $n = 12$?



- For each location (1-16) on the circle x , the pixel at that position relative to p (denoted by $p \rightarrow x$) can have one of three states:

$$S_{p \rightarrow x} = \begin{cases} d, & I_{p \rightarrow x} \leq I_p - t & \text{(darker)} \\ s, & I_p - t < I_{p \rightarrow x} < I_p + t & \text{(similar)} \\ b, & I_p + t \leq I_{p \rightarrow x} & \text{(brighter)} \end{cases}$$

Train decision tree to maximize information gain:

$$H(P) - H(P_d) - H(P_s) - H(P_b)$$

FAST Performance

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 webcams) have approximately the same pixel rate, the percentages are widely applicable. Approximately 500 features per field are detected.

- On average, 2.26 (for $n = 9$) and 2.39 (for $n = 12$) questions are asked per pixel to determine whether or not it is a feature. By contrast, the handwritten detector asks on average 2.8 questions.

SIFT

- Scale Invariant Feature Transform.
- Detects “scale-space extrema”.
- Highly stable features
- Now widely used in computer vision.

- D.G. Lowe, IJCV Vol. 60(2) 91-110 2004.

SIFT Application: Autostitch



DoG Scale Space

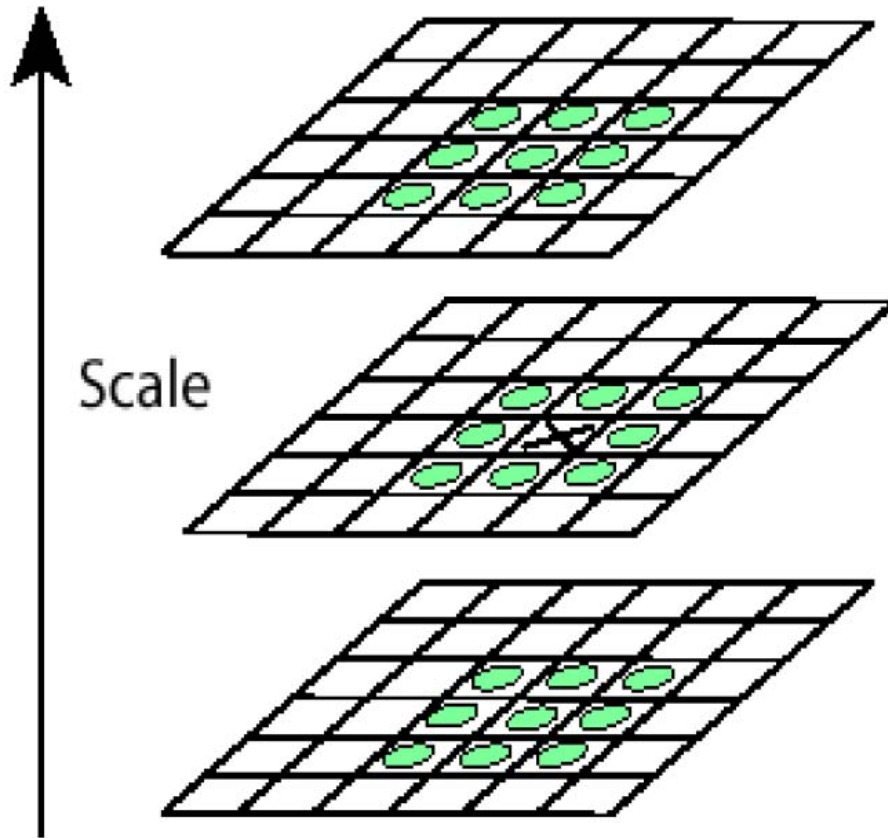


Subtract



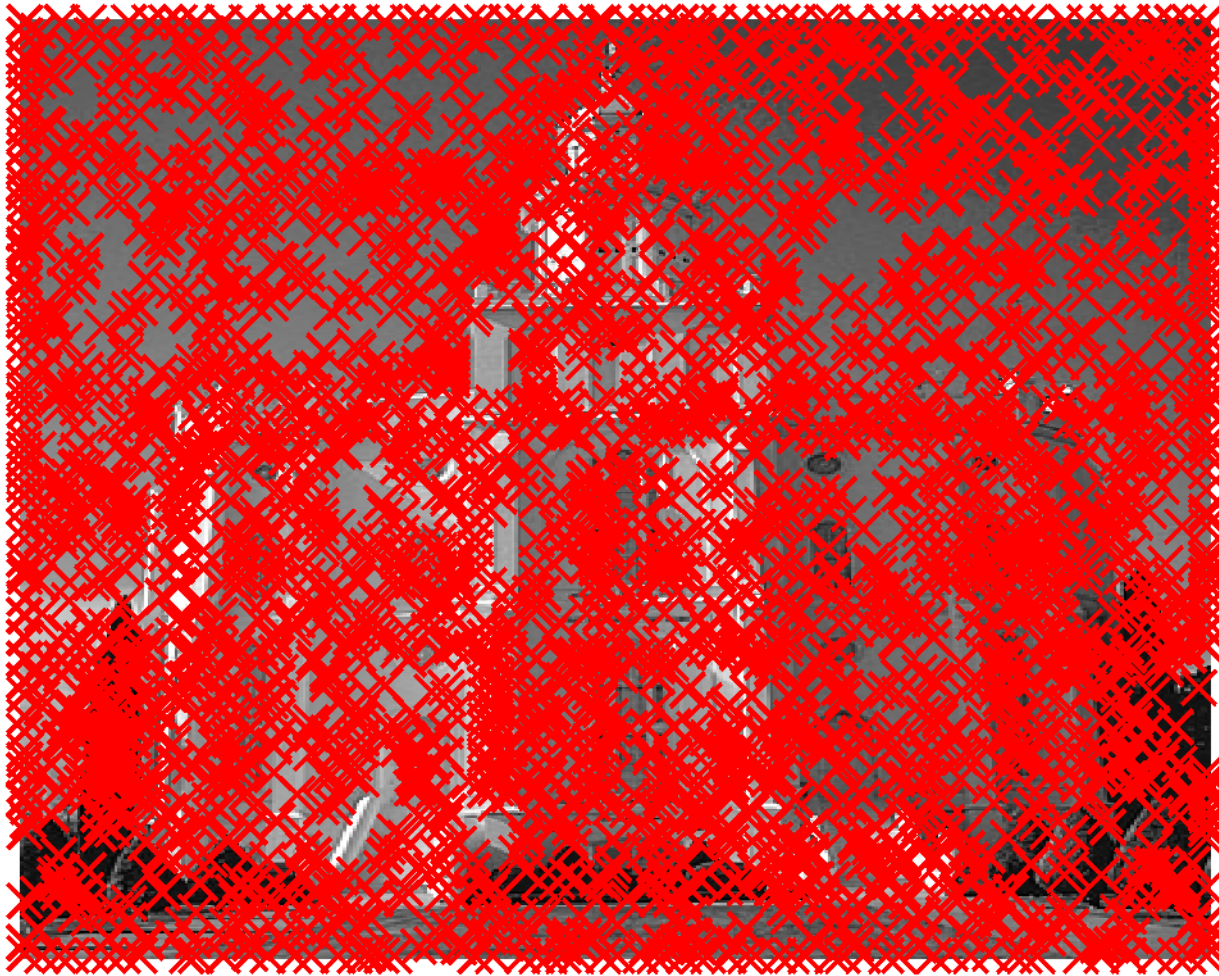
Efficiently
computed
Laplacian scale
space.

Scale space extrema



- Find local maxima and minima in the scale space stack.
- These are SIFT keypoints.

SIFT Keypoints



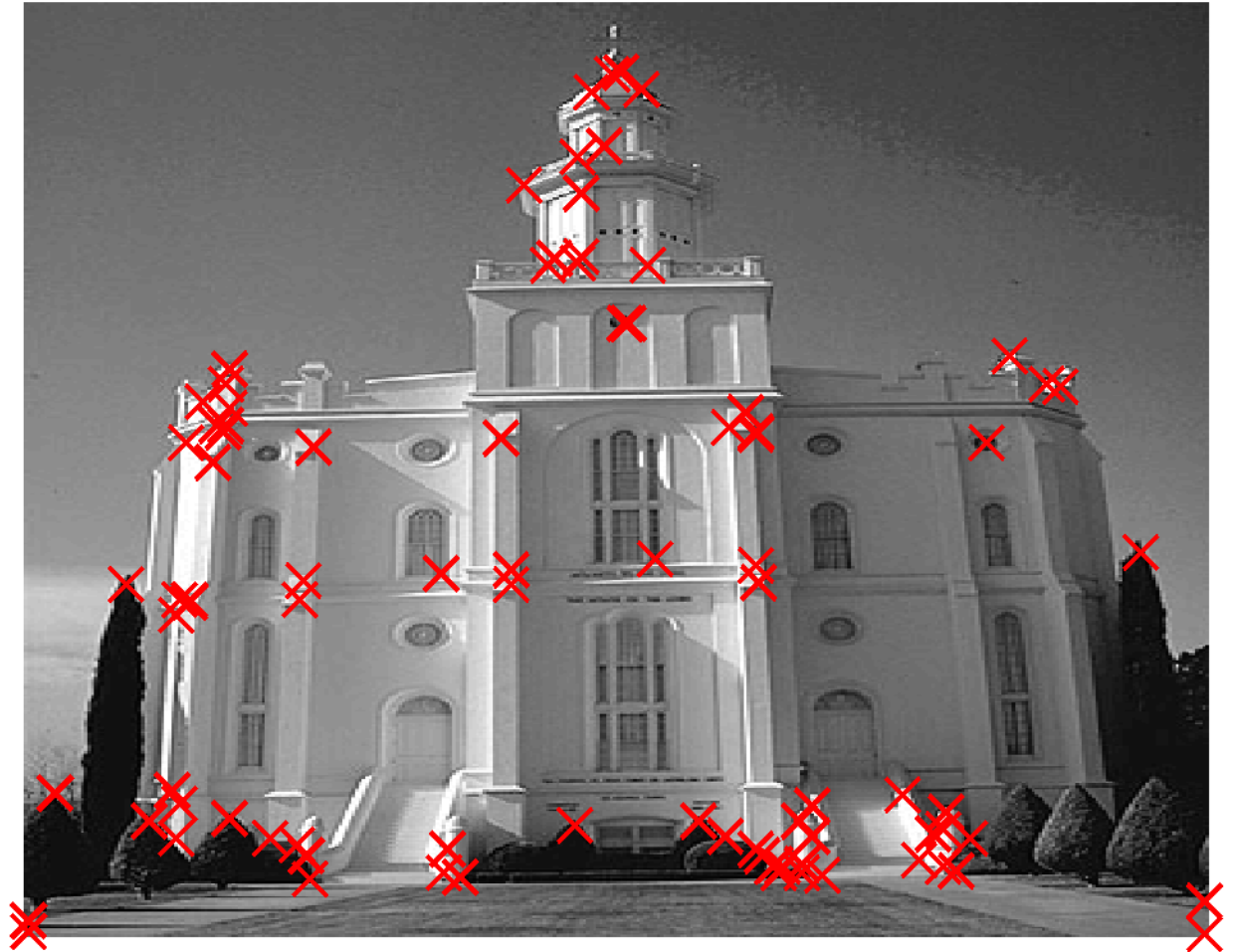
Threshold local variance

- Keep the top 50 %.
- Notice the response along edges.



Threshold Harris cornerness

Low thresholds a cornerness measure in the scale space.



SIFT Features

- Extrema in Laplacian are distinctive (after removing edges)
- Extrema in scale space give scale independence.
- Lowe creates features at each keypoint from the histogram of local edge orientations
- Very stable features for affine matching

Evaluation of Interest-point Detectors?

- Ideas?
- Innovations?

Summary

- Corners and point features
- Various algorithms
 - Moravec
 - Harris (image structure tensor)
 - Sub-pixel accuracy
 - SUSAN
 - FAST
- Example descriptor: SIFT