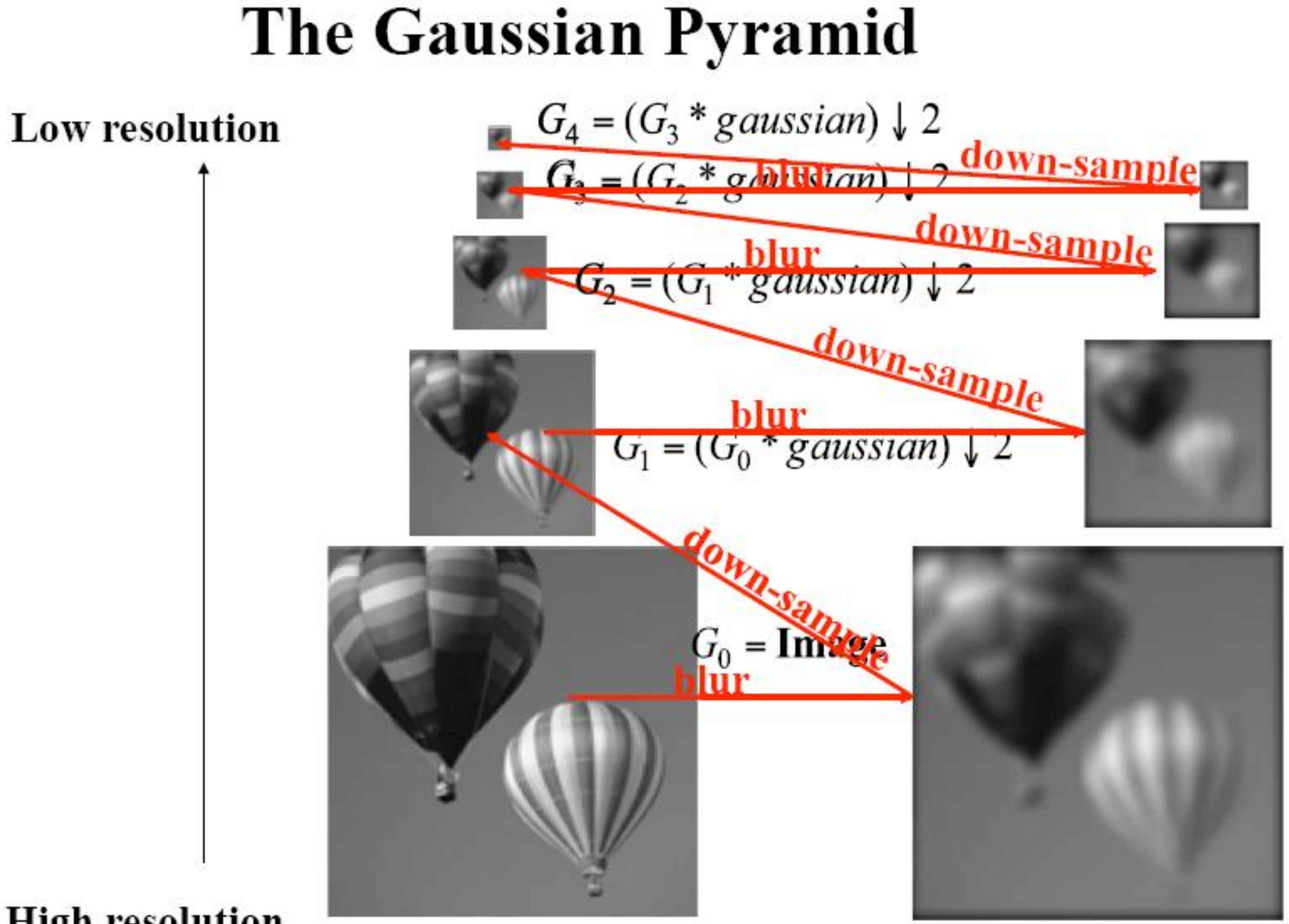
Last Week



Source: Irani

Last Week: Performance

- How can we evaluate edge-detector performance?
  - Probability of false edges
  - Probability of missing edges
  - Error in edge angle
  - Mean squared distance from true edge

Last Week: Summary

- Resize by re-sampling (must pre-filter!!)
- Image pyramids applications
- Edge detection
  - Simple
  - Canny
- Hough Transform

Corner Detection

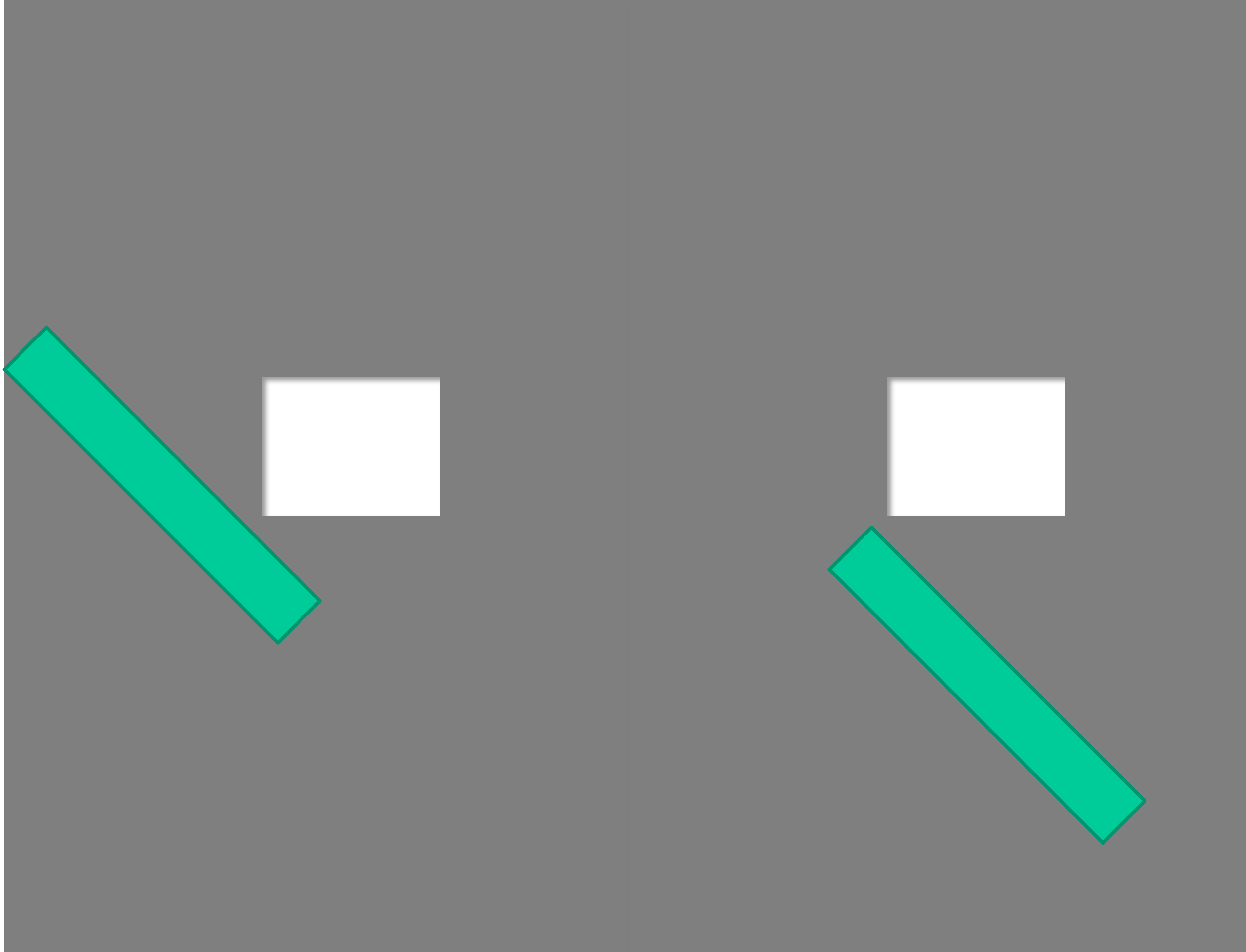
Outline

- Corners and point features
- Moravec operator
- Image structure tensor
- Harris corner detector
- SUSAN
- FAST
- SIFT

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

- 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)



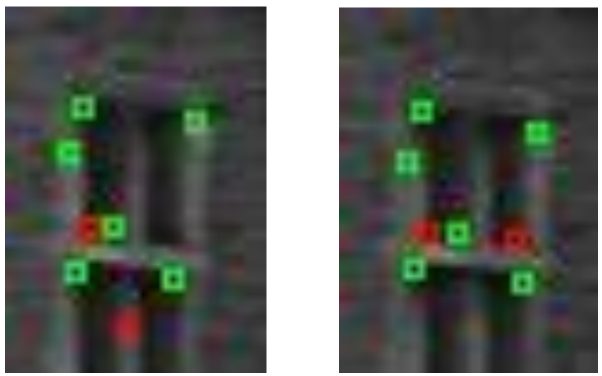
Tracking & Reconstruction

- 2D3 Commercial application: Boujou
  - Match-moving for special effects
  - Computes a point-cloud + camera poses

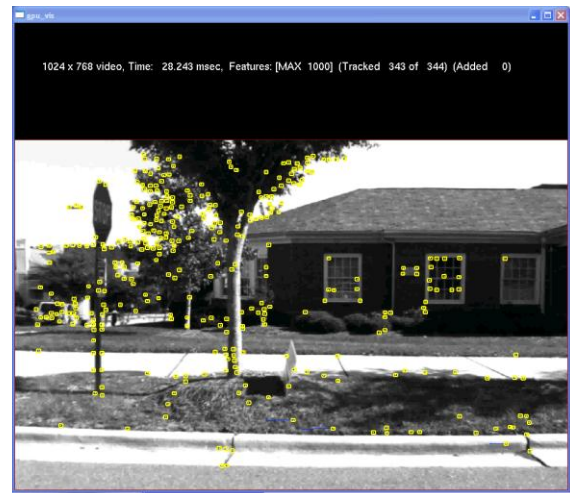
- UNC city-scanning (video) (Pollefeys et al.)

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



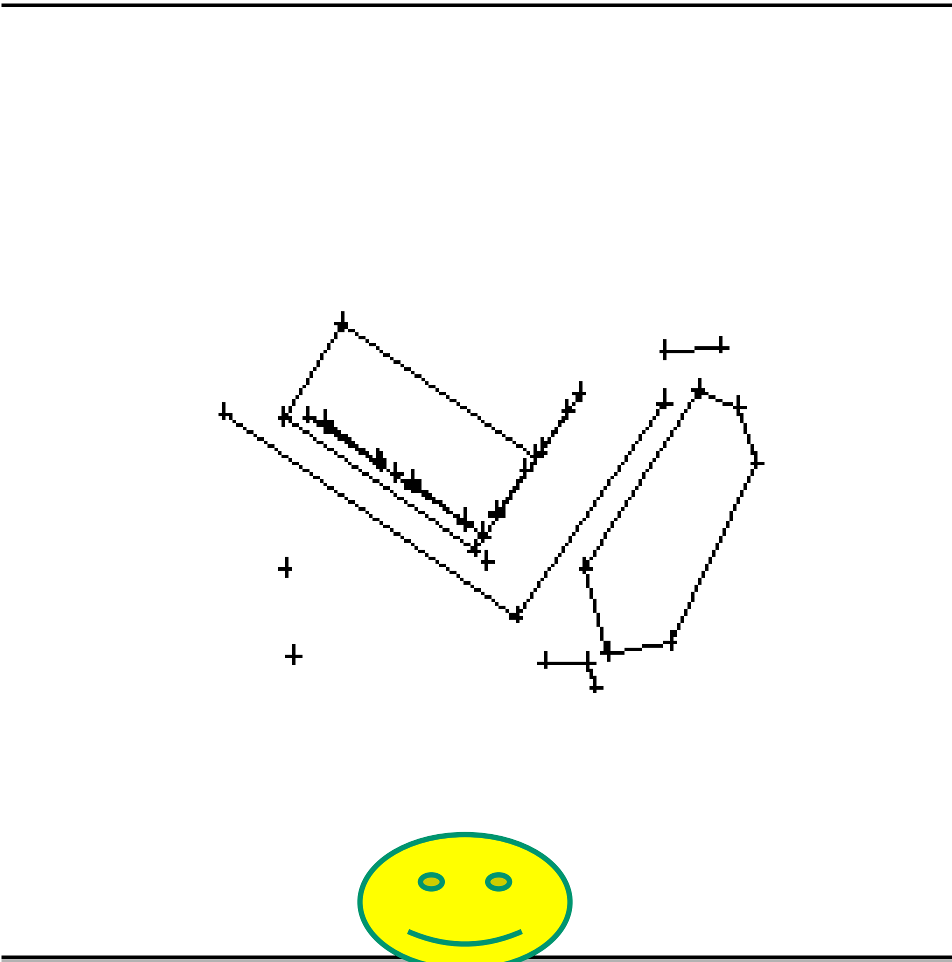
Extract features in first images and then try to find same feature back in next view

What is a good feature?

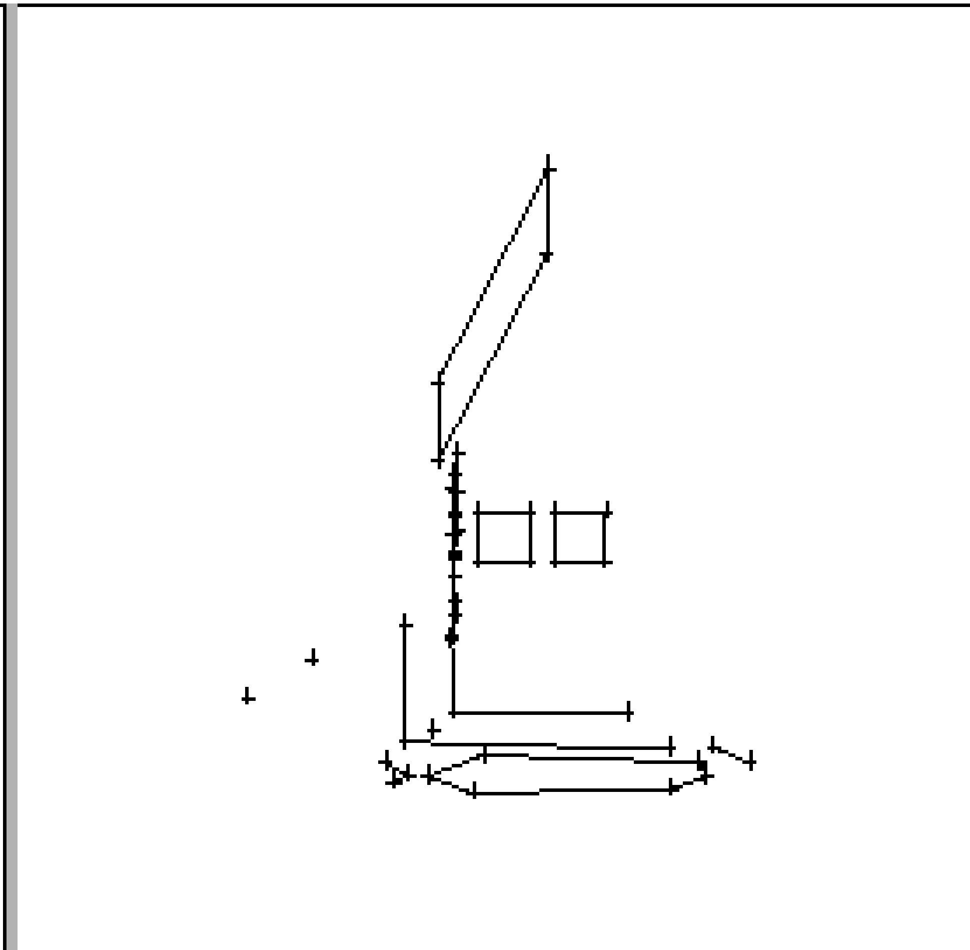
From Marc Pollefeys

Stereo Example

Stereo Example



top view



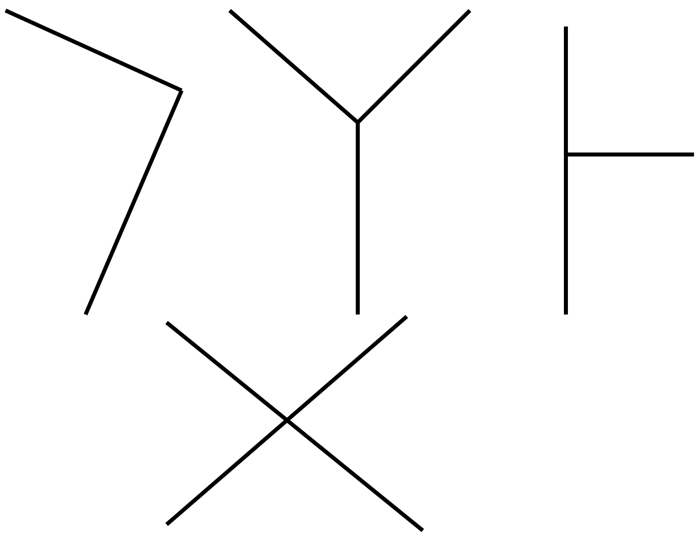
lateral view

"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

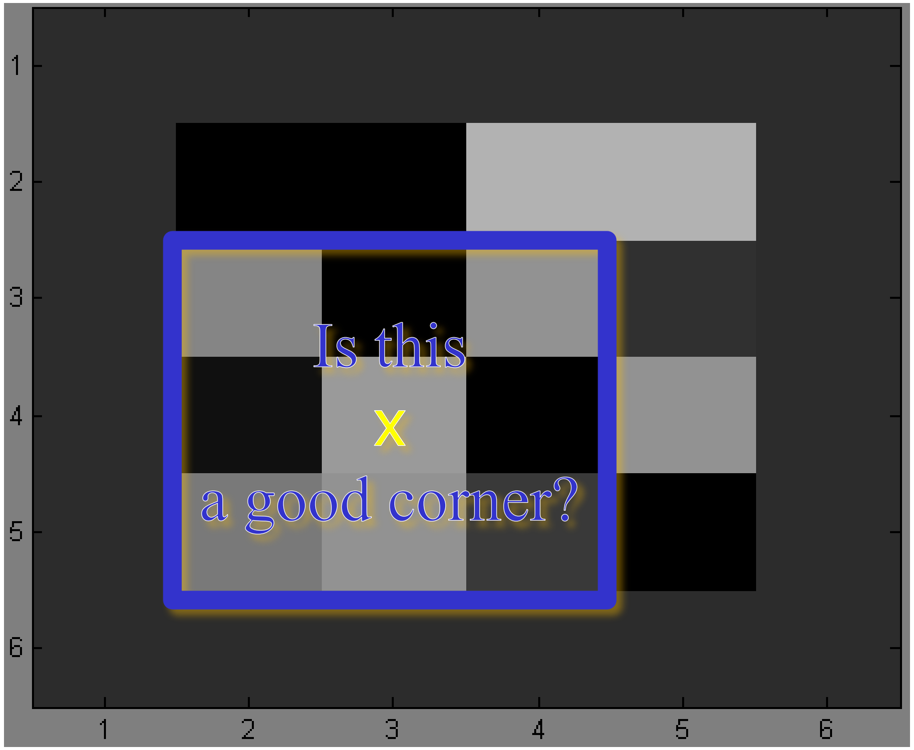
- Edge and line intersections provide recognizable features



Intersections can be stable or unstable

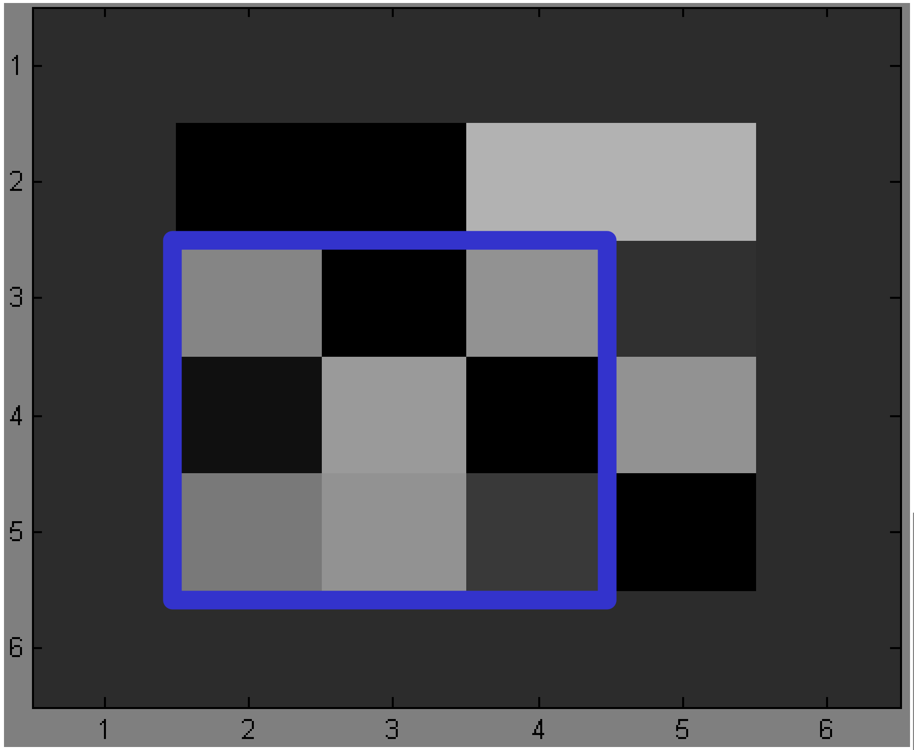
Moravec "Interest" Operator

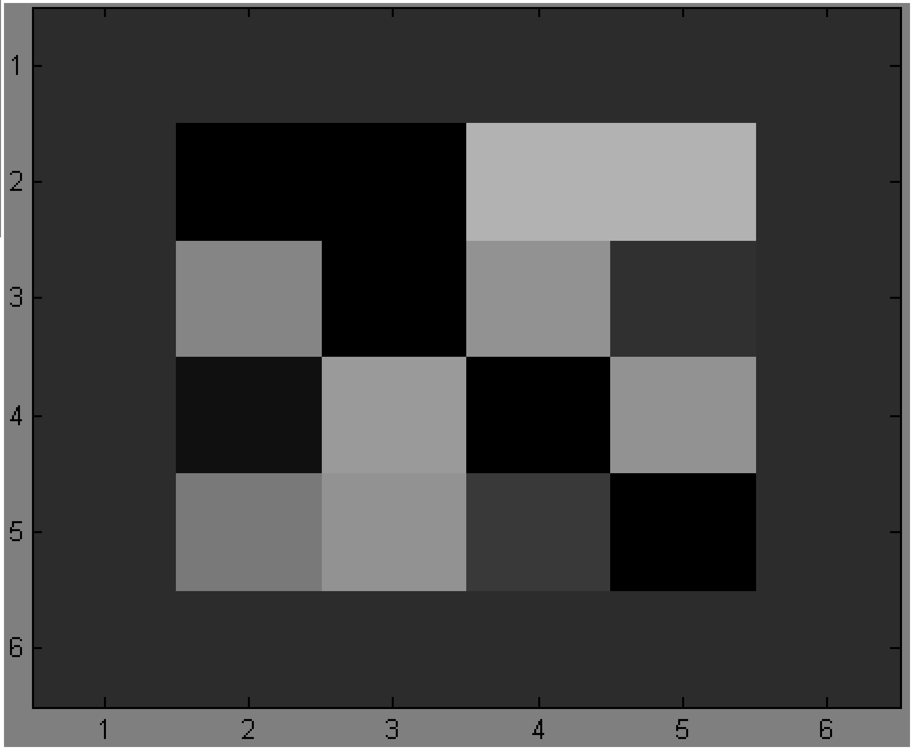
- Use a window surrounding each pixel as its own matching template
- Tests local *autocorrelation* of the image:
  - SSD = Sum of Squared Differences
- Good matches in any direction
  - Flat image region
- Good matches in only one direction
  - Linear feature or edge
- No good matches in any direction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



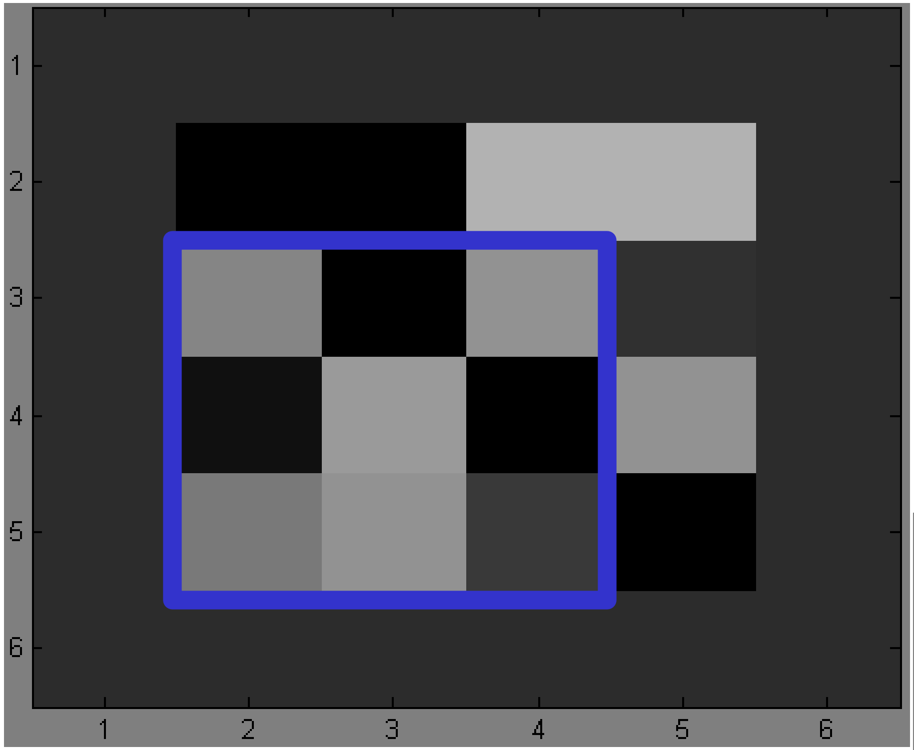
Neighborhood stands out?

Check SSDs

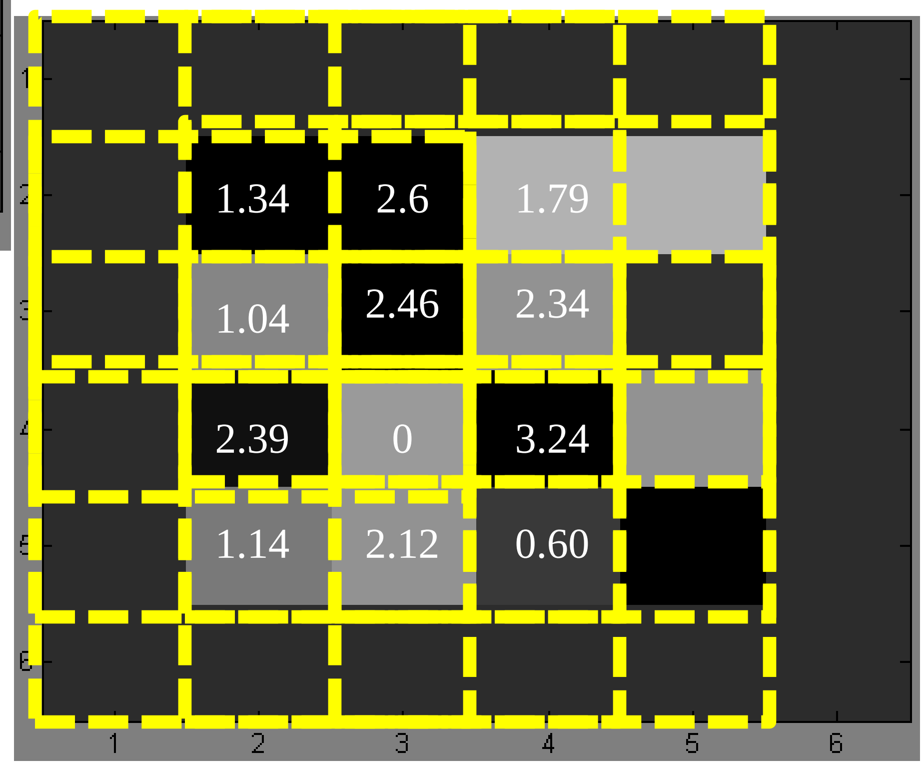





```
sum(sum(  (patch - Ipad(rowStart:rowStart+2, colStart:colStart+2) ).^2 ));
```


1
2
3
4
5
6
1
2
3
4
5
6

Check SSDs


1.34
2.6
1.79
1.04
2.46
2.34
2.39
0
3.24
1.14
2.12
0.60
1
2
3
4
5
6

Implementation

- Match quality for displacements (Δx, Δy) uses L_2 norm:

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

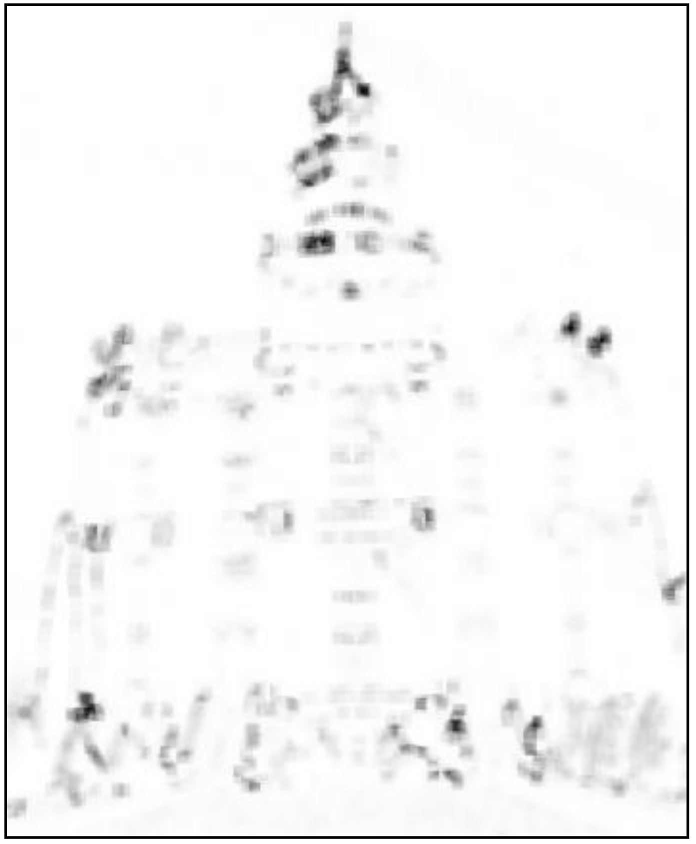
- *w* is a window function
  - e.g. constant or Gaussian
- Cornerness is min ε over the eight-neighbourhood *N*, excluding (0, 0)

Moravec "Interest" Operator

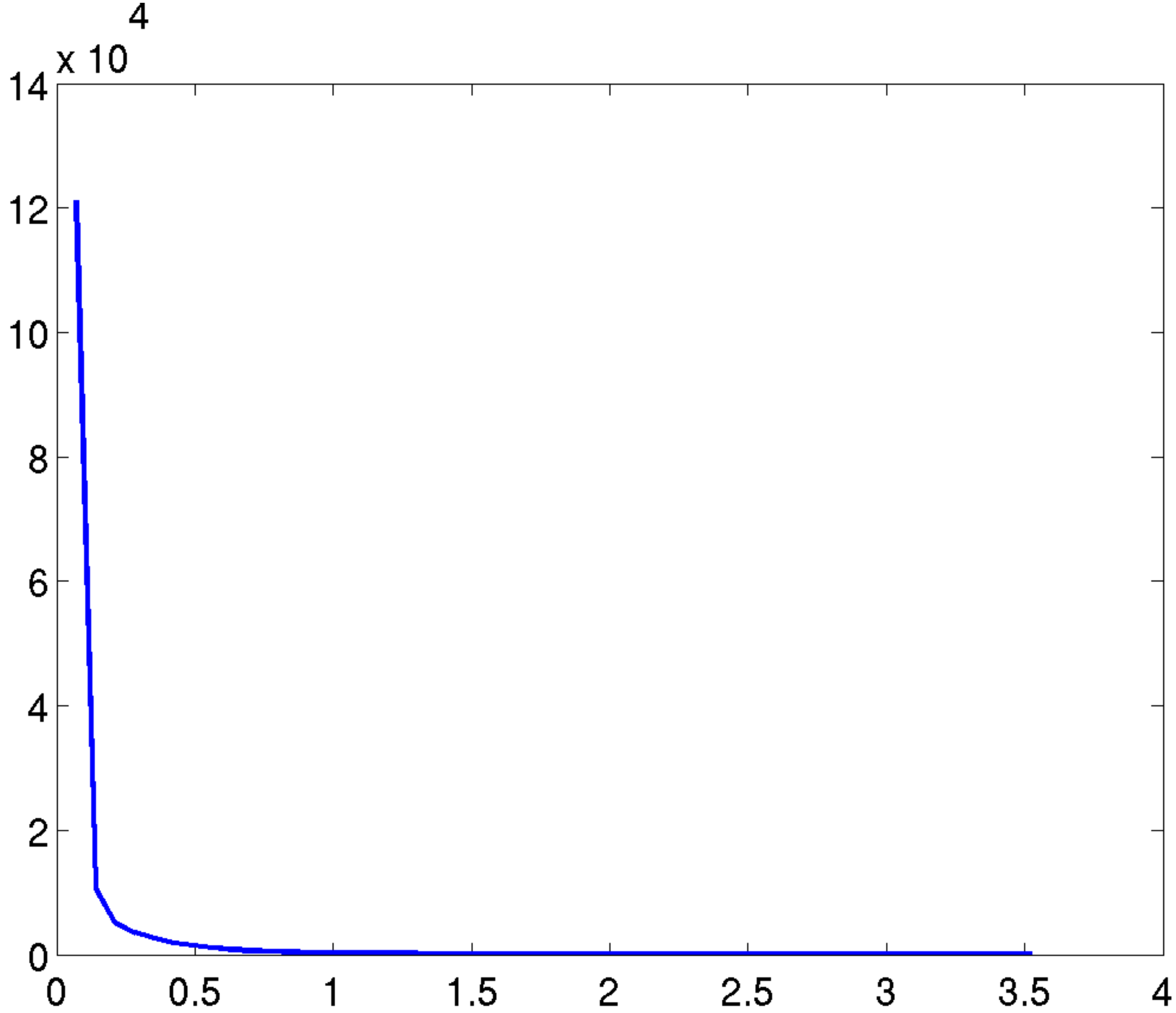
- Use a window surrounding each pixel as its own matching template *T*.
- Tests local *autocorrelation* of the image: SSD
- Good matches in any direction
    - Flat image region
- Good matches in only one direction
    - Linear feature or edge
- No good matches in any direction
    - Distinctive point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

Cornerness

- Using 7x7 matching window.



Cornerness distribution



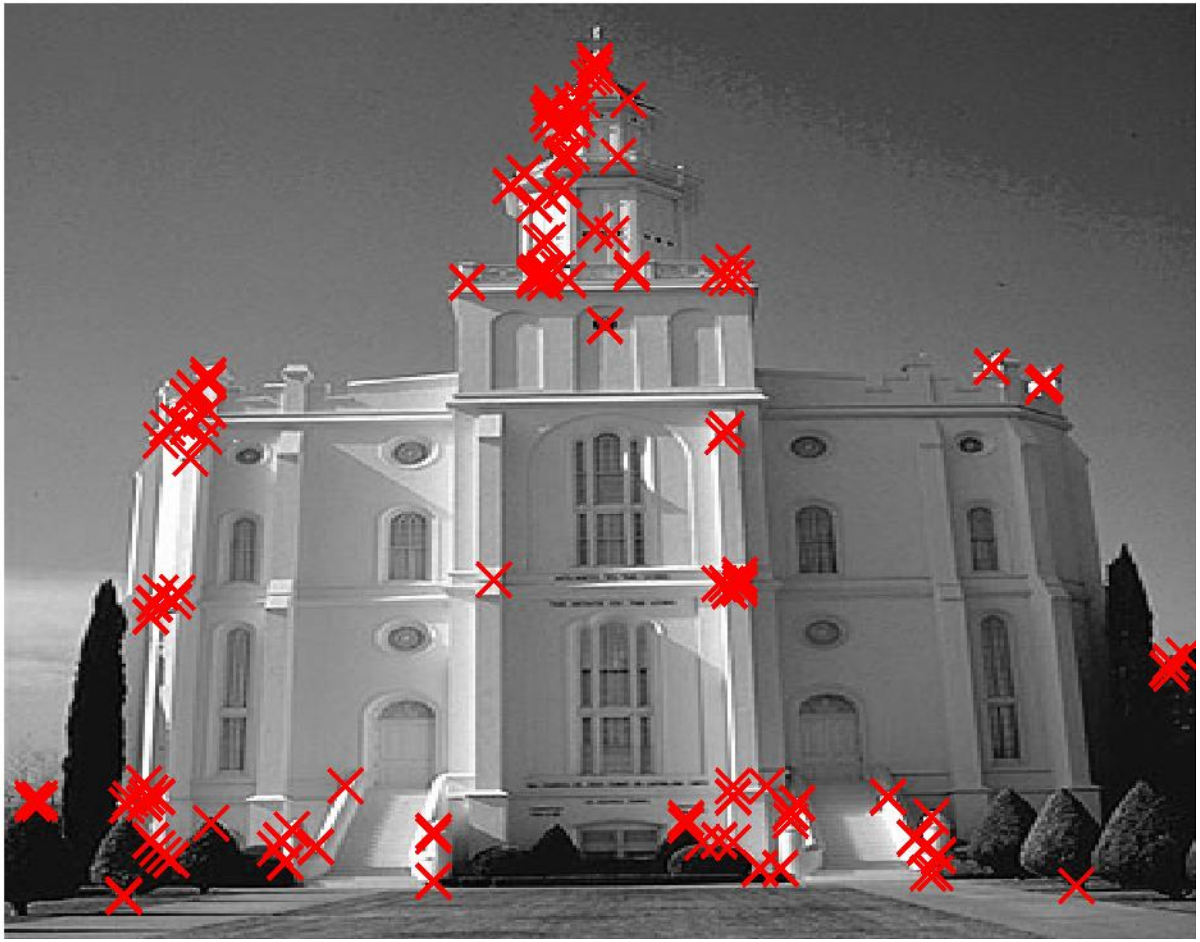
Pixels

Cornerness

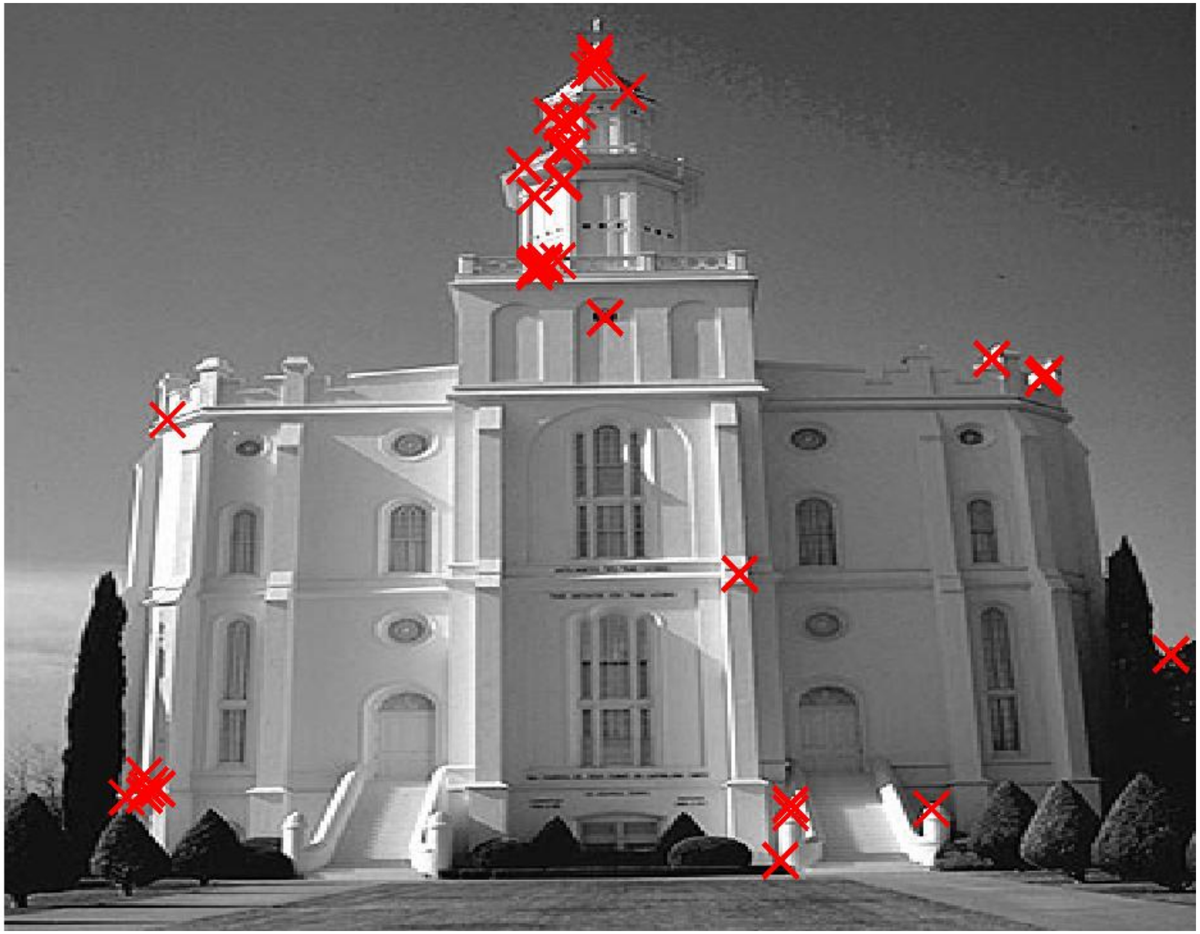
Non-maximal suppression

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

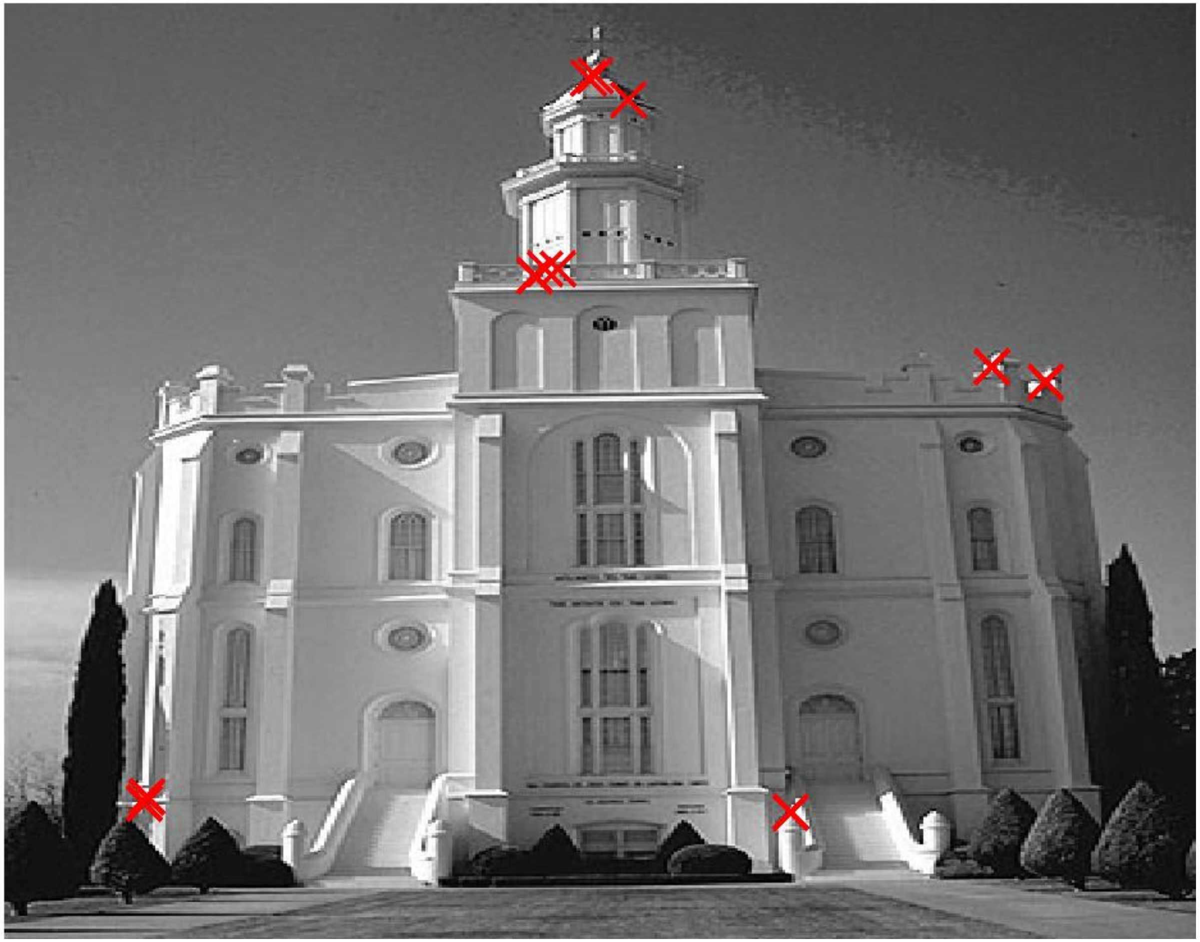
Threshold T = 1



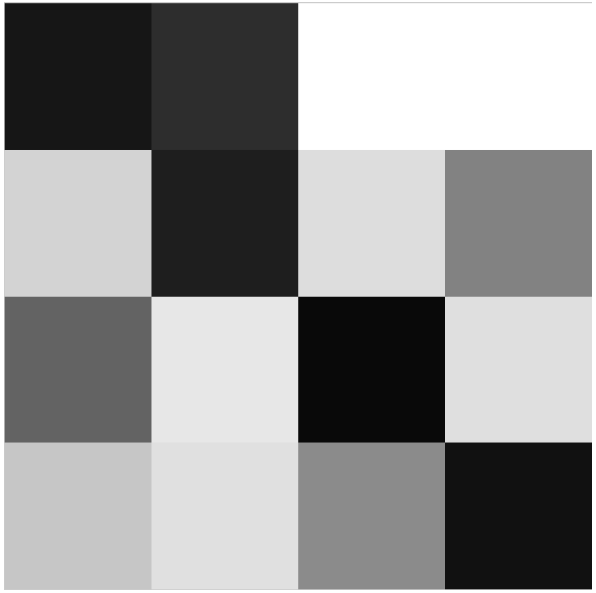
Threshold T = 2



Threshold T = 3

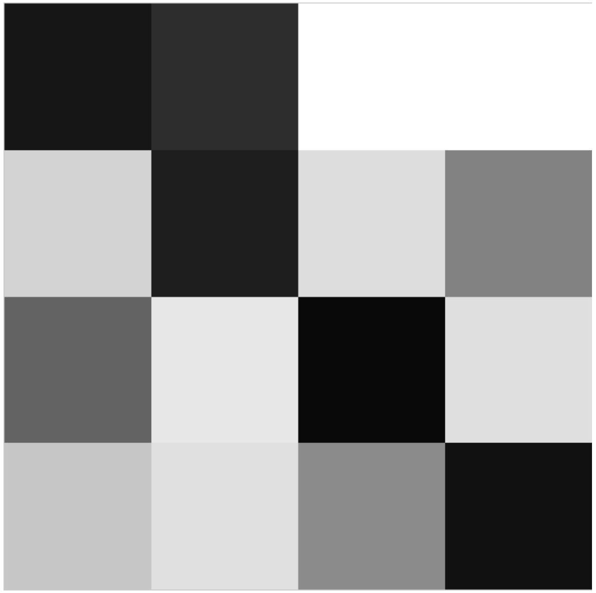


Non-max Suppression Efficiently?



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

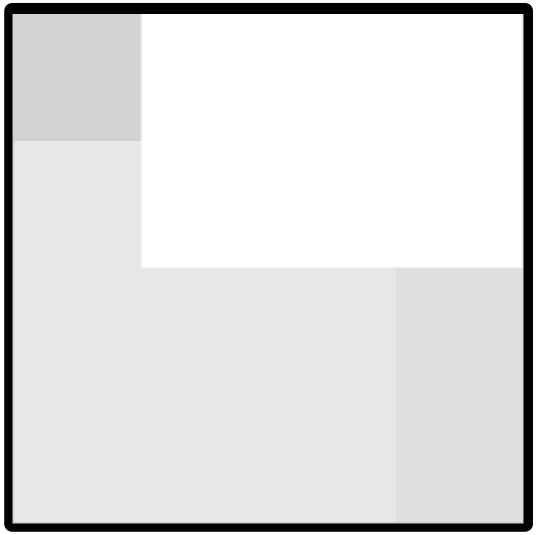
Non-max Suppression by Dilation?



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"

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

$$\underline{I(x+\Delta x, y+\Delta y)} \approx I(x,y) + \left[\frac{\partial I(x,y)}{\partial x}, \frac{\partial I(x,y)}{\partial y}\right]\begin{bmatrix}\Delta x \\ \Delta y\end{bmatrix}$$

(See online Taylor series expansion/approximation)

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

$$\underline{I(x+\Delta x, y+\Delta y)} \approx I(x,y) + \left[\frac{\partial I(x,y)}{\partial x}, \frac{\partial I(x,y)}{\partial y}\right]\begin{bmatrix}\Delta x\\ \Delta y\end{bmatrix}$$

$$\varepsilon(x,y) =$$

$$\sum_{x_i\in N}\sum_{y_i\in N}\left(I(x_i,y_i) - I(x_i,y_i) + \left[\frac{\partial I(x,y)}{\partial x}, \frac{\partial I(x,y)}{\partial y}\right]\begin{bmatrix}\Delta x\\ \Delta y\end{bmatrix}\right)^2$$

$$\sum_{x_i\in N}\sum_{y_i\in N}\left(\left[\frac{\partial I(x,y)}{\partial x}, \frac{\partial I(x,y)}{\partial y}\right]\begin{bmatrix}\Delta x\\ \Delta y\end{bmatrix}\right)^2$$ (Note : $\mathbf{u}^2 = \mathbf{u}^T\mathbf{u}$)

$$\sum_{x_i\in N}\sum_{y_i\in N}[\Delta x, \Delta y]\left(\begin{bmatrix}\frac{\partial I}{\partial x}\\ \frac{\partial I}{\partial y}\end{bmatrix}\left[\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}\right]\right)\begin{bmatrix}\Delta x\\ \Delta y\end{bmatrix}$$

General Form

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

$$I(x+\Delta x, y+\Delta y) \approx I(x,y) + \left[\frac{\partial I(x,y)}{\partial x}, \frac{\partial I(x,y)}{\partial y}\right]\begin{bmatrix}\Delta x \\ \Delta y\end{bmatrix}$$

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

General Form

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

- <…> denotes average over the window *N*.
- *S* is the image structure tensor
- a.k.a 'Harris' Matrix

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

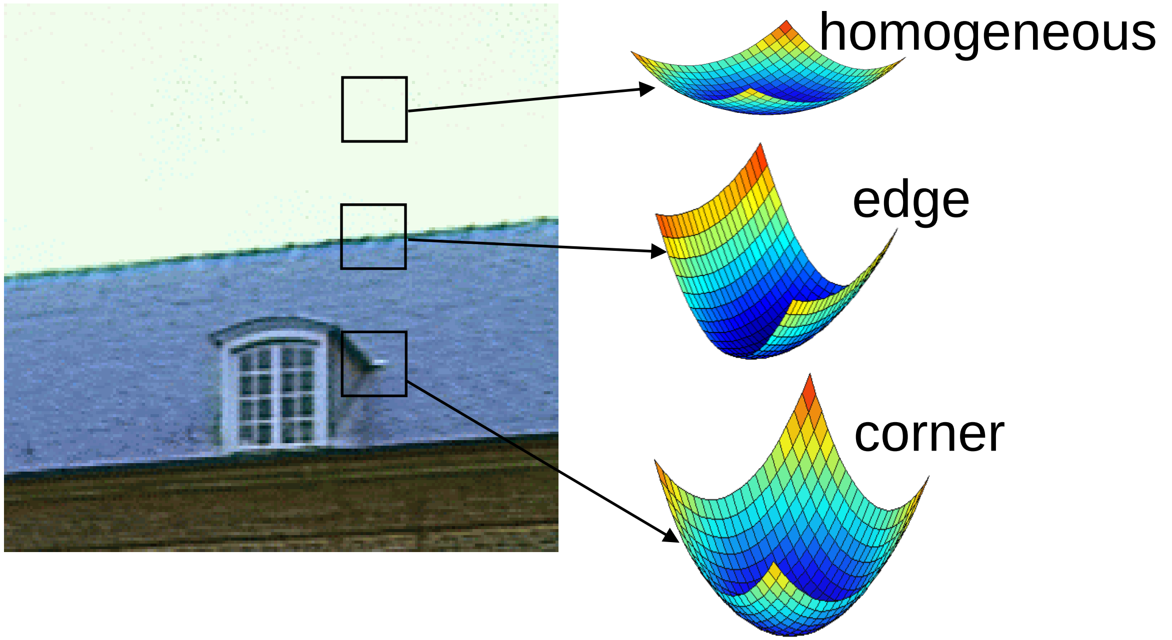
Structure tensor

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

- They are the solutions to

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

Principal Axes (i.e. Eigenvectors)
(Note: see Simon Prince's SVD slides online)



i.e. maximize smallest eigenvalue of S

From Marc Pollefeys

Harris Corner Detector

(Example Video)

- Defines cornerness as size of smallest eigenvalue, or

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

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

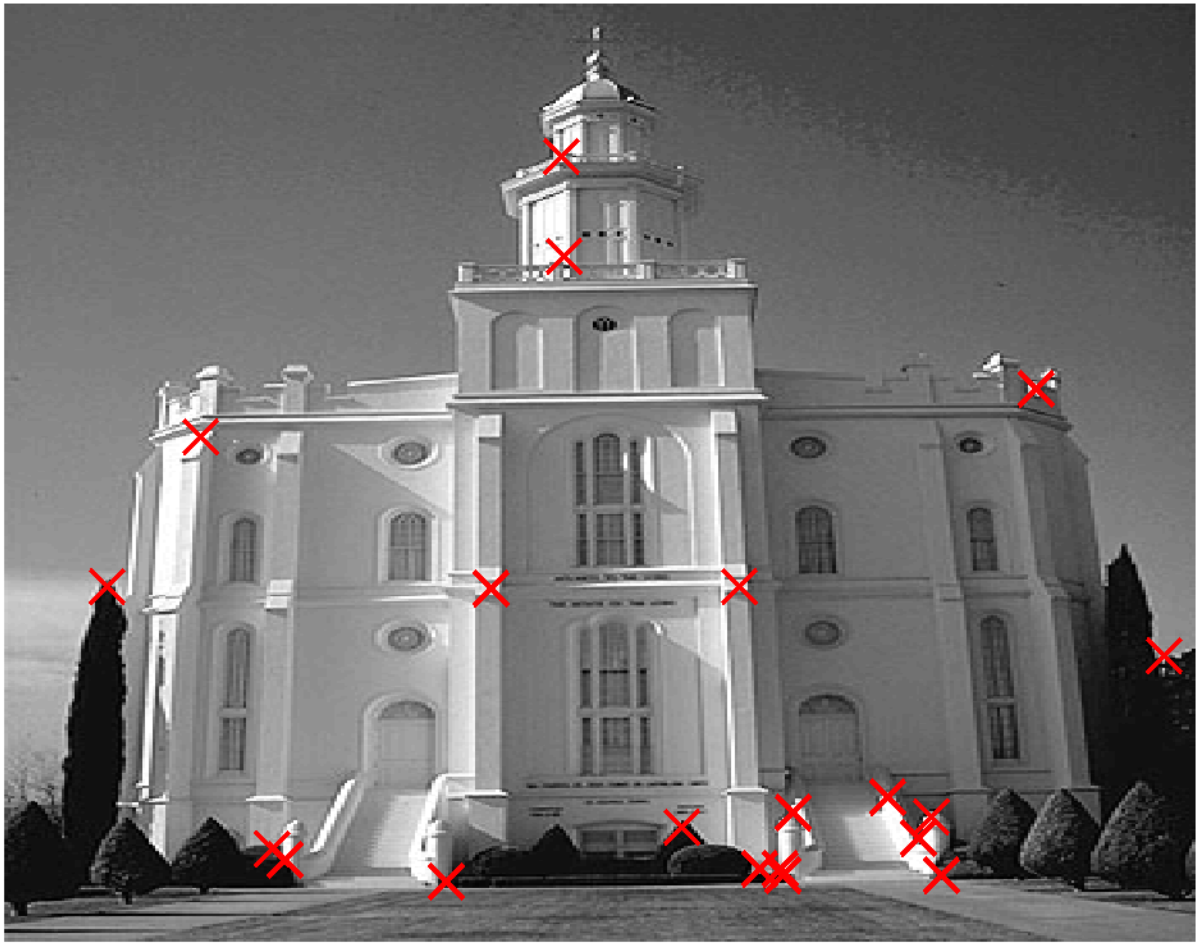
$$C = \lambda_1 \lambda_2 / (\lambda_1 + \lambda_2)$$

- Non-maximal suppression and thresholding as before

Cornerness and NMS

Sigma = 1, Threshold T=0.1

Sigma = 1, Threshold T=0.15



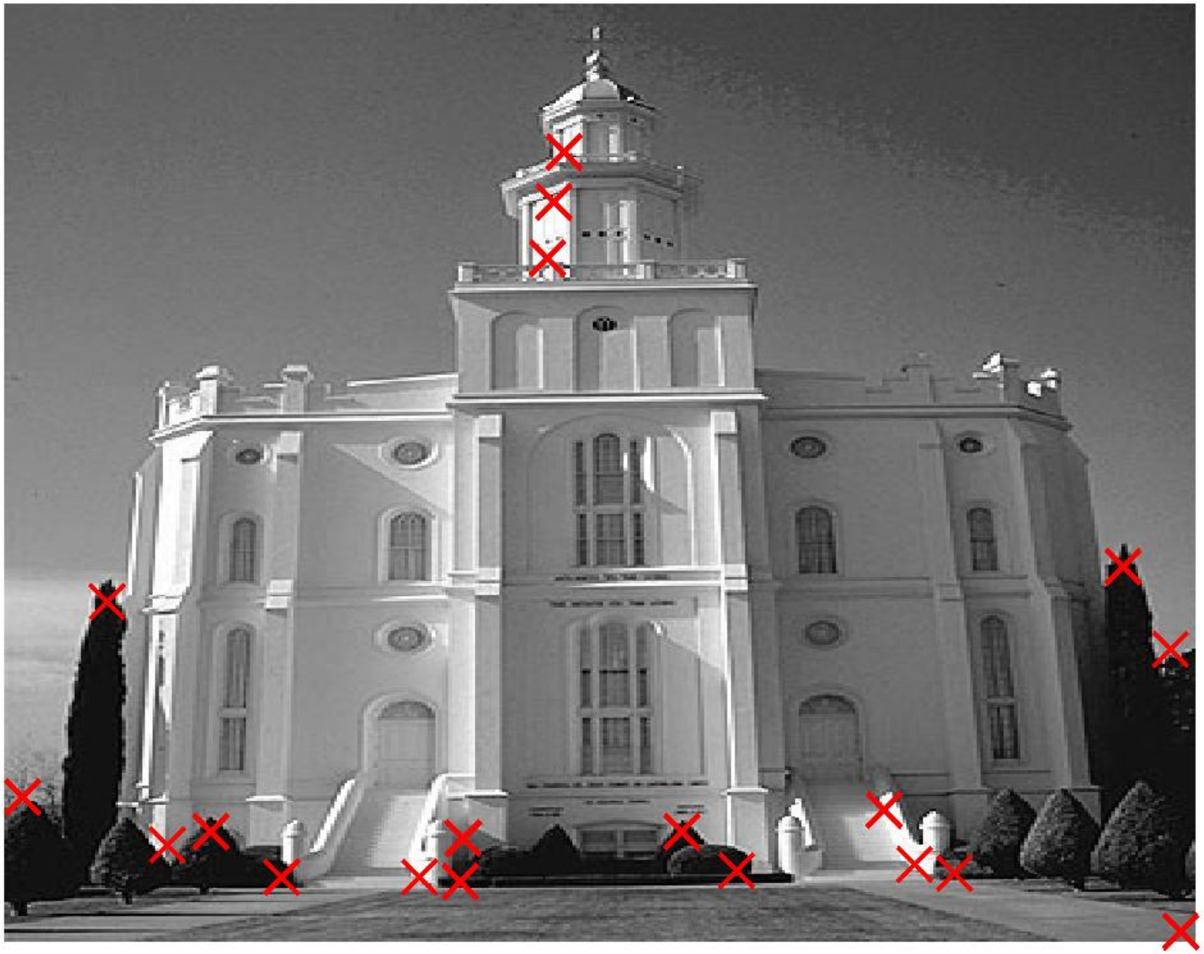
Sigma = 1, Threshold T=0.2

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
    - $R = \det(S) - k\mathrm{Tr}^2(S)$
- 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

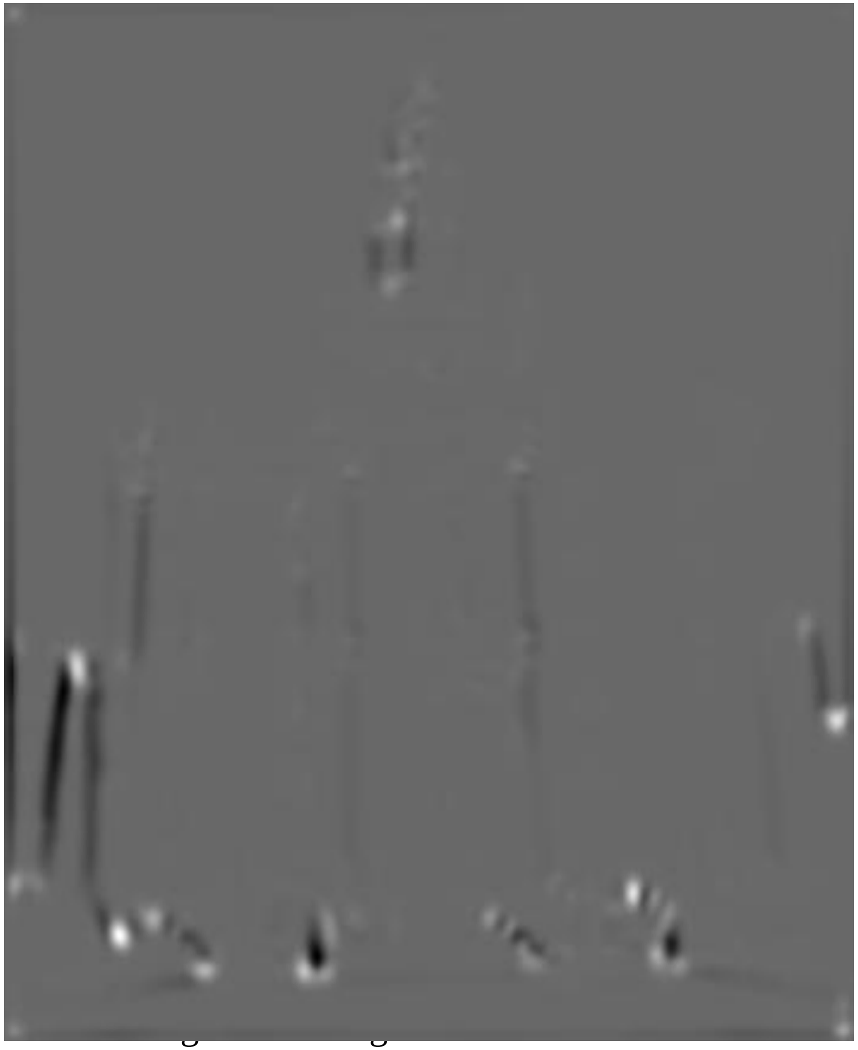
Sigma = 5, Threshold T=0.004



Sigma = 5, Threshold T=0.007

Harris and Stephens

$k = 0.02$



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

SUSAN

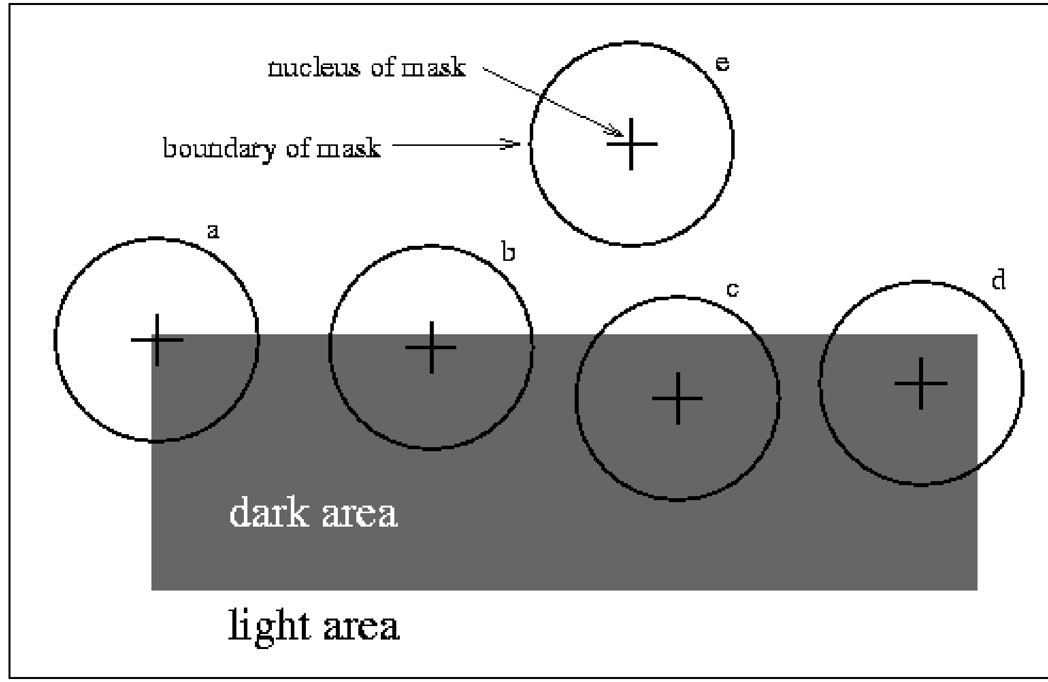
- Method for edge and corner detection
- No image derivatives
- Insensitive to noise

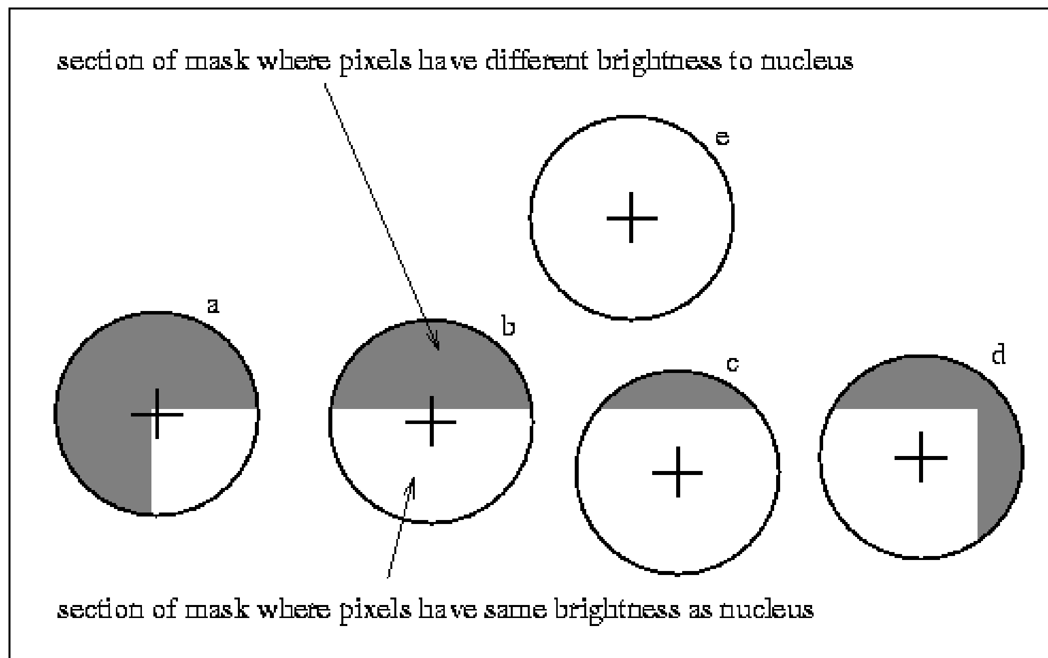
- Smith & Brady, IJCV, 23(1):45-78, 1997

USAN

"Univalue Segment Assimilating Nucleus"

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

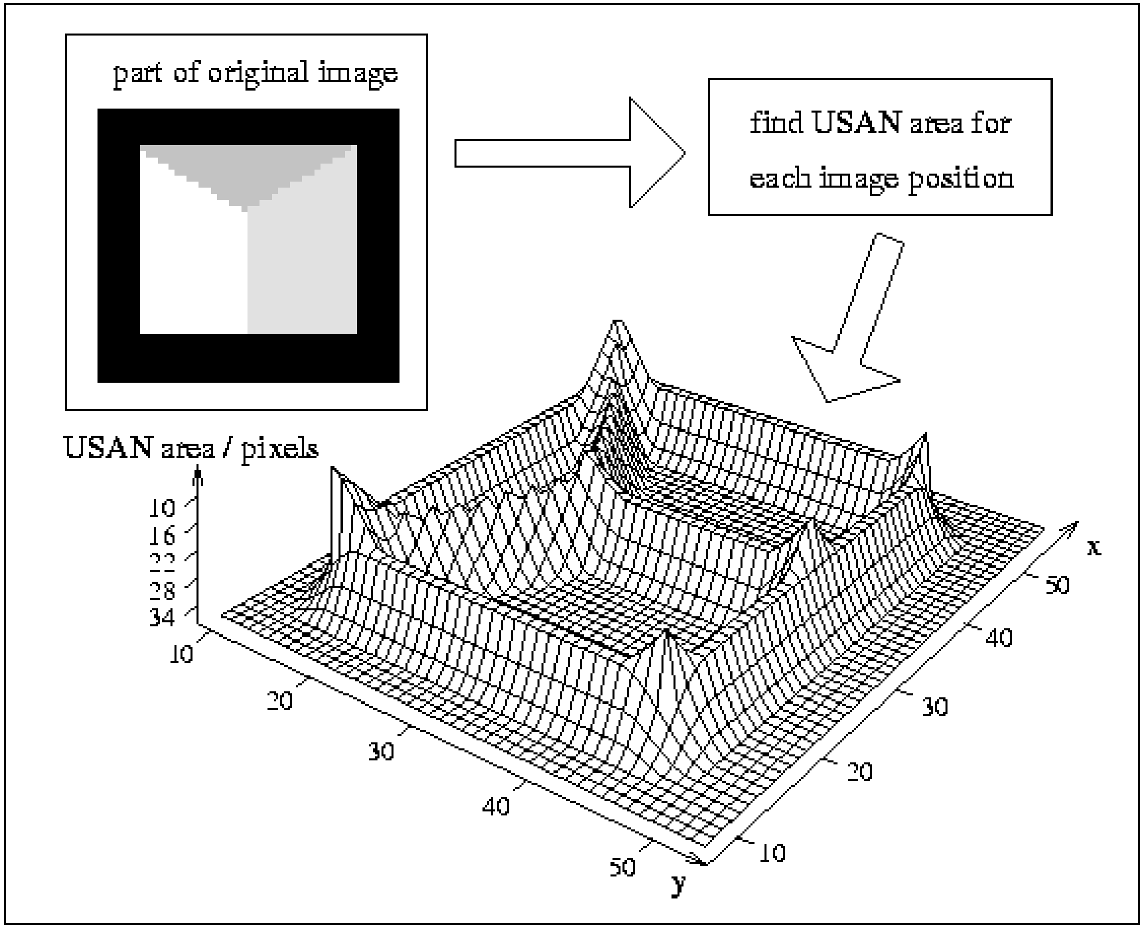




Edges and Corners

- 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



part of original image
find USAN area for each image position
USAN area / pixels
10
16
22
28
34
10
20
30
40
50
y
10
20
30
40
50
x

Implementation

- Circular mask M with radius 3.4 ($|M| = 37$ pixels).
- The nucleus is the centre pixel r_0
- U is the USAN area.
- C is the corner/edge strength

```
  0011100
  0111110
  1111111
M=1111111
  1111111
  0111110
  0011100
```

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

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

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

t=3|M|/4 for edge detection

t=|M|/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

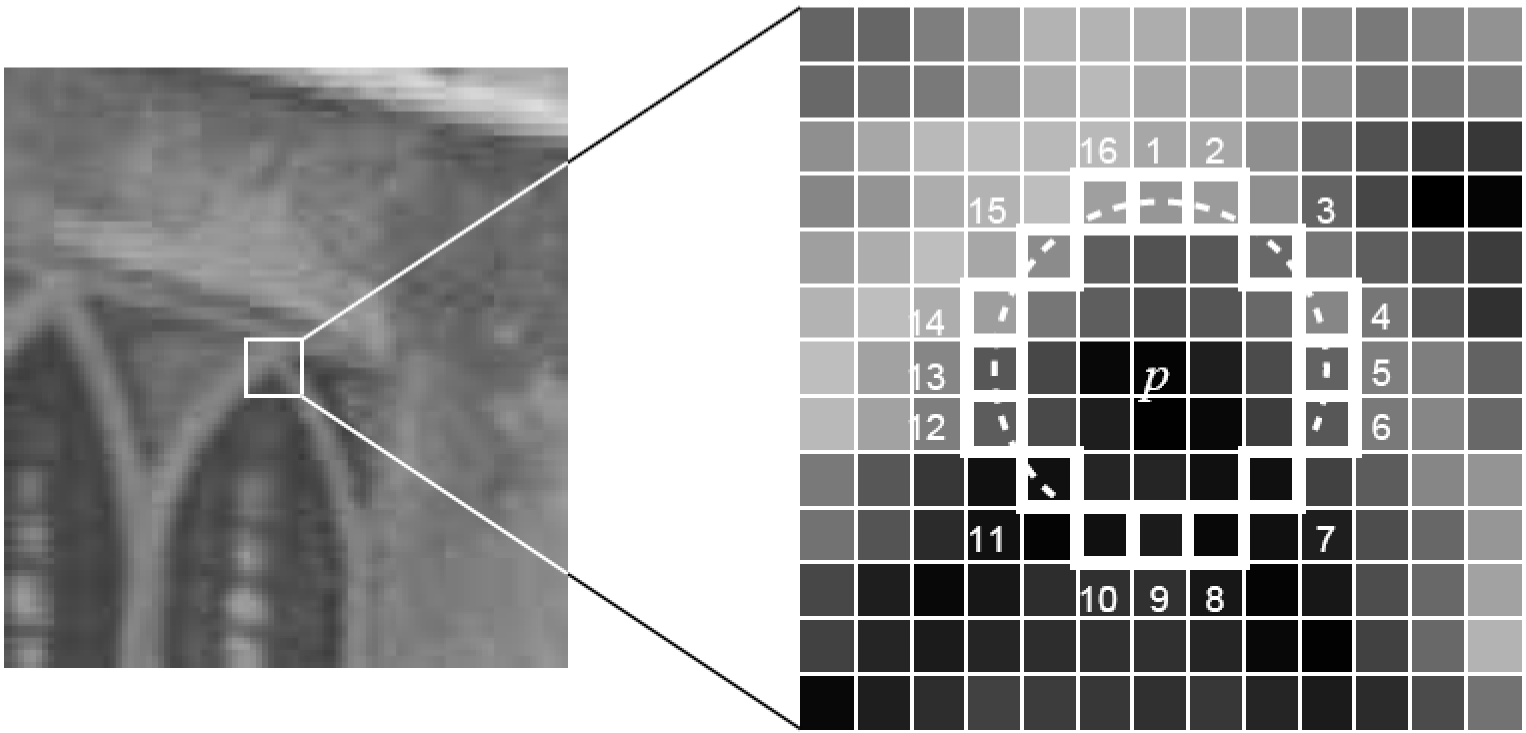
Machine learning for high-speed corner detection
Rosten & Drummond, ECCV 2006

- 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

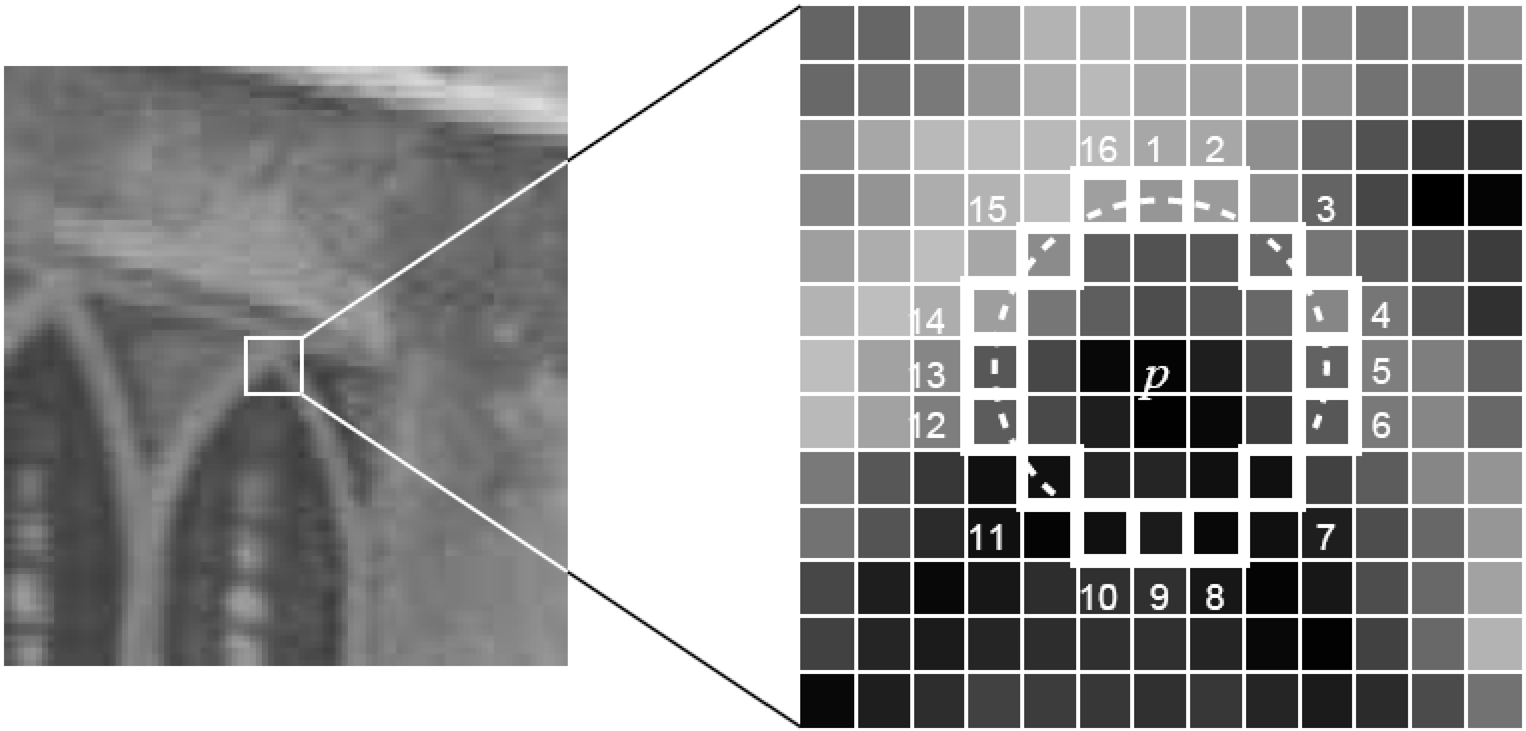
$$C = \min_{P} \left(f_P - f_C\right)^2 + \left(f_{P'} - f_C\right)^2$$



From Rosten & Drummond

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* → *x*) can have one of three states:

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

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 web-cams) 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.

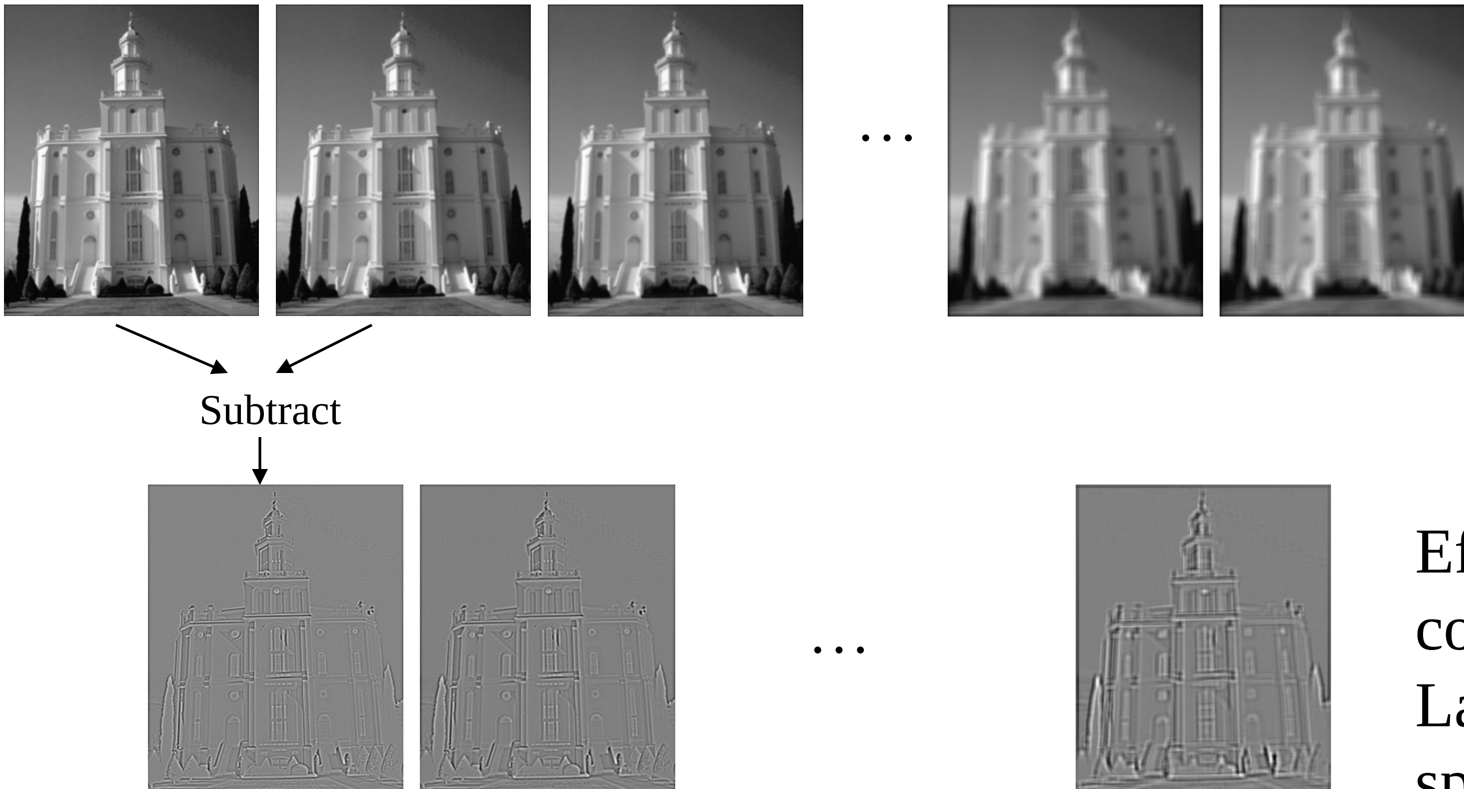
SIFT

- 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.

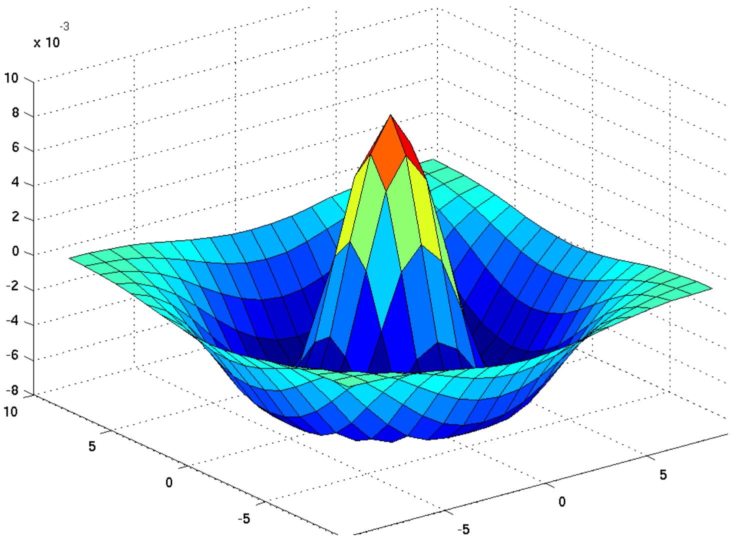
SIFT Application: Autostitch

DoG Scale Space

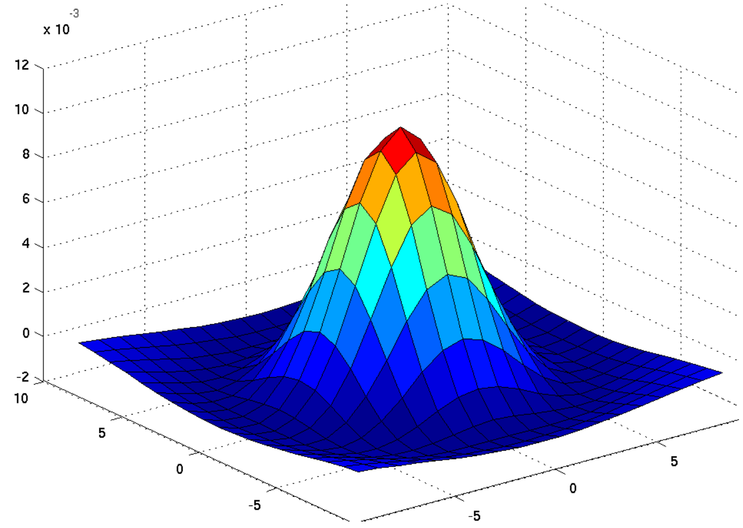


Efficiently computed Laplacian scale space.

LoG vs DoG




```
k=LoG(9,3);
surf(x,y,-k);
```




```
k= GausKern(9,3)-GausKern(9,4.8);
surf(x,y,-k);
```

$$G(x, y, k\sigma) - G(x, y, \sigma) \approx (k-1)\sigma^2 \nabla^2 G$$

This shows that when the difference-of-Gaussian function has scales differing by a constant factor it already incorporates the σ^2 scale normalization required for the scale-invariant Laplacian.

- From Lowe, 2004

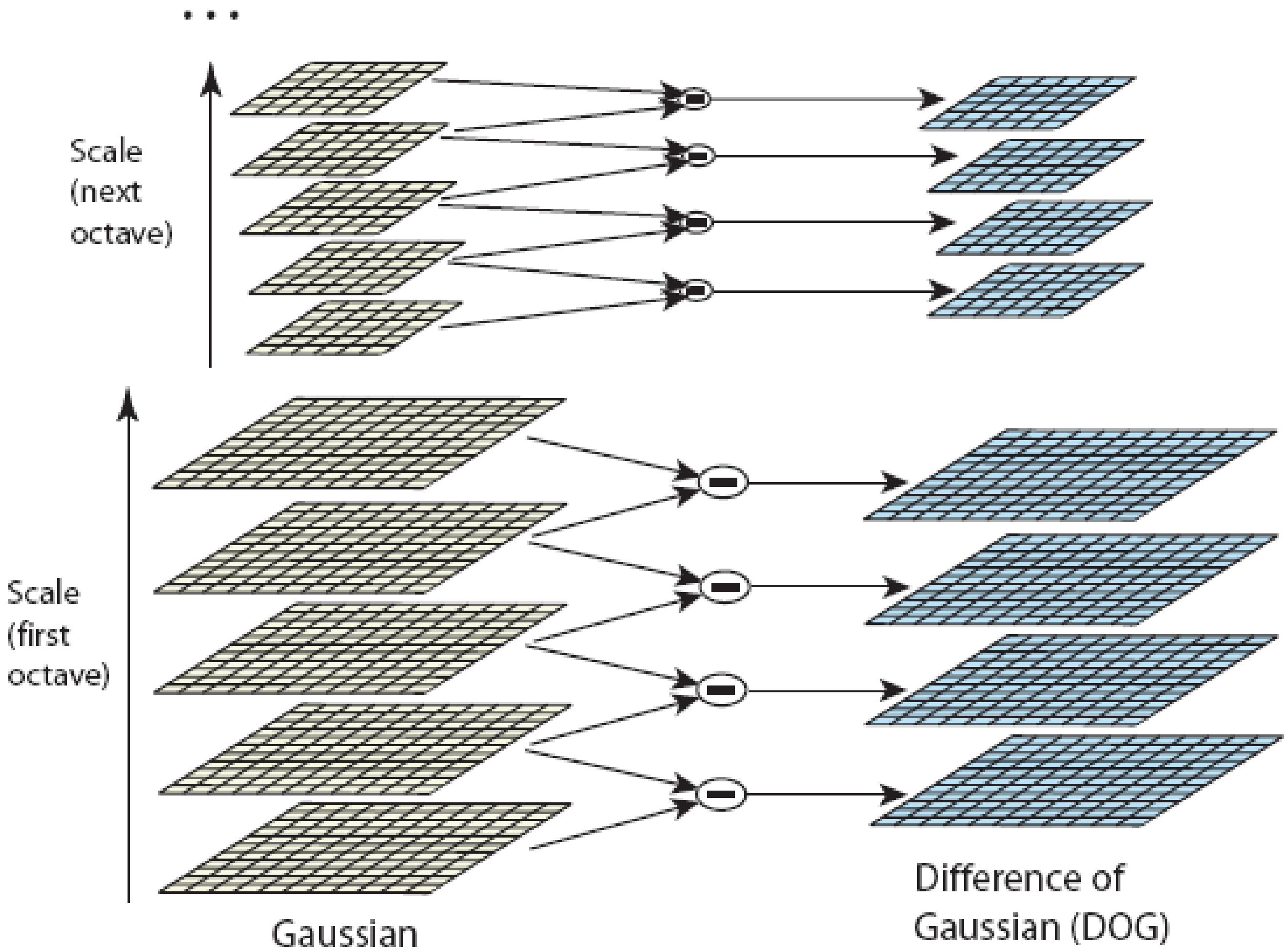
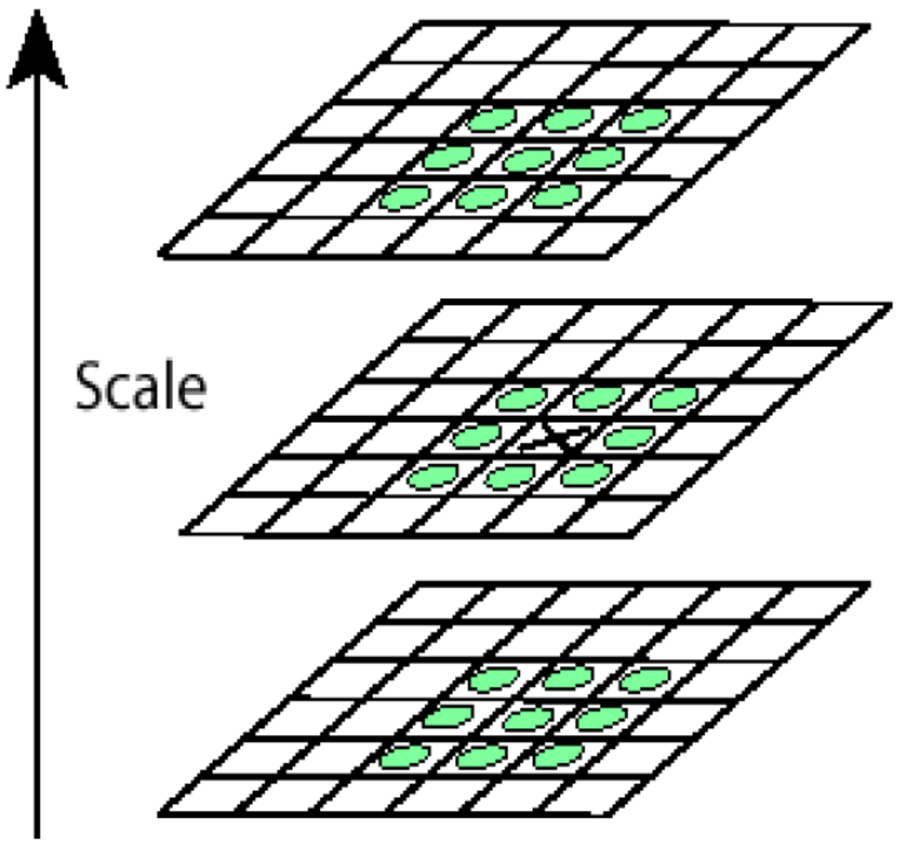


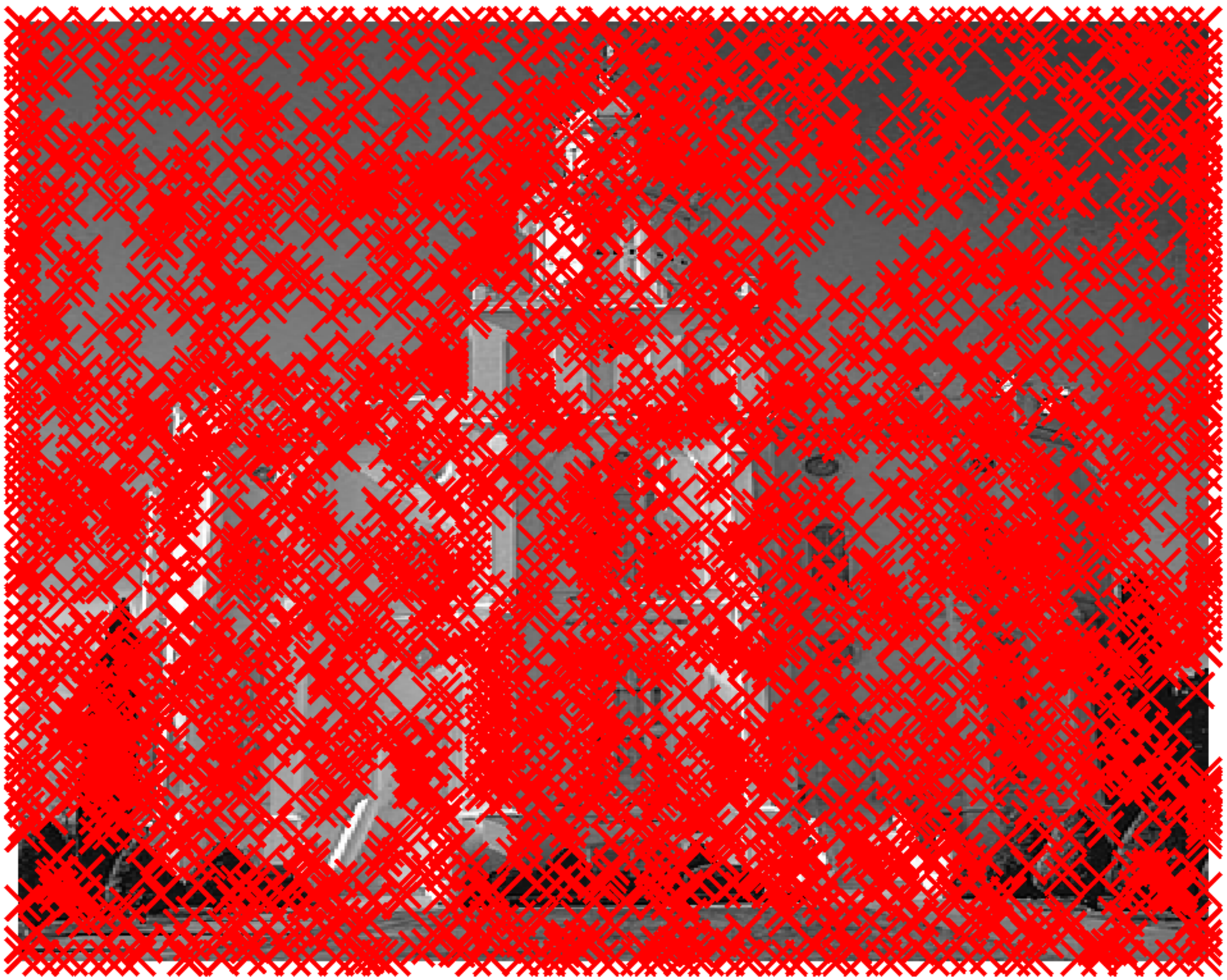
Figure 1: For each octave of scale space, the initial image is repeatedly convolved with Gaussians to produce the set of scale space images shown on the left. Adjacent Gaussian images are subtracted to produce the difference-of-Gaussian images on the right. After each octave, the Gaussian image is down-sampled by a factor of 2, and the process repeated.

Scale space extrema



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

SIFT Keypoints

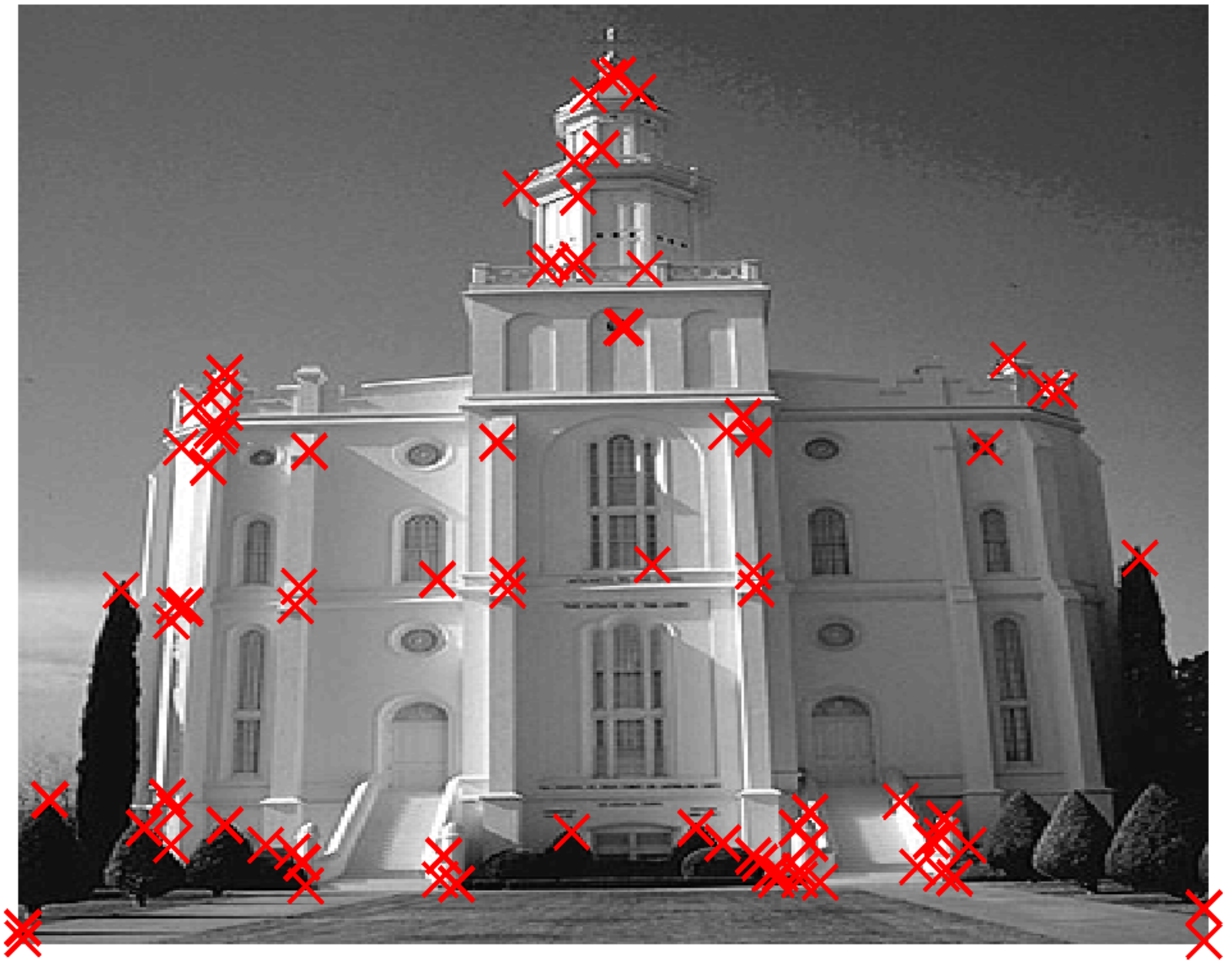


Threshold local variance

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

Threshold Harris cornerness

Lowe 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)
  - SUSAN
  - FAST

- Coming up: Description!