Features

Digital Visual Effects
Yung-Yu Chuang

Outline
- Features
- Harris corner detector
- SIFT
- Extensions
- Applications

Features

- Also known as interesting points, salient points or keypoints. Points that you can easily point out their correspondences in multiple images using only local information.
Desired properties for features

- Distinctive: a single feature can be correctly matched with high probability.
- Invariant: invariant to scale, rotation, affine, illumination and noise for robust matching across a substantial range of affine distortion, viewpoint change and so on. That is, it is repeatable.

Components

- **Feature detection** locates where they are
- **Feature description** describes what they are
- **Feature matching** decides whether two are the same one

Applications

- Object or scene recognition
- Structure from motion
- Stereo
- Motion tracking
- ...

Harris corner detector
Moravec corner detector (1980)

- We should easily recognize the point by looking through a small window
- Shifting a window in any direction should give a large change in intensity
Moravec corner detector

Change of intensity for the shift \([u,v]\):

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

Four shifts: \((u,v) = (1,0), (1,1), (0,1), (-1,1)\)

Look for local maxima in \(\min\{E\}\)

Problems of Moravec detector

- Noisy response due to a binary window function
- Only a set of shifts at every 45 degree is considered
- Only minimum of \(E\) is taken into account

\(\Rightarrow\) Harris corner detector (1988) solves these problems.

Harris corner detector

Noisy response due to a binary window function

- Use a Gaussian function

\[
w(x, y) = \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)
\]

Window function \(w(x,y)\) = Gaussian
Harris corner detector

Only a set of shifts at every 45 degree is considered
➢ Consider all small shifts by Taylor’s expansion

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

\[
E(u, v) = Au^2 + 2Cu v + Bv^2
\]

\[
A = \sum_{x,y} w(x, y)I_x^2(x, y)
\]

\[
B = \sum_{x,y} w(x, y)I_y^2(x, y)
\]

\[
C = \sum_{x,y} w(x, y)I_x(x, y)I_y(x, y)
\]

Harris corner detector (matrix form)

Equivalently, for small shifts \([u,v]\) we have a bilinear approximation:

\[
E(u, v) \approx [u\ v] M [\begin{array}{c}u \\ v \end{array}]
\]

, where \(M\) is a 2×2 matrix computed from image derivatives:

\[
M = \sum_{x,y} w(x, y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}
\]

\[
E(u) = \sum_{x_0 \in W(p)} w(x_0) \left| I(x_0 + u) - I(x_0) \right|^2
\]

\[
= \left( I_0 + \frac{\partial I}{\partial x} u \right)^2 - I_0
\]

\[
= \left( \frac{\partial I}{\partial x} \right)^2 u^2
\]

\[
= u^T \frac{\partial I}{\partial x} \frac{\partial I}{\partial x}^T u
\]

\[
= u^T M u
\]
Harris corner detector

Only minimum of $E$ is taken into account

- A new corner measurement by investigating the shape of the error function

$u^T M u$ represents a quadratic function; Thus, we can analyze $E$’s shape by looking at the property of $M$.

High-level idea: what shape of the error function will we prefer for features?

Quadratic forms

- Quadratic form (homogeneous polynomial of degree two) of $n$ variables $x_i$

\[
\sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_i x_j
\]

- Examples

\[
4x_1^2 + 5x_2^2 + 3x_3^2 + 2x_1x_2 + 4x_1x_3 + 6x_2x_3
\]

\[
\begin{pmatrix}
x_1 & x_2 & x_3
\end{pmatrix}
\begin{pmatrix}
4 & 1 & 2 \\
1 & 5 & 3 \\
2 & 3 & 3
\end{pmatrix}
\begin{pmatrix}
x_1 \\
x_2 \\
x_3
\end{pmatrix}
\]

Symmetric matrices

- Quadratic forms can be represented by a real symmetric matrix $A$ where

\[
a_{ij} = \begin{cases} 
  c_{ij} & \text{if } i = j, \\
  \frac{1}{2}c_{ij} & \text{if } i < j, \\
  \frac{1}{2}c_{ji} & \text{if } i > j.
\end{cases}
\]

\[
\sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_i x_j = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} x_i x_j
\]

\[
\begin{pmatrix}
x_1 & \cdots & x_n
\end{pmatrix}
\begin{pmatrix}
a_{11} & \cdots & a_{1n} \\
\vdots & \ddots & \vdots \\
a_{n1} & \cdots & a_{nn}
\end{pmatrix}
\begin{pmatrix}
x_1 \\
\vdots \\
x_n
\end{pmatrix}
\]

\[
x^T A x
\]
**Eigenvalues of symmetric matrices**

Suppose $A \in \mathbb{R}^{n \times n}$ is symmetric, i.e., $A = A^T$.

**Fact:** the eigenvalues of $A$ are real.

Suppose $Av = \lambda v$, $v \neq 0$, $v \in \mathbb{C}^n$.

$$v^T A v = v^T (Av) = \lambda v^T v = \lambda \sum_{i=1}^{n} |v_i|^2$$

$$v^T A v = (Av)^T v = (\lambda v)^T v = \lambda \sum_{i=1}^{n} |v_i|^2$$

We have $\lambda = \bar{\lambda}$, i.e., $\lambda \in \mathbb{R}$

(hence, can assume $v \in \mathbb{R}^n$)

Brad Osgood

---

**Eigenvectors of symmetric matrices**

Suppose $A \in \mathbb{R}^{n \times n}$ is symmetric, i.e., $A = A^T$.

**Fact:** there is a set of orthonormal eigenvectors of $A$

$$A = Q\Lambda Q^T$$

---

**Harris corner detector**

Intensity change in shifting window: eigenvalue analysis

$$E(u, v) \equiv [u, v] M \begin{bmatrix} u \\ v \end{bmatrix}$$

$\lambda_1, \lambda_2$ – eigenvalues of $M$

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

- Direction of the fastest change
- Direction of the slowest change

$$z^T z = 1$$

Brad Osgood
Visualize quadratic functions

\[ A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \]
Harris corner detector

Classification of image points using eigenvalues of $M$:

- $\lambda_1$ and $\lambda_2$ are large; $\lambda_1 \sim \lambda_2$; $E$ increases in all directions.
- $\lambda_1$ and $\lambda_2$ are small; $E$ is almost constant in all directions.

Measure of corner response:

$$R = \det M - k (\text{trace} M)^2$$

($k$ - empirical constant, $k = 0.04-0.06$)

Another view
Summary of Harris detector

1. Compute x and y derivatives of image

\[ I_x = G^x \ast I \quad I_y = G^y \ast I \]

2. Compute products of derivatives at every pixel

\[ I_{x^2} = I_x \cdot I_x \quad I_{y^2} = I_y \cdot I_y \quad I_{xy} = I_x \cdot I_y \]

3. Compute the sums of the products of derivatives at each pixel

\[ S_{x^2} = G^2 \ast I_{x^2} \quad S_{y^2} = G^2 \ast I_{y^2} \quad S_{xy} = G^2 \ast I_{xy} \]

4. Define the matrix at each pixel

\[ M(x, y) = \begin{bmatrix} S_{x^2}(x, y) & S_{xy}(x, y) \\ S_{xy}(x, y) & S_{y^2}(x, y) \end{bmatrix} \]

5. Compute the response of the detector at each pixel

\[ R = \det M - k(\text{trace}(M))^2 \]

6. Threshold on value of R; compute nonmax suppression.
Harris corner detector (input)

Corner response $R$

Threshold on $R$

Local maximum of $R$
Harris corner detector

Harris detector: summary

- Average intensity change in direction $[u,v]$ can be expressed as a bilinear form:

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

- Describe a point in terms of eigenvalues of $M$:

$$R = \lambda_1 \lambda_2 - k(\lambda_1 + \lambda_2)^2$$

- A good (corner) point should have a large intensity change in all directions, i.e. $R$ should be large positive

Harris detector: some properties

- Partial invariance to affine intensity change
  - Only derivatives are used $\Rightarrow$ invariance to intensity shift $I \rightarrow I + b$
  - Intensity scale: $I \rightarrow aI$

Now we know where features are

- But, how to match them?
- What is the descriptor for a feature? The simplest solution is the intensities of its spatial neighbors. This might not be robust to brightness change or small shift/rotation.
Harris Detector: Some Properties

- Rotation invariance

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

*Corner response* $R$ is invariant to image rotation

Harris Detector is rotation invariant

Repeatability rate:
\[
\frac{\# \text{ correspondences}}{\# \text{ possible correspondences}}
\]

Harris detector: some properties

- Quality of Harris detector for different scale changes

Repeatability rate:
\[
\frac{\# \text{ correspondences}}{\# \text{ possible correspondences}}
\]
Scale invariant detection

- Consider regions (e.g. circles) of different sizes around a point
- Regions of corresponding sizes will look the same in both images

SIFT

(Scale Invariant Feature Transform)

Scale invariant detection

- The problem: how do we choose corresponding circles independently in each image?
- Aperture problem

SIFT

- SIFT is an carefully designed procedure with empirically determined parameters for the invariant and distinctive features.
SIFT stages:

- Scale-space extrema detection
- Keypoint localization
- Orientation assignment
- Keypoint descriptor

A 500x500 image gives about 2000 features

1. Detection of scale-space extrema

- For scale invariance, search for stable features across all possible scales using a continuous function of scale, scale space.
- SIFT uses DoG filter for scale space because it is efficient and as stable as scale-normalized Laplacian of Gaussian.

DoG filtering

Convolution with a variable-scale Gaussian

\[ L(x, y, \sigma) = G(x, y, \sigma) \ast I(x, y), \]
\[ G(x, y, \sigma) = \frac{1}{(2\pi \sigma^2)} \exp\left(-\frac{x^2+y^2}{\sigma^2}\right) \]

Difference-of-Gaussian (DoG) filter

\[ G'(x, y, k\sigma) - G(x, y, \sigma) \]

Convolution with the DoG filter

\[ D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) \ast I(x, y) \]
\[ = L(x, y, k\sigma) - L(x, y, \sigma). \]

Scale space

\( \sigma \) doubles for the next octave

Dividing into octave is for efficiency only.
Detection of scale-space extrema

Keypoint localization

- $X$ is selected if it is larger or smaller than all 26 neighbors

Decide scale sampling frequency

- It is impossible to sample the whole space, tradeoff efficiency with completeness.

- Decide the best sampling frequency by experimenting on 32 real image subject to synthetic transformations. (rotation, scaling, affine stretch, brightness and contrast change, adding noise...)

Decide scale sampling frequency

- Graph showing the number of keypoints per image against the number of scales sampled per octave.
Decide scale sampling frequency

For detector, repeatability
For descriptor, distinctiveness

$s=3$ is the best, for larger $s$, too many unstable features

Pre-smoothing

$\sigma = 1.6$, plus a double expansion

Scale invariance

2. Accurate keypoint localization

- Reject points with low contrast (flat) and poorly localized along an edge (edge)
- Fit a 3D quadratic function for sub-pixel maxima
2. Accurate keypoint localization

- Reject points with low contrast (flat) and poorly localized along an edge (edge)
- Fit a 3D quadratic function for sub-pixel maxima

\[ f(x) \approx f(0) + f'(0)x + \frac{f''(0)}{2} x^2 \]
\[ f(x) \approx 6 + 2x + \frac{-6}{2} x^2 = 6 + 2x - 3x^2 \]

\[ f'(x) = 2 - 6x = 0 \rightarrow \hat{x} = \frac{1}{3} \]
\[ f(\hat{x}) = 6 + 2 \cdot \frac{1}{3} - 3 \left( \frac{1}{3} \right)^2 = 6 \frac{1}{3} \]

Accurate keypoint localization

- Taylor expansion in a matrix form, \( x \) is a vector, \( f \) maps \( x \) to a scalar
\[
 f(x) - f + \frac{\partial f^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 f}{\partial x^2} x
\]

Hessian matrix (often symmetric)

\[
 \left( \begin{array}{c}
 \frac{\partial f}{\partial x}
 \\
 \frac{\partial f}{\partial x_1}
 \\
 \vdots
 \\
 \frac{\partial f}{\partial x_n}
 \end{array} \right)
\]

\[
 \begin{pmatrix}
 \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 x_n} \\
 \frac{\partial^2 f}{\partial x_2 x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 x_n} \\
 \vdots & \vdots & \ddots & \vdots \\
 \frac{\partial^2 f}{\partial x_n x_1} & \frac{\partial^2 f}{\partial x_n x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2}
 \end{pmatrix}
\]

2D illustration

\[ f(x) = f + \frac{\partial f^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 f}{\partial x^2} x \]

\[
\begin{array}{c|c|c|c}
 f_{-1,1} & f_{0,1} & f_{1,1} \\
 \hline
 f_{-1,0} & f_{0,0} & f_{1,0} \\
 f_{-1,-1} & f_{0,-1} & f_{1,-1} \\
\end{array}
\]

\[
\begin{align*}
 \frac{\partial f}{\partial x} &= (f_{1,0} - f_{-1,0})/2 \\
 \frac{\partial f}{\partial y} &= (f_{0,1} - f_{0,-1})/2 \\
 \frac{\partial^2 f}{\partial x^2} &= f_{1,0} - 2f_{0,0} + f_{-1,0} \\
 \frac{\partial^2 f}{\partial y^2} &= f_{0,1} - 2f_{0,-1} + f_{-1,-1} \\
 \frac{\partial^2 f}{\partial x \partial y} &= (f_{-1,1} - f_{-1,-1} - f_{1,-1} + f_{1,1})/4
\end{align*}
\]
2D example

\[ f(x) = f + \frac{\partial f^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 f}{\partial x^2} x \]

<table>
<thead>
<tr>
<th></th>
<th>-17</th>
<th>-1</th>
<th>-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>-9</td>
<td>7</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>-9</td>
<td>7</td>
<td>7</td>
<td></td>
</tr>
</tbody>
</table>

Derivation of matrix form

\[ f(x) = f + \frac{\partial f^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 f}{\partial x^2} x \]

\[ h(x) = g^T x \]

\[ \frac{\partial h}{\partial x} = \]

\[ = \sum_{i=1}^{n} g_i x_i \]
Derivation of matrix form

\[ f(x) = f + \frac{\partial f}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 f}{\partial x^2} x \]

\[ h(x) = x^T A x = (x_1, \ldots, x_n) \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \sum_{i=1}^n \sum_{j=1}^n a_{ij} x_i x_j \]

\[ \frac{\partial h}{\partial x} = \begin{pmatrix} \frac{\partial h}{\partial x_1} \\ \vdots \\ \frac{\partial h}{\partial x_n} \end{pmatrix} = \begin{pmatrix} \frac{\sum_{j=1}^n a_{1j} x_j + \sum_{j=1}^n a_{1j} x_j}{x_1} \\ \vdots \\ \frac{\sum_{j=1}^n a_{nj} x_j + \sum_{j=1}^n a_{nj} x_j}{x_n} \end{pmatrix} = (A^T + A)x \]

Accurate keypoint localization

\[ f(x) = f + \frac{\partial f}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 f}{\partial x^2} x \]

- \( x \) is a 3-vector
- Change sample point if offset is larger than 0.5
- Throw out low contrast (<0.03)

\[ D(\hat{x}) = D + \frac{\partial D}{\partial \hat{x}} \hat{x} + \frac{1}{2} \hat{x}^T \frac{\partial^2 D}{\partial \hat{x}^2} \hat{x} \]

\[ = D + \frac{\partial D}{\partial \hat{x}} \hat{x} + \frac{1}{2} \left( \frac{\partial^2 D}{\partial \hat{x}^2} \hat{x} \right) \frac{\partial^2 D}{\partial \hat{x}^2} \hat{x} \]

\[ = D + \frac{\partial D}{\partial \hat{x}} \hat{x} + \frac{1}{2} \left( \frac{\partial^2 D}{\partial \hat{x}^2} \hat{x} \right) \frac{\partial^2 D}{\partial \hat{x}^2} \hat{x} \]

\[ = D + \frac{\partial D}{\partial \hat{x}} \hat{x} + \frac{1}{2} \left( \frac{\partial^2 D}{\partial \hat{x}^2} \hat{x} \right) \frac{\partial^2 D}{\partial \hat{x}^2} \hat{x} \]

\[ = D + \frac{1}{2} \frac{\partial D}{\partial \hat{x}} \hat{x} \]
Eliminating edge responses

\[
\mathbf{H} = \begin{bmatrix}
    D_{xx} & D_{xy} \\
    D_{xy} & D_{yy}
\end{bmatrix}
\]

Hessian matrix at keypoint location

\[
\text{Tr}(\mathbf{H}) = D_{xx} - D_{yy} = \alpha + \beta,
\]

\[
\text{Det}(\mathbf{H}) = D_{xx}D_{yy} - (D_{xy})^2 = \alpha \beta.
\]

Let \(\alpha = r \beta\)

\[
\frac{\text{Tr}(\mathbf{H})^2}{\text{Det}(\mathbf{H})} = \frac{(r \beta + \beta)^2}{r \beta^2} = \frac{(r + 1)^2}{r}
\]

Keep the points with \(\frac{\text{Tr}(\mathbf{H})^2}{\text{Det}(\mathbf{H})} < \frac{(r + 1)^2}{r}, \quad r=10\)

Maxima in D

Keypoint detector

832 extrema

729 after contrast filtering

536 after curvature filtering
3. Orientation assignment

- By assigning a consistent orientation, the keypoint descriptor can be orientation invariant.
- For a keypoint, \( L \) is the Gaussian-smoothed image with the closest scale,
  
  \[
  m(x, y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2} \\
  \theta(x, y) = \tan^{-1}((L(x,y+1) - L(x,y-1))/(L(x+1,y) - L(x-1,y)))
  \]

Orientation assignment

- Keypoint location = extrema location
- Keypoint scale is scale of the DOG image
Orientation assignment

gradient magnitude
weighted by 2D gaussian kernel
$\sigma = 1.5 \times$ scale of the keypoint

Orientation assignment

weighted gradient magnitude

accurate peak position is determined by fitting

Orientation assignment

weighted orientation histogram.

36 buckets
10 degree range of angles in each bucket, i.e.
$0 \leq \theta < 10$ : bucket 1
$10 \leq \theta < 20$ : bucket 2
$20 \leq \theta < 30$ : bucket 3

Orientation assignment

There may be multiple orientations.

accurate peak position is determined by fitting

In this case, generate duplicate keypoints, one with orientation at 25 degrees, one at 135 degrees.

Design decision: you may want to limit number of possible multiple peaks to two.
Orientation assignment

36-bin orientation histogram over 360°, weighted by m and 1.5*scale falloff
Peak is the orientation
Local peak within 80% creates multiple orientations
About 15% has multiple orientations and they contribute a lot to stability

SIFT descriptor

4. Local image descriptor

- Thresholded image gradients are sampled over 16x16 array of locations in scale space
- Create array of orientation histograms (w.r.t. key orientation)
- 8 orientations x 4x4 histogram array = 128 dimensions
- Normalized, clip values larger than 0.2, renormalize

Why 4x4x8?

Correct nearest descriptor (%) vs Width n of descriptor (angle 50 deg, noise 1%)

With 16 orientations
With 8 orientations
With 4 orientations
Sensitivity to affine change

- Feature matching
  - for a feature $x$, he found the closest feature $x_1$ and the second closest feature $x_2$. If the distance ratio of $d(x, x_1)$ and $d(x, x_2)$ is smaller than 0.8, then it is accepted as a match.

SIFT flow

Maxima in D
Estimated rotation

- Computed affine transformation from rotated image to original image:
  0.7060  -0.7052  128.4230
  0.7057   0.7100 -128.9491
  0 0 1.0000

- Actual transformation from rotated image to original image:
  0.7071  -0.7071  128.6934
  0.7071   0.7071 -128.6934
  0 0 1.0000

SIFT extensions

PCA

Average face

Top ten eigenfaces (left = highest eigenvalue, right = lowest eigenvalue)

PCA-SIFT

- Only change step 4
- Pre-compute an eigen-space for local gradient patches of size 41x41
- 2x39x39=3042 elements
- Only keep 20 components
- A more compact descriptor
**GLOH (Gradient location-orientation histogram)**

- 17 location bins
- 16 orientation bins
- Analyze the $17 \times 16 = 272$-d eigen-space, keep 128 components
- SIFT is still considered the best.

**Multi-Scale Oriented Patches**

- Simpler than SIFT. Designed for image matching. [Brown, Szeliski, Winder, CVPR’2005]
- Feature detector
  - Multi-scale Harris corners
  - Orientation from blurred gradient
  - Geometrically invariant to rotation
- Feature descriptor
  - Bias/gain normalized sampling of local patch (8x8)
  - Photometrically invariant to affine changes in intensity

**Multi-Scale Harris corner detector**

- Image stitching is mostly concerned with matching images that have the same scale, so sub-octave pyramid might not be necessary.

\[
P_0(x, y) = I(x, y) \\
P'_l(x, y) = P_l(x, y) \ast g_P(x, y) \\
P_{l+1}(x, y) = P'_l(sx, sy) \\
S = 2 \quad \sigma_P = 1.0
\]

**Multi-Scale Harris corner detector**

\[
H_l(x, y) = \nabla_{\sigma_d} P_l(x, y) \nabla_{\sigma_d} P_l(x, y)^T \ast g_{\sigma_l}(x, y)
\]

\[
\nabla_{\sigma} f(x, y) \triangleq \nabla f(x, y) \ast g_{\sigma}(x, y)
\]

- Smoother version of gradients
- $\sigma_l = 1.5 \quad \sigma_d = 1.0$

Corner detection function:

\[
f_{HM}(x, y) = \frac{\text{det} \ H_l(x, y)}{\text{tr} \ H_l(x, y)} = \frac{\lambda_1 \lambda_2}{\lambda_1 + \lambda_2}
\]

- Pick local maxima of 3x3 and larger than 10
Keypoint detection function

- **Harris**
  \[ f_H = \lambda_1 \lambda_2 - 0.04(\lambda_1 + \lambda_2)^2 = \det H - 0.04(\text{tr } H)^2 \]

- **Harmonic mean**
  \[ f_{HM} = \frac{\lambda_1 \lambda_2}{(\lambda_1 + \lambda_2)} = \frac{\det H}{\text{tr } H} \]

- **Shi-Tomasi**
  \[ f_{ST} = \min(\lambda_1, \lambda_2) \]

Experiments show roughly the same performance.

Non-maximal suppression

- Restrict the maximal number of interest points, but also want them spatially well distributed.
- Only retain maximums in a neighborhood of radius \( r \).
- Sort them by strength, decreasing \( r \) from infinity until the number of keypoints (500) is satisfied.

Non-maximal suppression

(a) Strongest 250
(b) Strongest 500
(c) ANMS 250, \( r = 24 \)
(d) ANMS 500, \( r = 16 \)

Sub-pixel refinement

\[
f(x) = f + \frac{\partial f}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 f}{\partial x^2} x
\]

\[
x_m = - \frac{\partial^2 f}{\partial x^2}^{-1} \frac{\partial f}{\partial x}
\]

\[
\begin{array}{|c|c|c|c|}
\hline
f_{-1,1} & f_{0,1} & f_{1,1} \\
\hline
f_{-1,0} & f_{0,0} & f_{1,0} \\
\hline
f_{-1,-1} & f_{0,-1} & f_{1,-1} \\
\hline
\end{array}
\]

\[
\begin{align*}
\frac{\partial f}{\partial x} &= (f_{1,0} - f_{-1,0})/2 \\
\frac{\partial f}{\partial y} &= (f_{0,1} - f_{0,-1})/2 \\
\frac{\partial^2 f}{\partial x^2} &= f_{1,0} - 2f_{0,0} + f_{-1,0} \\
\frac{\partial^2 f}{\partial y^2} &= f_{0,1} - 2f_{0,0} + f_{0,-1} \\
\frac{\partial^2 f}{\partial x \partial y} &= (f_{-1,-1} - f_{-1,1} - f_{1,-1} + f_{1,1})/4 \\
\end{align*}
\]
Orientation assignment

- Orientation = blurred gradient

\[ u_l(x, y) = \nabla_{\sigma_o} P_l(x, y) \]

\[ \sigma_o = 4.5 \]

\[ [\cos \theta, \sin \theta] = u/|u| \]

---

Descriptor Vector

- Rotation Invariant Frame
  - Scale-space position \((x, y, s) + \text{orientation } (\theta)\)

---

MSOP descriptor vector

- 8x8 oriented patch sampled at 5 x scale. See TR for details.
- Sampled from \(P_l(x, y) \ast g_{2 \times \sigma_p}(x, y)\) with spacing=5

---

MSOP descriptor vector

- 8x8 oriented patch sampled at 5 x scale. See TR for details.
- Bias/gain normalisation: \(I' = (I - \mu)/\sigma\)
- Wavelet transform
Detections at multiple scales

Summary
- Multi-scale Harris corner detector
- Sub-pixel refinement
- Orientation assignment by gradients
- Blurred intensity patch as descriptor

Feature matching
- Exhaustive search
  - for each feature in one image, look at all the other features in the other image(s)
- Hashing
  - compute a short descriptor from each feature vector, or hash longer descriptors (randomly)
- Nearest neighbor techniques
  - $k$-trees and their variants (Best Bin First)

Wavelet-based hashing
- Compute a short (3-vector) descriptor from an 8x8 patch using a Haar “wavelet”
- Quantize each value into 10 (overlapping) bins ($10^3$ total entries)
- [Brown, Szeliski, Winder, CVPR’2005]
Nearest neighbor techniques

- **k-D tree**
- **Best Bin First (BBF)**

Applications

Recognition

3D object recognition
3D object recognition

Office of the past

Image retrieval

Image retrieval

- Video of desk
- Images from PDF
- Internal representation
- Scene Graph
- Desk T
- Desk T+1

- Change in viewing angle
- > 5000 images
- 22 correct matches
Image retrieval

Robot location

Robotics: Sony Aibo

Structure from Motion

• The SFM Problem
  - Reconstruct scene geometry and camera motion from two or more images

SIFT is used for
- Recognizing charging station
- Communicating with visual cards
- Teaching object recognition
- Soccer

The SFM Problem
- Reconstruct scene geometry and camera motion from two or more images

Track 2D Features
Estimate 3D
Optimize (Bundle Adjust)
Fit Surfaces

SFM Pipeline
Structure from Motion

Augmented reality

Automatic image stitching
Automatic image stitching

Reference

- *SIFT Keypoint Detector*, David Lowe.
- *Matlab SIFT Tutorial*, University of Toronto.