Speeded-Up Robust Features

Dr. V Masilamani
masila@iiitdm.ac.in

Department of Computer Science and Engineering
IIITDM Kancheepuram
Chennai-127

Overview

Outline of SURF

Steps Involved:

- Find Key points
  - Define key point using Hessian matrix
  - Approximate Hessian matrix using box filter
  - Do Non Maxima suppression
  - Find true location of key points using quadratic interpolation
- Find descriptor for each key point
  - Use Haar wavelet response of each sub window of the window around the key point

To find approximation of Hessian matrix and response of Wavelet

- Compute integral image once
- Use integral images

Integral Image

- The integral image $II(x,y)$ of an image $I(x',y')$ represents the sum of all pixels in $I(x',y')$ of a rectangular region formed by $(0,0)$ and (x,y)

$$II(x,y) = \sum_{x' \leq x, y' \leq y} I(x',y')$$

1	5
2	4

1	6
3	12

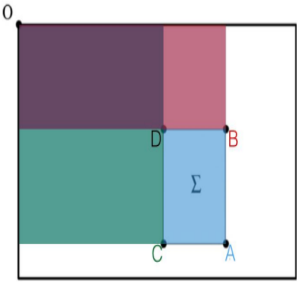
Figure 1: Image and its Integral Image

- The time complexity of finding Integral Image II of an Image I of size $n \times n$ is $O(n^4)$ $(2 + 3 + 4 + ... + n^2$ additions)

Integral Image (cont.)

- Using dynamic programming, II can be computed in $O(n^2)$
- Using II of image I, the sum of pixels over any rectangular region of any size can be computed in constant time (Without II, $O(n^2)$)

$$\sum = A - B - C + D$$



Integral Image (cont.)

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

1	3	5	9	10
4	10	13	22	25
6	15	21	32	39
10	20	31	46	59
16	29	42	58	74

- Given a point $X = (x, y)$ in an image I, the Hessian matrix $H(X, \sigma)$ in X at scale σ is defined as follows

$$H(X, \sigma) = \begin{bmatrix} G_{xx}(X, \sigma) & G_{xy}(X, \sigma) \\ G_{xy}(X, \sigma) & G_{yy}(X, \sigma) \end{bmatrix}$$

where $G_{xx}(X, \sigma)$ is the convolution of the Gaussian second order derivative $\frac{\partial^2}{\partial x^2} g(\sigma)$ with the image I in point X.

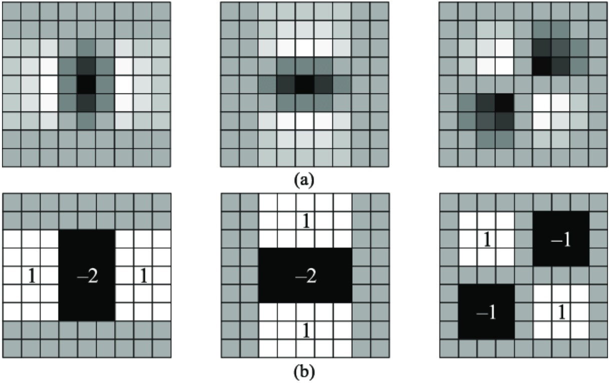
The locations where the determinant of the Hessian is maximum are called **interest points**

Hessian Approximation

- The actual computation of the Hessian matrix is expensive
- The Hessian can be approximated using filters called **Box Filters**
- **Box Filter** is a filter that consists of rectangular regions with constant value in each region
  - Average filters, Sum filters are Box Filters

Hessian Approximation (cont.)

Box filters(bottom row) that approximate second order partial derivatives of Gaussian filters in X, Y and XY directions



- Instead of exact Hessian, find approximate Hessian using box filters

Hessian Approximation (cont.)

$$H_{apprax}(X,\sigma) = \begin{bmatrix} B_{xx}(X,\sigma) & B_{xy}(X,\sigma) \\ B_{xy}(X,\sigma) & B_{yy}(X,\sigma) \end{bmatrix}$$

where $B_{xx}(X,\sigma)$, $B_{xy}(X,\sigma)$ $B_{yy}(X,\sigma)$ are approximations to G_{xx}, G_{xy}, G_{yy} respectively

$$det(H_{approx}) = B_{xx}B_{yy} - (wB_{xy})^2$$

(w = 0.9)

- det(H_{approx}) will be low for isolated points(noise), high for corner points

Hessian Approximation (cont.)

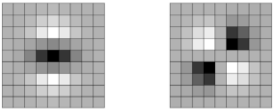


Figure 2: Gaussian second order partial derivative filters in y-direction(G_{xx}) and xy-direction(G_{xy})

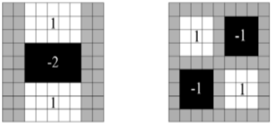


Figure 3: Box filter approximations of the Gaussian second order partial derivative filter in x dir (B_{xx}) and in xy (B_{xy})

Hessian Approximation (cont.)

- Box filter (B_{xx}) is an Approximation of second order derivative of Gaussian filter in x direction (G_{xx})
- Box filter (B_{yy}) is an Approximation of second order derivative of Gaussian filter in y direction (G_{yy})
- Box filter (B_{xy}) is an Approximation of second order derivative of Gaussian in x followed by y direction (G_{xy})
- At each point (x, y) in an image, the response of box filter at (x, y) can be computed in constant time(using Integral Image)

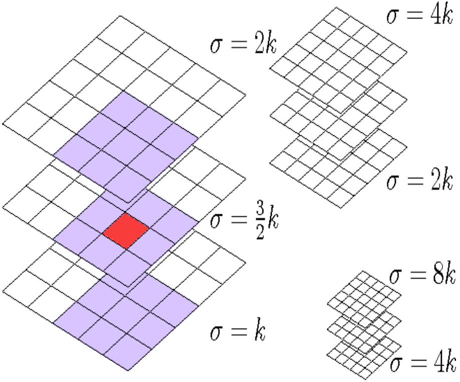
Scale Space Representation

- Scale Space: The input image is convolved with a box filter of various dimensions such as 9x9, 15x15, 27x27 etc.
- Size of the filter corresponds to the standard deviation(σ) of Gaussian.
  - Recall that the filter is second order partial derivative of Gaussian
  - The σ is called as scale
- The scale space is further divided into octaves (sets of filter responses)

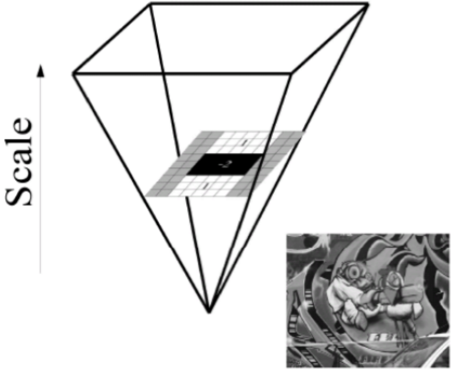
Scale Space Representation (cont.)

Three octaves:

- $\sigma = k, \sigma = 1.5k, \sigma = 3k$
- $\sigma = 4k, \sigma = 6k, \sigma = 8k$
- $\sigma = 2k, \sigma = 3k, \sigma = 4k$

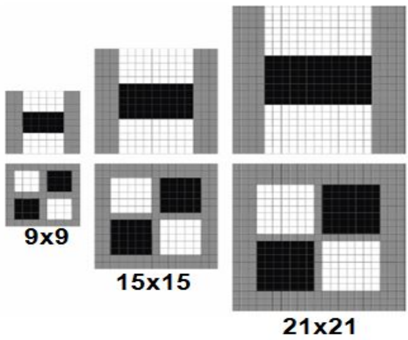


Scale Space Representation (cont.)



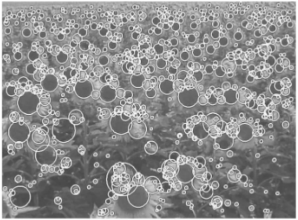
Scale Space Representation (cont.)

Box filters of different scales



Interest Point Localization

- To localize interest points in the image over different scales, a non-maximum suppression in a $3x3x3$ neighborhood is applied
  - Non-maxima points are not interest points
- For each three maxima of the determinant of the Hessian matrix
  - Do quadratic interpolation
  - Call the extrema as key point, dropping those three points (or farthest point from the extrema)

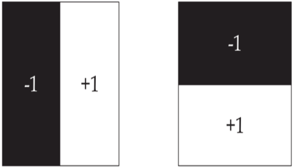


Feature Vector

To find Feature vector for each key point (x, y, s, o)

- Define the axis-orientated square window of size $20s \times 20s$ centered around the interest point
  - Find the orientation of square such that $\sum dx$ is max (To keep rotation invariance)
    - Find the circle, centered at key point, and find the min-bounding rectangle for the circle
    - Find $\sum dx$ for the window
    - Rotate the circle 30 degree, and find $\sum dx$
    - Repeat the pre-step till 360 degree
    - Find the max $\sum dx$, and call the corresponding degree as the orientation of point
- Subdivide the window into 4×4 grid
- Let H_x be Haar wavelet in x dir, and H_y be the Haar wavelet in y dir

Feature Vector (cont.)



- The following four metrics are extracted from each sub window w_i(of size $5s \times 5s$)
  - Sum of all values of dx in w_i, where $dx = Conv(w_i, H_x)$
  - Sum of all values of dy in w_i, where $dy = Conv(w_i, H_y)$
  - Sum of absolute values of all values of dx in w_i, where $dx = Conv(w_i, H_x)$, call the resultant as $\sum |dx|$

Feature Vector (cont.)

- Sum of absolute values of all values of dy in w_i, where $dy = Conv(w_i, H_y)$, call the resultant as $\sum |dy|$

- For each sub window w_i, find the feature vector

$$V_i = \begin{bmatrix} \sum \mathrm{d}x \\ \sum \mathrm{d}y \\ \sum |\mathrm{d}x| \\ \sum |\mathrm{d}y| \end{bmatrix}$$

- The final feature vectors for the key point (x, y):
  - Concatenate all V_is to obtain 64 dimensional vector: FV(x,y)=$(V1, ..., V_{16})$