

# Assignment 3

Rama Lesmana (1313617011)

M. Luthfi Albiansyah (1313617033)

# SIFT

## Scale-Invariant Feature Transform

SIFT adalah algoritma untuk mencari point of interest / feature dari gambar yang bersifat invariant terhadap scale. SIFT terbagi menjadi 4 step utama:

- Scale-space extrema detection

- Keypoint localization

- Orientation assignment

- Keypoint descriptor

# Point of Interest



Figure: Detected Point of Interest

# Scale-space extrema detection

Langkah ini mencari semua kandidat keypoint dari suatu image. Extrema Detection membutuhkan Pyramid of Difference of Gaussian, yaitu selisih dari Gaussian-weighted Image dalam beberapa scale dan size berbeda.

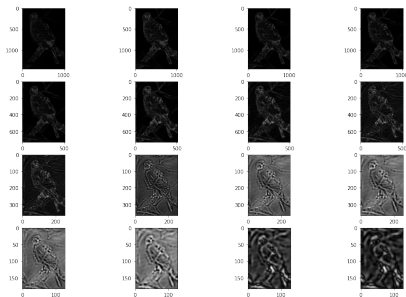


Figure: Pyramid of Difference of Gaussian

# Difference Of Gaussian (DoG)

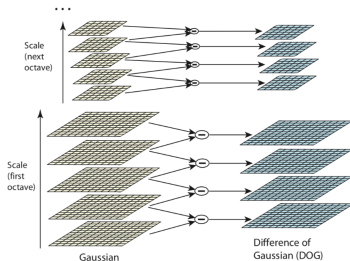
DoG adalah selisih dari dua Gaussian. Difference of Gaussian Filter juga dapat digunakan sebagai alternatif dari turunan kedua Gaussian, terlebih lebih mudah dikalkulasi. Rumus dari Gaussian kernel dan Difference of Gaussian Filter adalah:

$$G_s = \frac{1}{2\pi\sigma^2} e^{\frac{-(x^2+y^2)}{2\sigma^2}} \quad (1)$$

$$DoG_{s1s2} = G_{s1} - G_{s2} \quad (2)$$

# Pyramid Of Difference of Gaussian

Untuk setiap octave, image akan diperkecil menjadi setengah dari ukuran sebelumnya, dan setiap ukuran akan dikonvolusi dengan gaussian dengan skala berbeda. Hasil konvolusi akan diselisihkan dengan hasil konvolusi sesama octave disebelahnya.



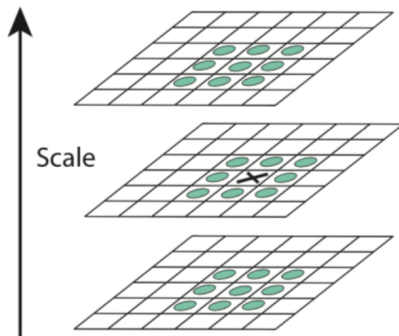
DoG pyramid used for SIFT

# Extrema Detection

Setelah Pyramid of Difference of Gaussian didapatkan, untuk semua titik dalam suatu Image dalam Octave akan dibandingkan dengan 26 tetangga dari titik tersebut, yaitu tetangga dalam image yang sama, dan titik-titik dari Image sebelumnya dalam sesama octave dalam lokasi yang sama.

Bila titik tersebut memiliki value lebih besar dari semua tetangganya, maka dia adalah kandidat maxima, sebaliknya bila value titik lebih kecil dari semua tetangganya, maka dia adalah kandidat minima

# Extrema Detection



Neighborhood for the initial scale-space extrema detection

Figure: Perbandingan titik dengan tetangga sesama octave



# Keypoint Localization

Titik yang ditemukan oleh Extrema Detection belum dapat diandalkan, langkah selanjutnya adalah melakukan filtering dengan menggunakan value contrast dan membuang titik yang berada di dekat edge.

# Keypoint Localization (Jacobian)

Dengan menggunakan Jacobian dan Hessian, nilai contrast dan edge dari setiap candidate akan dihitung. Jacobian dan didefinisikan dengan:

$$D(x_0 + h) \approx D(x_0) + \left( \frac{\partial D}{\partial x} \right)^T \Big|_{x=x_0} h + \frac{1}{2} h^T H(x) h \quad (3)$$

dimana

$$\frac{\partial D}{\partial x} = \begin{pmatrix} \frac{\partial D}{\partial x} \\ \frac{\partial D}{\partial y} \\ \frac{\partial D}{\partial \sigma} \end{pmatrix} = \begin{pmatrix} \frac{D(x+1,y,\sigma) - D(x-1,y,\sigma)}{2} \\ \frac{D(x,y+1,\sigma) - D(x,y-1,\sigma)}{2} \\ \frac{D(x,y,\sigma+1) - D(x,y,\sigma-1)}{2} \end{pmatrix} \quad (4)$$

# Keypoint Localization (Hessian)

dan Hessian didefinisikan dengan:

$$H(x) = \begin{pmatrix} D_{xx} & D_{xy} & D_{x\sigma} \\ D_{yx} & D_{yy} & D_{y\sigma} \\ D_{\sigma x} & D_{\sigma y} & D_{\sigma\sigma} \end{pmatrix} \quad (5)$$

dimana

$$D_{xy} = \frac{\frac{D(x+1,y+1,\sigma) - D(x-1,y+1,\sigma)}{2} - \frac{D(x-1,y+1,\sigma) - D(x-1,y-1,\sigma)}{2}}{2} \quad (6)$$

# Keypoint Localization (Contrast)

Nilai Contrast didefinisikan dengan:

$$D(\hat{x}) = D + \frac{1}{2} \frac{\partial D^T}{\partial x} * \hat{x} \quad (7)$$

Bilai absolut nilai contrast lebih tinggi dari threshold, maka keypoint itu diterima. Default nilai threshold adalah 0.03.

# Keypoint Localization (Edge)

Nilai Edge difilter dengan persamaan:

$$\frac{Tr(H)^2}{Det(H)} < \frac{(r+1)^2}{r} \quad (8)$$

dimana

$$Tr(H) = D_{xx} + D_{yy} \quad (9)$$

$$Det(H) = D_{xx}D_{yy} - D_{xy}^2 \quad (10)$$

$$H(x) = \begin{pmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{pmatrix} \quad (11)$$

# Orientation Assignment

Memberi orientasi untuk setiap keypoint berdasarkan nilai gradient dan Gaussian-Weighted Image dengan skala sigma 1.5 lebih besar dari skala asli. Nilai gradient dan Orientation didapat dari rumus berikut:

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \quad (12)$$

$$\theta(x, y) = \arctan\left(\frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)}\right) \quad (13)$$

# Orientation Assignment

Untuk setiap keypoint histogram berukuran 36 bin dibuat, setiap bin untuk setiap 10 derajat. Dengan weighted voting berdasarkan gradient dan gaussian Image. Nilai histogram tertinggi diambil, dan setiap angle yang memiliki nilai lebih dari  $0.8 * \text{nilai histogram tertinggi}$  dijadikan orientasi dari keypoint tersebut. Sebuah keypoint dapat memiliki 2 atau lebih orientasi.

# Keypoint Descriptor

Keypoint descriptor merupakan identitas dari setiap keypoint. ditahap ini,  $16 \times 16$  windows yang berasal dari tahap sebelumnya akan dipecah menjadi  $4 \times 4$  subregion, dan setiap subregion akan dihitung orientasinya dengan cara yang mirip dengan tahap sebelumnya.

