

Widzenie Komputerowe - Recognition - Feature detection

Wykład 8.

Magdalena Mazur-Milecka

Katedra Inżynierii Biomedycznej, WETI, PG

4 kwietnia 2019

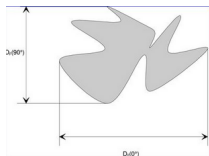
Rodzaje rozpoznania:

- 1 Rozpoznanie (object recognition) - detekcja wcześniej określonego lub nauczonego obiektu lub klasy obiektu
- 2 Identyfikacja - rozpoznanie konkretnej instancji obiektu (konkretna twarz lub odcisk palca, identyfikacja ręcznego pisma)
- 3 Detekcja - przeszukiwanie danych w celu detekcji określonych warunków, np. obiektów odchylonych od normy, nienaturalnych zachowań itp.

Spis treści:

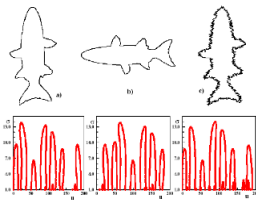
- 1 Detektory, deskryptory
- 2 Ekstrakcja cech - krawędzie
 - filtry gradientowe,
 - LoG,
 - Canny,
- 3 Ekstrakcja cech - punkty
 - detektor Moraveca,
 - detektor Harrisa (Shi-Tomasi),
 - FAST,
 - SIFT+HoG,
- 4 Ekstrakcja cech - linie - Transformacja Hough
- 5 Ekstrakcja cech - bloby
- 6 Rozpoznanie twarzy
 - kaskada Haara,
 - Eigenfaces,
 - Active Appearance Model,

- Detektor - Algorytm wykrywania określonych cech (punktów, krawędzi itp.)
- Deskrytor - Sposób opisu wyznaczonych cech
Przykładowe deskrytory:
 - Histogram - deskrytor koloru,
 - Momenty centralne, współczynniki kształtu (Fereta, Malinowskiej, cyrkularność itp.)



$$W7 = \frac{r_{\min}}{R_{\max}}$$

r_{\min} - minimalna odległość konturu od środka ciężkości obiektu
 R_{\max} - maksymalna odległość konturu od środka ciężkości obiektu



Spis treści:

- 1 Detektory, deskryptory
- 2 **Ekstrakcja cech - krawędzie**
 - filtry gradientowe,
 - LoG,
 - Canny,
- 3 Ekstrakcja cech - punkty
 - detektor Moraveca,
 - detektor Harrisa (Shi-Tomasi),
 - FAST,
 - SIFT+HoG,
- 4 Ekstrakcja cech - linie - Transformacja Hough
- 5 Ekstrakcja cech - bloby
- 6 Rozpoznanie twarzy
 - kaskada Haara,
 - Eigenfaces,
 - Active Appearance Model,

Krawędź - ważna struktura obrazu.

Wynikiem detekcji krawędzi jest obraz binarny, który w miejscu wystąpienia krawędzi ma wartość 1, a w pozostałych 0.

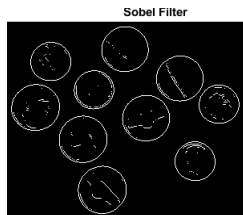


- Operatory gradientowe - maximum pierwszej pochodnej

Sobel

Odnajduje krawędzie przy pomocy dwuwymiarowego splotu macierzy obrazu z odpowiednim jądrem. Pozwala na wykrycie krawędzi w kierunkach co 45° .

$$\begin{vmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{vmatrix}$$
$$\begin{vmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{vmatrix}$$



Algorytmy detekcji krawędzi

- Operatory gradientowe - maximum pierwszej pochodnej

Original



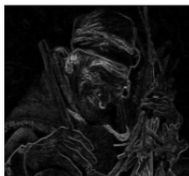
Sobel



Prewitt



Robert



Algorytmy detekcji krawędzi

- Operatory gradientowe - maximum pierwszej pochodnej
- Operatory Laplacian - przejście zero drugiej pochodnej

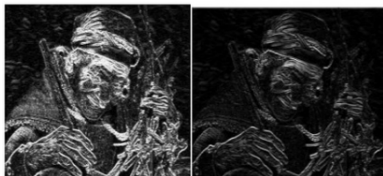
LoG (Laplacian Gausjanów)

Podatny na zakłócenia, dlatego na początku wykonywane jest wygładzenie f. Gaussa, a następnie splot z jądrem. W rezultacie powstają punkty reprezentujące krawędzie i narożniki.

$$\begin{vmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{vmatrix} \quad \begin{vmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{vmatrix}$$
$$\begin{vmatrix} -1 & 2 & -1 \\ 2 & -4 & 2 \\ -1 & 2 & -1 \end{vmatrix}$$

Laplacian

Laplacian of Gaussian



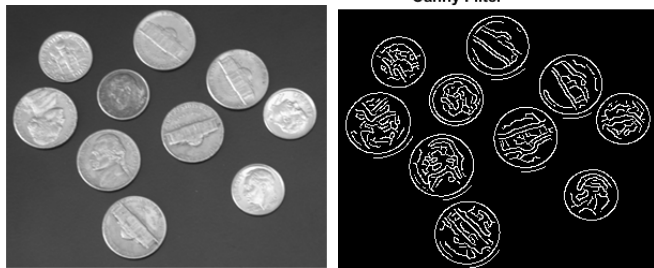
Algorytmy detekcji krawędzi

- Operatory gradientowe - maximum pierwszej pochodnej
- Operatory Laplacian - przejście zero drugiej pochodnej
- Canny

Canny

Algorytm wyszukuje lokalne maxima gradientu obrazu. Używa 2 różnych progów, wykrywa silne i słabsze krawędzie, słabsze krawędzie są rozpatrywane jeżeli łączą się z silniejszymi.

Canny Filter

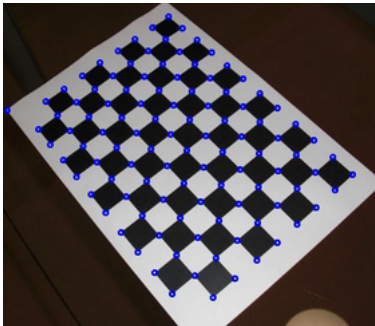


MathWorks

Spis treści:

- 1 Detektory, deskryptory
- 2 Ekstrakcja cech - krawędzie
 - filtry gradientowe,
 - LoG,
 - Canny,
- 3 **Ekstrakcja cech - punkty**
 - detektor Moraveca,
 - detektor Harrisa (Shi-Tomasi),
 - FAST,
 - SIFT+HoG,
- 4 Ekstrakcja cech - linie - Transformacja Hough
- 5 Ekstrakcja cech - bloby
- 6 Rozpoznanie twarzy
 - kaskada Haara,
 - Eigenfaces,
 - Active Appearance Model,

Narożniki - punkty zainteresowania



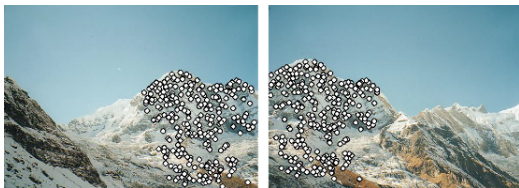
- Wyznaczenie macierzy przekształceń pomiędzy parą obrazów,

Przykładowe zastosowania

- Wyznaczenie macierzy przekształceń pomiędzy parą obrazów,
- Estymacja ruchu,

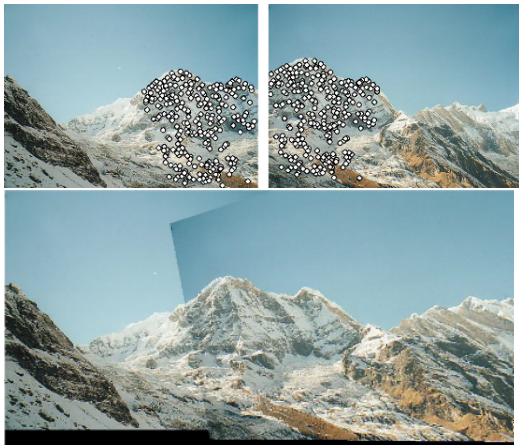
Przykładowe zastosowania

- Wyznaczenie macierzy przekształceń pomiędzy parą obrazów,
- Estymacja ruchu,
- Dopasowanie obrazów, wzorców, obiektów,



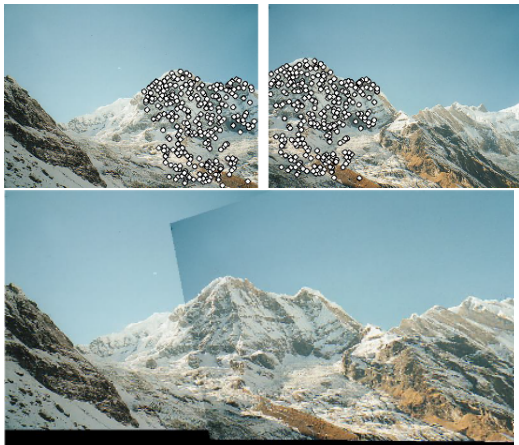
Przykładowe zastosowania

- Wyznaczenie macierzy przekształceń pomiędzy parą obrazów,
- Estymacja ruchu,
- Dopasowanie obrazów, wzorców, obiektów,
- Łączenie obrazów (panorama),

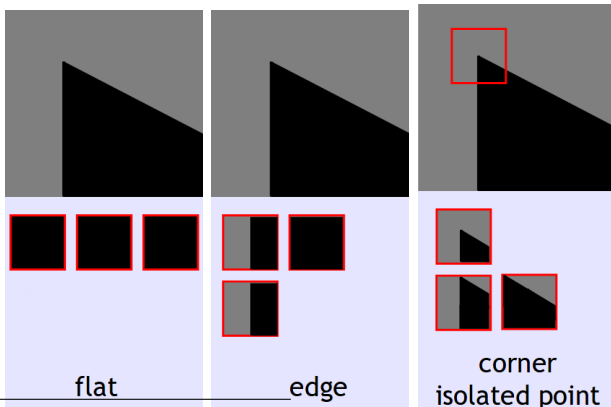


Przykładowe zastosowania

- Wyznaczenie macierzy przekształceń pomiędzy parą obrazów,
- Estymacja ruchu,
- Dopasowanie obrazów, wzorców, obiektów,
- Łączenie obrazów (panorama),
- Robotyka - nawigacja.



$$E(u, v) = \sum_x \sum_y w(x, y) (I(x + u, y + v) - I(x, y))^2$$

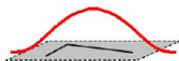


H. P. Moravec: Towards Automatic Visual Obstacle Avoidance. In: Proceedings of the 5th International Joint Conference on Artificial Intelligence. 1977, S. 584

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

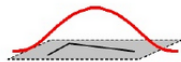
$$w(x, y) = \exp\left(-\frac{(x^2+y^2)}{2\sigma^2}\right)$$

Window function $w(x, y) =$



Gaussian

Window function $w(x, y) =$

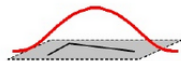


Gaussian

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

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

Window function $w(x, y) =$



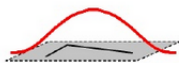
Gaussian

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

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

⇓ po rozwinięciu w szereg Taylora ...

Window function $w(x,y) =$



Gaussian

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

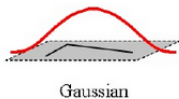
$$E(u, v) = \sum_{x,y} w(x, y) (I(x + u, y + v) - I(x, y))^2$$

↓ po rozwinięciu w szereg Taylora ...

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

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

Window function $w(x,y) =$



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

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

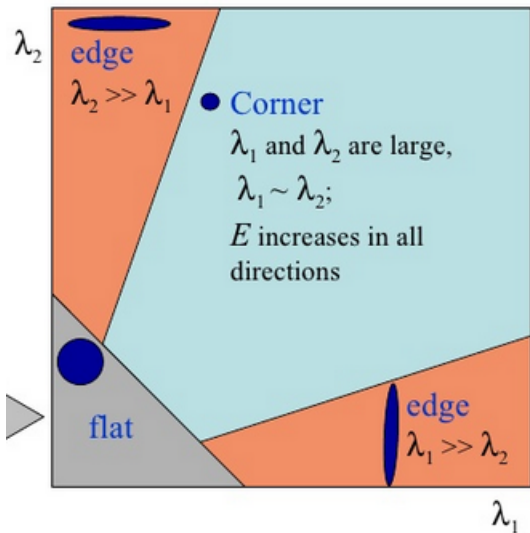
↓ po rozwinięciu w szereg Taylora ...

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

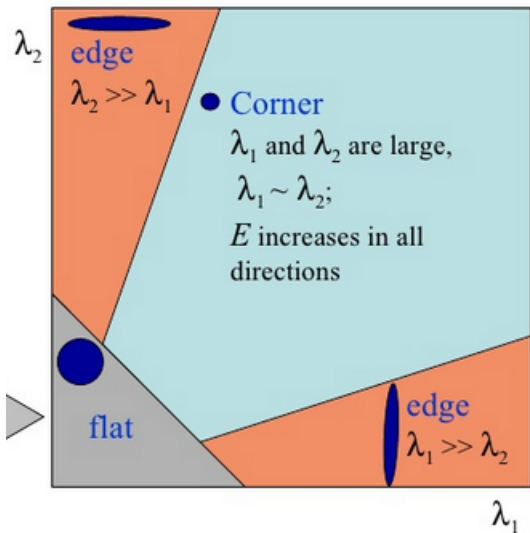
$$M = \sum_{x,y} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

λ_1, λ_2 - wartości własne macierzy M

Detektor Harrisa



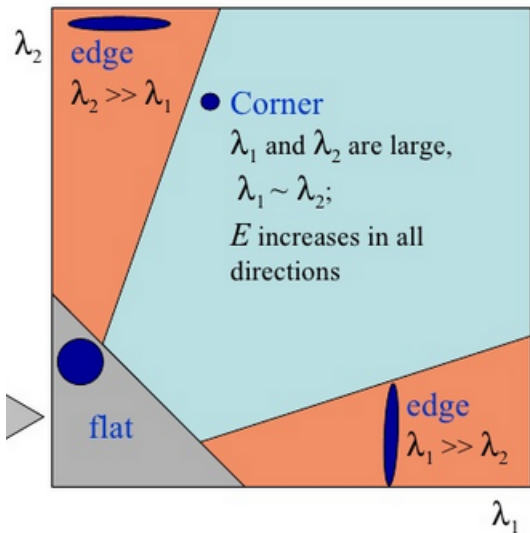
Detektor Harrisa



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

$$\lambda_1 \lambda_2 = \det M$$

$$\lambda_1 + \lambda_2 = \text{trace} M$$



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

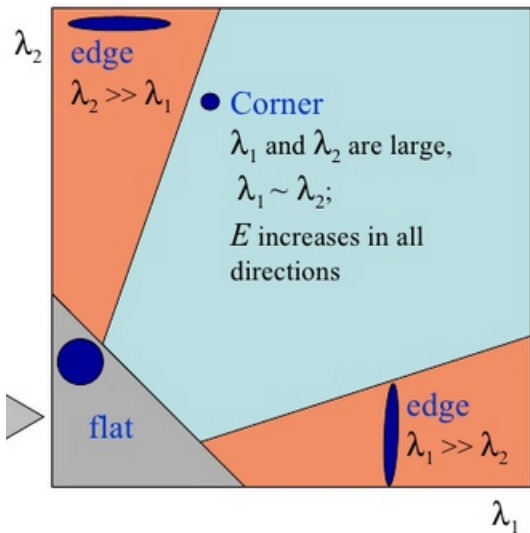
$$\lambda_1 \lambda_2 = \det M$$

$$\lambda_1 + \lambda_2 = \text{trace} M$$

$R > 0$ - narożnik

$R < 0$ = krawędź

$R \simeq 0$ - pow. jednorodna



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

$$\lambda_1 \lambda_2 = \text{det}M$$

$$\lambda_1 + \lambda_2 = \text{trace}M$$

$R > 0$ - narożnik

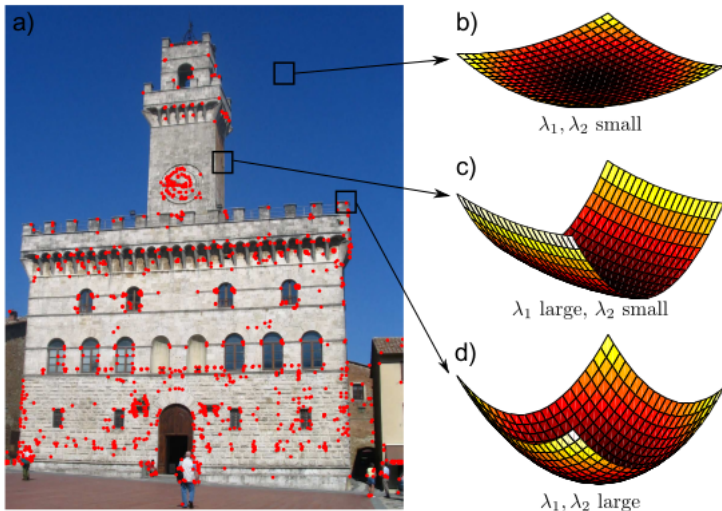
$R < 0$ = krawędź

$R \simeq 0$ - pow. jednorodna

Detektor Shi-Tomasi:

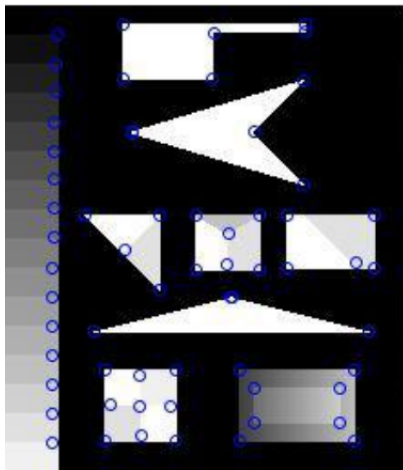
$$R = \min(\lambda_1, \lambda_2)$$

Detektor Harrisa

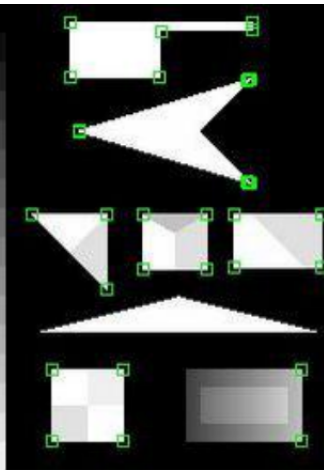


Simon J.D. Prince, Computer vision: models, learning and inference, Cambridge University Press 2012

Moravec vs. Harris

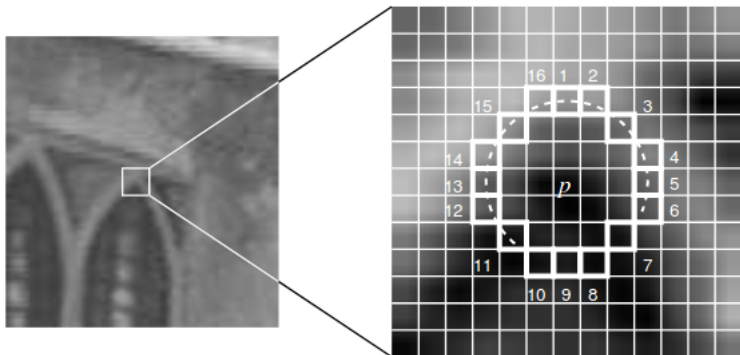


Harris



Moravec

Detektor FAST - Features from Acelerated Segment Test



Jeżeli wśród 16-tu pikseli wokół piksela p o intensywności I istnieje n pikseli, które są jaśniejsze niż $I + t$ lub ciemniejsze niż $I - t$, punkt p uznawany jest za narożnik.

E. Rosten and T. Drummond. "Fusing Points and Lines for High Performance Tracking." In: Proceedings of the IEEE International Conference on Computer Vision. 2005, pp. 1508–1511

Detektor SIFT - Scale Invariant Feature Transform - detektor i deskryptor!

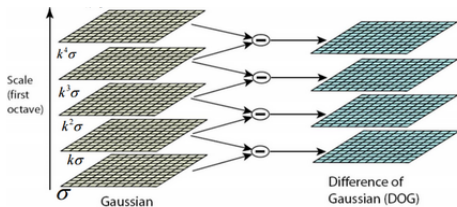
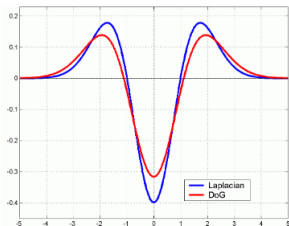
SIFT wyznacza punkty niezmiennicze ze względu na skalowanie i obroty oraz częściowo niezmiennicze na zmiany oświetlenia i punktu patrzenia.

Kroki algorytmu:

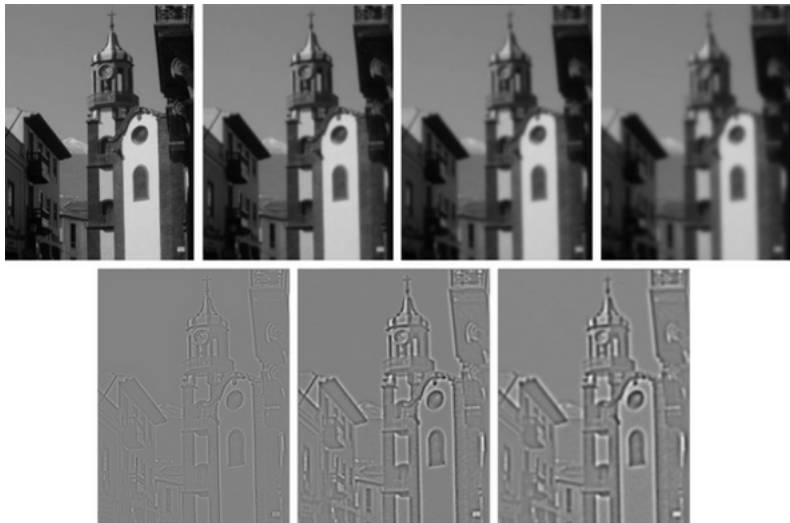
- 1 **Detekcja** punktów ekstremalnych,
- 2 Dokładna lokalizacja punktów charakterystycznych (kluczowych),
- 3 Przypisanie orientacji punktom charakterystycznym,
- 4 Tworzenie **deskryptorów** punktów charakterystycznych.

Przybliża LoG (Laplacian of Gaussian) przy pomocy DoG (Difference of Gaussian)

$$DoG = G_{k\sigma}(x, y) - G_{\sigma}(x, y)$$



Wynik DoG



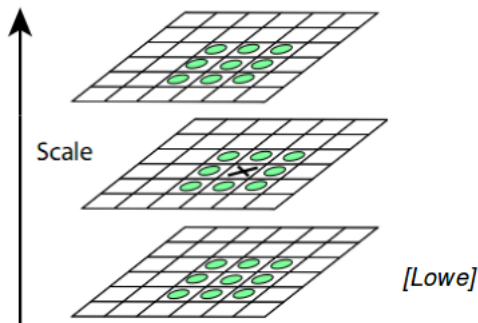
SIFT, krok 1. - skalowanie



SIFT, krok 1. - detekcja lokalnych ekstremów

Wykrywanie lokalnych ekstremów odbywa się poprzez porównanie punktu z jego sąsiadami w otoczeniu $3 \times 3 \times 3$. Punkt jest kandydatem na punkt kluczowy, gdy ma wartość mniejszą lub większą od sąsiadów.

W wyniku powstają punkty niezmiennicze względem skalowania.



SIFT, krok 2. - lokalizacja punktów kluczowych

Usunięcie mało-znaczących minimów i maksimów na obrazie wynikowym DoG używając 2. testów

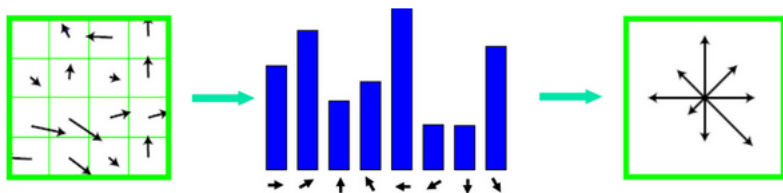
1. test: usunięcie punktów znajdujących się na jednolitym tle (lokalne minima),
2. test: usunięcie punktów leżących wewnątrz odcinków.



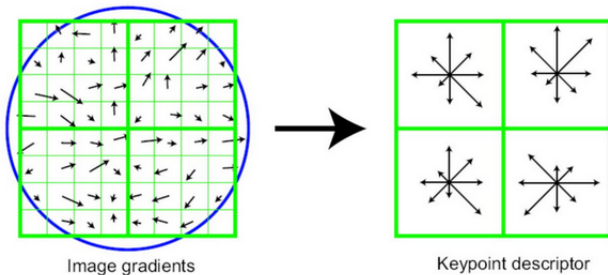
SIFT, krok 3. - przypisanie orientacji punktom

Celem jest wybranie punktów niezmienniczych względem orientacji.

- wyznaczenie orientacji gradientu dla każdego punktu z otoczenia,
- stworzenie histogramu orientacji gradientu (HoG) biorąc pod uwagę odległość od środka,
- wyznaczenie orientacji dominującej (orientacji punktu).



SIFT, krok 4. - tworzenie deskryptorów



Moduł gradientu oraz orientacja z otoczenia 16x16,
podział na regiony 4x4,

Deskryptor cechy punktu kluczowego to wektor 4x4x8 orientacji =
128 elementów.

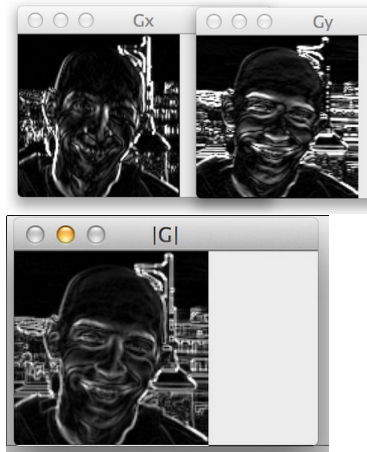
Zastosowanie SIFT



Deskryptor HoG - Histogram of Oriented Gradients

Algorytm:

- 1 Wyznaczenie gradientów (kierunek i długość),

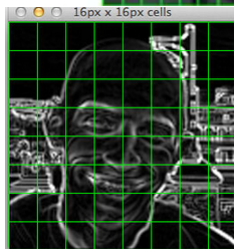
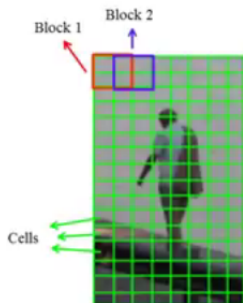


UCF Computer Vision Video Lectures, Dr. Mubarak Shah,
<https://gurus.pyimagesearch.com>

Deskryptor HoG - Histogram of Oriented Gradients

Algorytm:

- 1 Wyznaczenie gradientów (kierunek i długość),
- 2 Podział obszaru na bloki 16x16 z 50% nałożeniem (4 komórki x 8x8),



UCF Computer Vision Video Lectures, Dr. Mubarak Shah,
<https://gurus.pyimagesearch.com>

Deskryptor HoG - Histogram of Oriented Gradients

Algorytm:

- 1 Wyznaczenie gradientów (kierunek i długość),
- 2 Podział obszaru na bloki 16x16 z 50% nałożeniem (4 komórki x 8x8),
- 3 Kwantyzacja kierunku gradientu do 9 (8) pojemników (wagi zależne od długości wektora, opcjonalnie od położenia - filtr Gaussa).

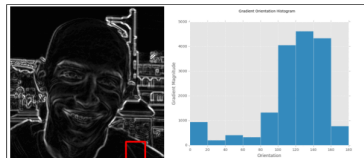
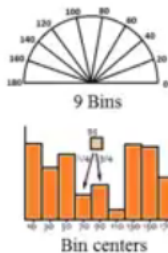


FIGURE 9. COMPUTING A HISTOGRAM OF ORIENTED GRADIENTS FOR EACH CELL IN THE IMAGE.

UCF Computer Vision Video Lectures, Dr. Mubarak Shah,
<https://gurus.pyimagesearch.com>

Spis treści:

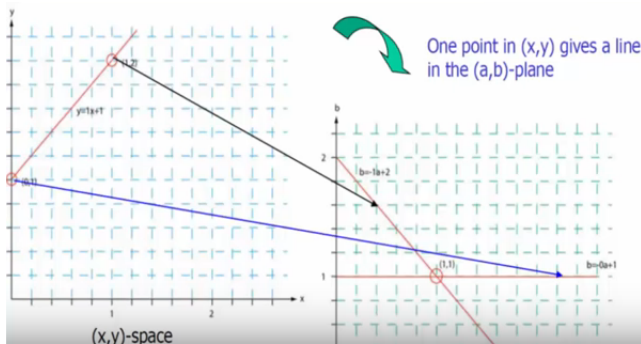
- 1 Detektory, deskryptory
- 2 Ekstrakcja cech - krawędzie
 - filtry gradientowe,
 - LoG,
 - Canny,
- 3 Ekstrakcja cech - punkty
 - detektor Moraveca,
 - detektor Harrisa (Shi-Tomasi),
 - FAST,
 - SIFT+HoG,
- 4 **Ekstrakcja cech - linie - Transformacja Hough**
- 5 Ekstrakcja cech - bloby
- 6 Rozpoznanie twarzy
 - kaskada Haara,
 - Eigenfaces,
 - Active Appearance Model,

Transformacja Hough - linie

Metoda detekcji prostych oraz krawędzi dających opisać się analitycznie (proste i okręgi) poprzez stopniowe kumulowanie wiarygodności

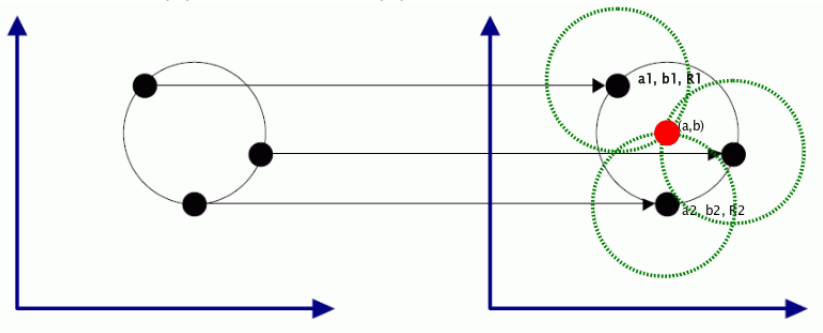


$$y = mx + c, \quad c = (-x)m + y$$



Transformacja Hough - okręgi

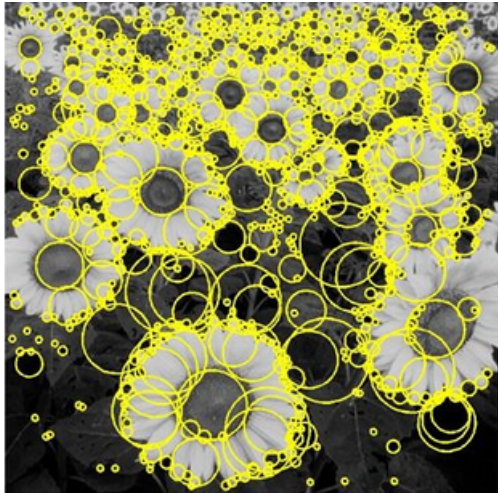
$$(x - a)^2 + (y - b)^2 = R^2$$
$$x = a + R\cos(\theta), y = b + R\sin(\theta)$$



Bloby - plamy, obszary, które wyraźnie różnią się od sąsiednich rejonów obrazu (jasnością, teksturą itp.)

Detekcja blobów

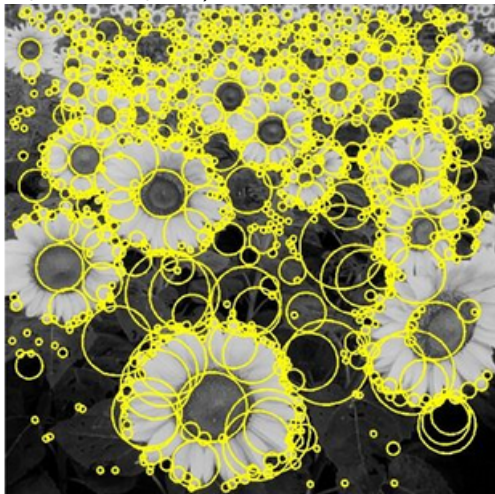
Bloby - plamy, obszary, które wyraźnie różnią się od sąsiednich rejonów obrazu (jasnością, teksturą itp.)



Detekcja blobów

Bloby - plamy, obszary, które wyraźnie różnią się od sąsiednich rejonów obrazu (jasnością, teksturą itp.)

- LoG
- DoG



Spis treści:

- 1 Detektory, deskryptory
- 2 Ekstrakcja cech - krawędzie
 - filtry gradientowe,
 - LoG,
 - Canny,
- 3 Ekstrakcja cech - punkty
 - detektor Moraveca,
 - detektor Harrisa (Shi-Tomasi),
 - FAST,
 - SIFT+HoG,
- 4 Ekstrakcja cech - linie - Transformacja Hough
- 5 Ekstrakcja cech - bloby
- 6 **Rozpoznanie twarzy**
 - kaskada Haara,
 - Eigenfaces,
 - Active Appearance Model,