

Segmentacja obrazu

Krzysztof Krawiec

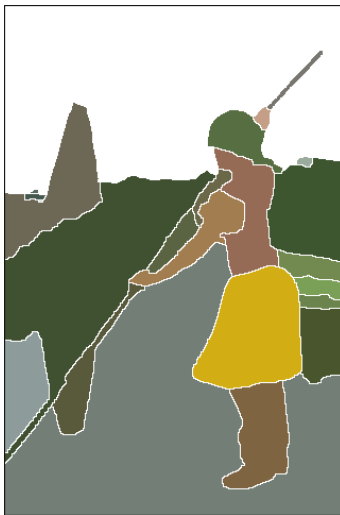
Przetwarzanie i Rozpoznawanie Obrazów

February 3, 2015

Definicja: Identyfikowanie¹ obszarów² obrazu, których wygląd jest jednolity³ dla obserwatora.

Komentarze:

1. Oznacza jednoznaczne ustalenie, do jakiego obszaru (docelowo: składowej pierwotnej) należy dany punkt obrazu,
2. Obszarów lub innych elementów (składowych pierwotnych), np. krawędzi,
3. Jednolitość może być definiowana na różne sposoby (poziom jasności, charakterystyka kolorystyczna, tekstura, etc.)

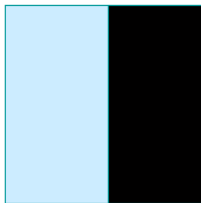
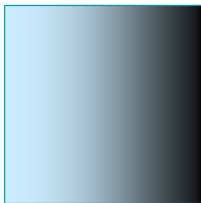


<https://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/>

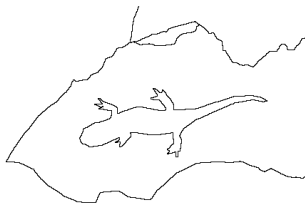
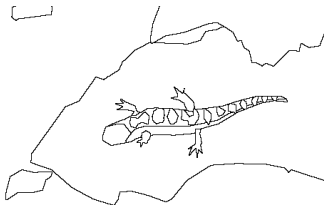
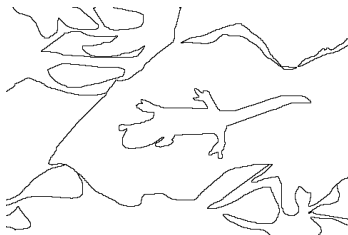
- Przejście od ilościowej do jakościowej informacji (reprezentacji obrazu).
- Segmentacja jest często zależna od zastosowania (specjalizowana); implikuje to konieczność uwzględnienia wiedzy dziedzinowej (w stopniu znacznie wyższym od etapów wcześniej omawianych).
- W ogólności segmentacja jest źle sformułowanym (ill-posed) problemem z matematycznego punktu widzenia.
- W konsekwencji algorytmy można podzielić na:
 - uniwersalne,
 - dedykowane (np. analiza 3D scen rzeczywistych).

W rzeczywistych obrazach często występują regiony, które, choć z semantycznego punktu widzenia nie powinny ulegać podziałowi na 'segmenty', ulegają mu ze względu na zmienną charakterystykę, wynikającą np. z:

- gradientu oświetlenia,
- zniekształceń perspektywicznych,



Wyniki segmentacji ręcznej (trzy niezależne osoby)



<https://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/>

Zazwyczaj rozpatrywane kryteria:

- 1 złożoność obliczeniowa,
- 2 złożoność pamięciowa,
- 3 dokładność znajdowania granic obszarów,
- 4 wierność względem obrazu oryginalnego,

Uwagi:

- przetarg – głównie pomiędzy kryterium 1 a kryterium 2

(podział ze względu na metodę):

- przez progowanie,
- przez analizę skupień,
- przez wykrywanie krawędzi,
- metody obszarowe, tj. przez rozrost i/lub podział obszarów,
- hybrydowe.
- Inne, np.
 - przez dopasowywanie wzorca (template matching)
 - wododziałowa
 - przez podział grafu
 - algorytm mean-shift

Uwagi

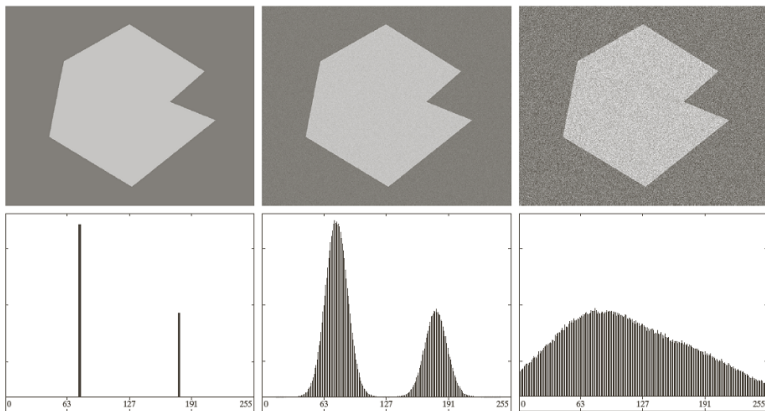
- Jedno z najbardziej intensywnie studiowanych zagadnień w widzeniu komputerowym i rozpoznawaniu obrazów (co najmniej 30 lat badań).

Idea: binaryzacja obrazu na podstawie histogramu.
(stąd alternatywna nazwa: histogram-based segmentation)

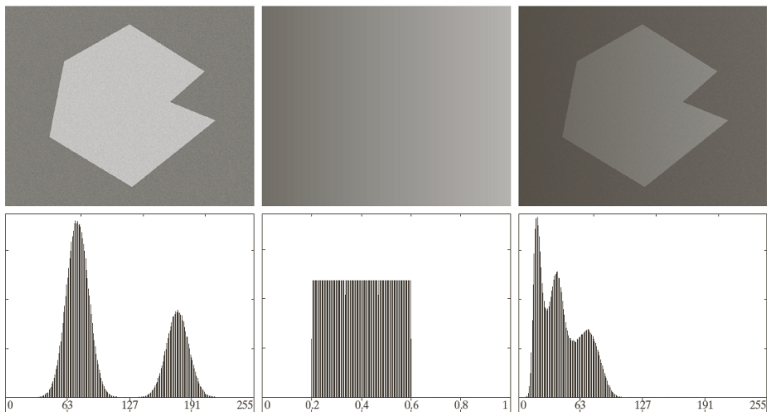
$$g(x,y) = \begin{cases} 1 & \text{if } f(x,y) > T \\ 0 & \text{if } f(x,y) \leq T \end{cases}$$

Problemy:

- dobór progu(-ów) T ,
- ignorowanie relacji sąsiedztwa (przestrzennego charakteru obrazu),
- problematyczne w obrazach kolorowych (wielomodalnych).



Obrazy i histogramy z szumem o odchyleniu standardowym: 0, 10, 50.



Obraz w prawej kolumnie to iloczyn obrazów z lewej i środkowej kolumny.

T_0 = średnia jasność w całym obrazie

$k = 0$

Do

przeprowadź progowanie z progiem T_k

$k++$

L_B = średnia jasność pikseli o jasnościach $< T_k$

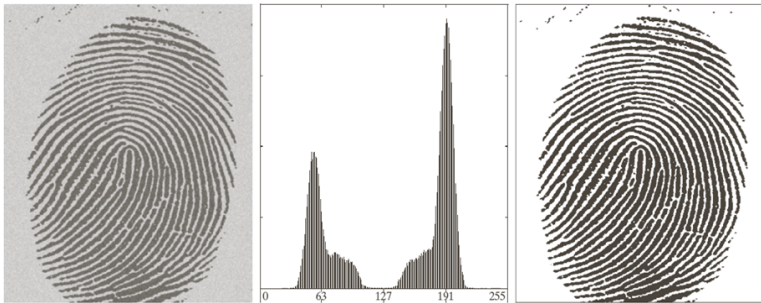
L_F = średnia jasność pikseli o jasnościach $\geq T_k$

$T_k = (L_B + L_F)/2$

While $T_k \neq T_{k+1}$

Uwagi:

- Znany także jako ISODATA
- Zasadniczo równoważny algorytmowi k-means



Idea: minimalizacja wariancji wewnątrzklasowej¹:

$$\sigma_{intra}^2(t) = P_1\sigma_1^2 + P_2\sigma_2^2$$

gdzie:

σ_i – średnia jasność w klasie [jasności] i

P_i – prawdopodobieństwo przynależności piksela do klasy i , tj.

$$P_1 = \Pr(f(x,y) < t), \quad P_2 = \Pr(f(x,y) \geq t),$$

Optymalny próg (wyszukiwany przeszukiwaniem pełnym):

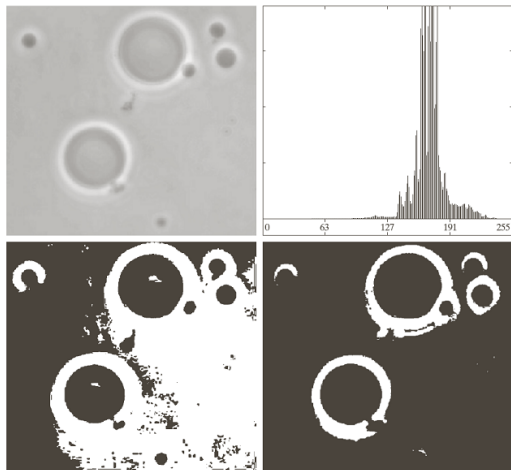
$$\arg \max_k \sigma_{intra}^2(t)$$

Otrzymana wielkość, po normalizacji (σ^2 - wariancja jasności w całym obrazie):

$$\frac{\sigma_{intra}^2(t)}{\sigma^2}$$

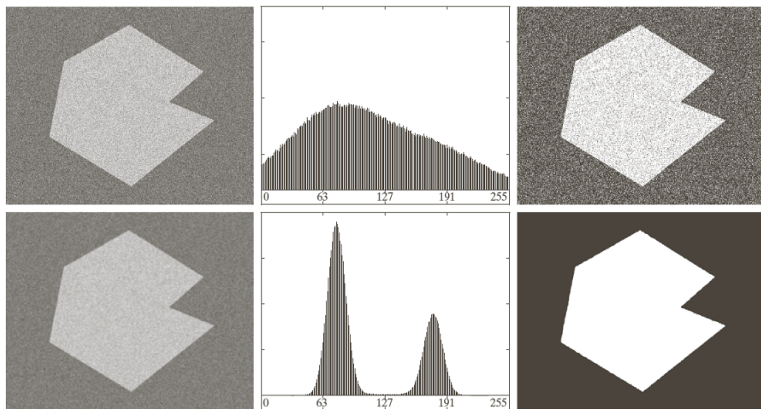
może być interpretowana jako miara separowalności.

¹Równoważna maksymalizacji wariancji międzyklasowej.

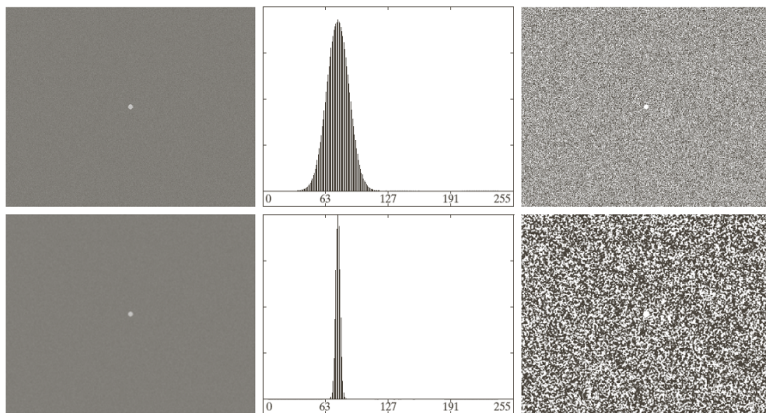


Dół lewa: wynik segmentacji algorytmem Isodata; dół prawa: wynik segmentacji algorytmem Otsu.

Przykład: Metoda Otsu w obecności szumu

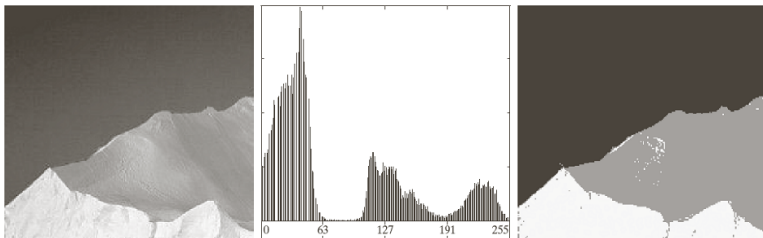


Góra: segmentacja obrazu oryginalnego. Dół: segmentacja obrazu po uprzednim przefiltrowaniu prostym filtrem uśredniającym 5×5 .



Góra: segmentacja obrazu oryginalnego. Dół: segmentacja obrazu po uprzednim przefiltrowaniu prostym filtrem uśredniającym 5×5 .

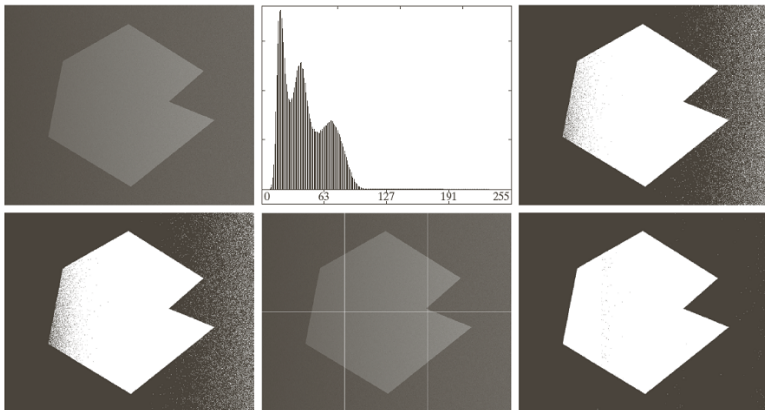
Ideę maksymalizacji odległości międzyskupieniowej da się łatwo uogólnić na wiele progów.



Idea: zamiast z całego obrazu, próg wyznaczany jest na podstawie fragmentu obrazu (lub, w skrajnym przypadku, [dużego] lokalnego otoczenia badanego punktu).

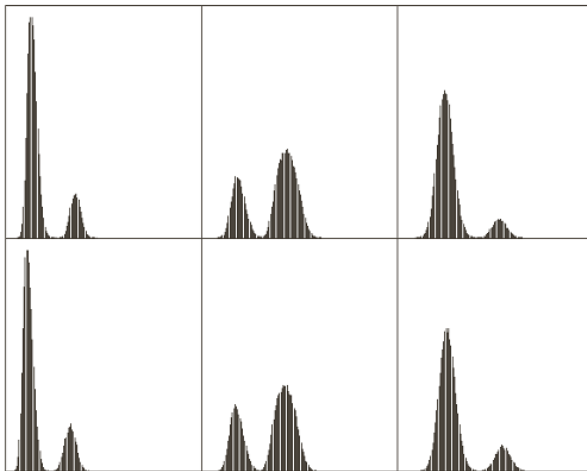
Metody wyznaczania lokalnego progu:

- Średnia,
- Mediana,
- Midrange
- Otsu

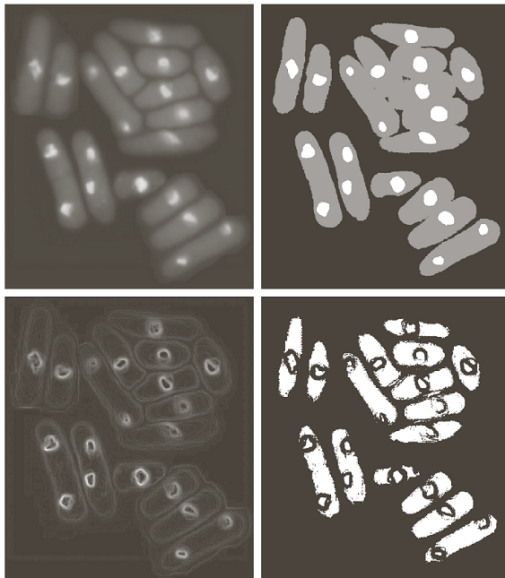


PG: Isodata; LD: Otsu; PD: Otsu na podobrazach

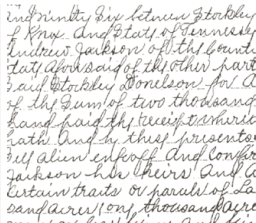
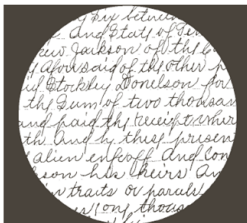
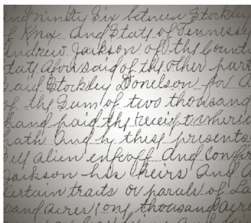
Histogramy podobrazów:



Idea: próg dla danego punktu wyliczany na podstawie jego lokalnego otoczenia
(otoczenie $n \times n$)



Próg wyliczany z fragmentu wiersza pikseli (otoczenie $n \times 1$)

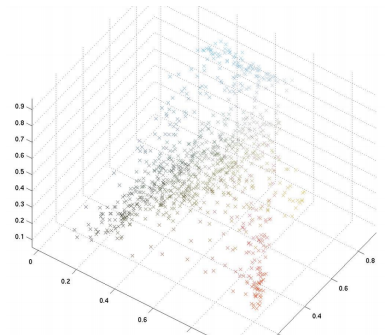
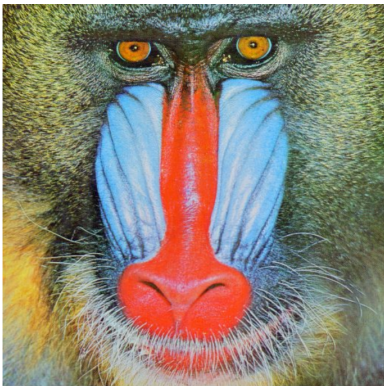


Uogólnienie segmentacji przez progowanie: punkty obrazu grupowane są na podstawie ich wielowymiarowej charakterystyki (progowanie – przypadek jednowymiarowy).

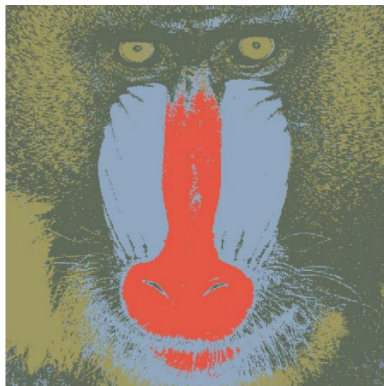
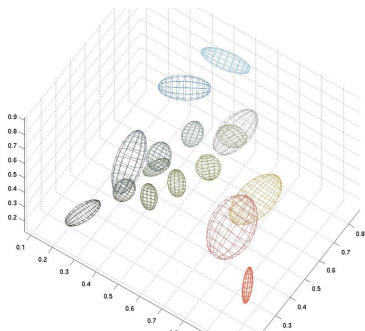
Typowa implementacja:

- klasyczna analiza skupień (algorytm Lloyd'a)
- algorytm EM (*expectation maximization*) – modelowanie wielowymiarowego rozkładu cech za pomocą sumy ważonej funkcji Gaussowskich (mixture of Gaussians)

(za: Silvio Savarese)



(za: Silvio Savarese)



Idea: zamiast wykrywać obszary, spróbujmy wykrywać granice pomiędzy nimi.

Najczęściej stosowane:

- splot (np. filtry Sobela, Robertsa, etc.),
- gradient, Laplasjan,
- podejścia morfologiczne.

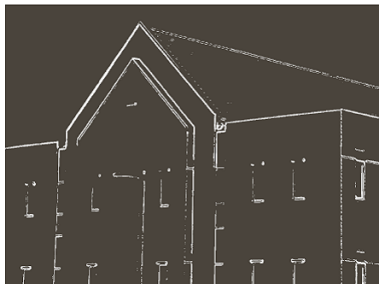
Segmentacja przez wykrywanie krawędzi

Zalety:

- dość szybkie,
- w porównaniu z segmentacją przez progowanie: brak problemów z ustalaniem wartości progu.

Wady:

- konieczność prześledzenia krawędzi i analizy topologicznej (to już nie jest tak szybkie),
- podatne na szumy,
- problemy z wykrywaniem „rozmytych” granic pomiędzy obszarami,
- niezbędny element: operator wykrywania krawędzi (por. wcześniejsze wykłady).



(obraz wejściowy, moduł gradientu, wynik algorytmu Canny)

Segmentacja przez wykrywanie krawędzi - przykład



Procedura podziału obszaru R na rozłączne obszary R_1, R_2, \dots, R_N takie, że:

- 1 Procedura jest kompletna: $\bigcup_{i=1}^N R_i = R$,
- 2 Obszary są parami rozłączne: $i \neq j \implies R_i \cap R_j = \emptyset$
- 3 Obszary R_i są spójne,
- 4 Każdy obszar powinien spełniać pewien test jednolitości (predykat logiczny P),
- 5 Dla żadnej pary przyległych obszarów, ich suma nie może spełniać P .

Zauważ: Warunki 1 i 2 definiują *partycję* (podział zbioru R)

Podejście w pewnym sensie dualne względem segmentacji przez wykrywanie krawędzi.

Wymaga sprecyzowania dwóch komponentów:

- *Strategii przeszukiwania* przestrzeni wszystkich możliwych partycji
Na przykład: startując od małych elementów obrazu, sukcesywnie łącz sąsiadujące elementy obrazu w coraz większe, o ile spełniają przyjęte test jednolitości.
- *Testu jednolitości*: funkcja, która dla pary obszarów zwraca wartość tak/nie w zależności od tego, czy można je uznać za części (reprezentację) tego samego obiektu w obrazie.
 - Może zależeć od zawartości obrazu, tj. funkcji charakterystycznej f (jasność, barwa, tekstura), ale też niekiedy od wielkości obszarów.

Test jednolitości = maksymalne odchylenie jasności względem średniego poziomu jasności w obszarze

R - obszar (zbiór punktów),

f - oddaje poziom jasności,

Średni poziom jasności: $\bar{f}(R) = \frac{1}{|R|} \sum_{p \in R} f(p)$

Test jednolitości: $\max_{p \in R} |f(p) - \bar{f}(R)| \leq T$

Niestety: Prosty test jednolitości oparty na jasności poszczególnych punktów charakteryzuje się bardzo wysokim ryzykiem popełnienia błędów pierwszego i drugiego rodzaju, tj:

1. uznać obszar jednolity za niejednolity,
2. uznać obszar niejednolity za jednolity.

Rozwiązanie problemu:

- inne kryteria jednolitości,
- inne algorytmy segmentacji.
- Inne kryteria/testy jednolitości

1. Test „dynamiczny”: bazuje nie tylko na obszarze R , ale także na obszarach $R1$ i $R2$, które zostały połączone.

Idea: porównywanie charakterystyki R z charakterystykami $R1$ i $R2$.

2. Opieranie się o bardziej złożone wielkości, np. wektorowe (histogram, macierz sąsiedztwa, etc.).

3. Obserwowanie wariancji f w ramach R , $R1$ i $R2$,

4. Branie pod uwagę długości granicy pomiędzy obszarami.

- rozrost (łączenie),
- podział,
- mieszane.

Inaczej: algorytm segmentacji przez rozrost obszarów. Jest to uogólnienie algorytmu z ziarnem.

L - lista regionów

Inicjalizuj(L) // np. regiony=pojedyncze punkty

do

for wszystkie pary regionów przyległych $(R_1, R_2) \in L \times L$

$R \leftarrow R_1 \cup R_2$

if TestJednolitości(R) then

L.Usun(R_1)

L.Usun(R_2)

L.Dodaj(R)

endif

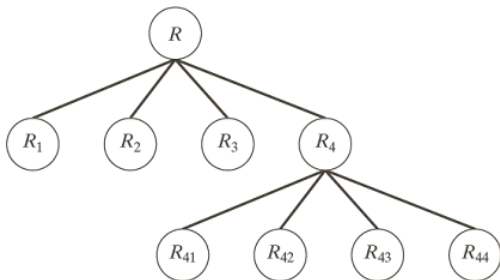
endfor

while not(WarunekStopu(L))

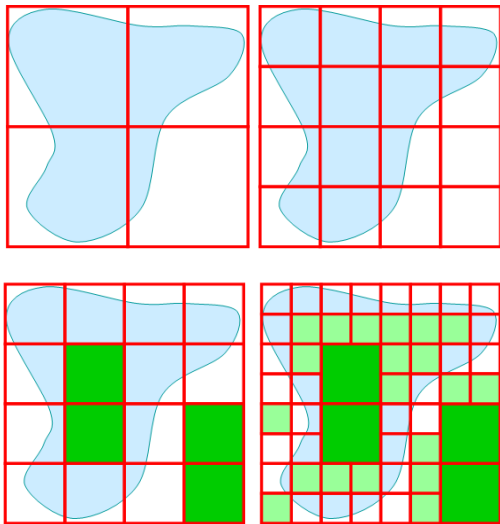
- Zastąpienie testu jednolitości **kryterium** jednolitości i wykonywanie operacji łączenia w kolejności wyznaczonej tym kryterium (podobne regiony będą łączone wcześniej)
 - czyli inaczej: przejście na algorytm zachłanny
- Zaleta: większa wierność obrazu wynikowego w stosunku do oryginalnego,
- Wada: większy koszt obliczeniowy

Idea: startując z całego obrazu, sukcesywnie dzieli go na coraz mniejsze podobrazy tak długo jak wynikowe podobrazy spełniają test jednolitości.

R_1	R_2	
R_3	R_{41}	R_{42}
	R_{43}	R_{44}



- Całkowita liczba węzłów drzewa: $\sum_{k=0}^n 4^k = \frac{4}{3}4^n$, gdzie n - liczba poziomów (głębokość) drzewa + 1.
- Zauważmy że pełne drzewo czwórkowe (dla całego obrazu) ma głębokość $\log_4(\text{liczbaPikseli})$
- Zatem: węzłów drzewa jest tylko o ok. 33% więcej niż punktów obrazu.

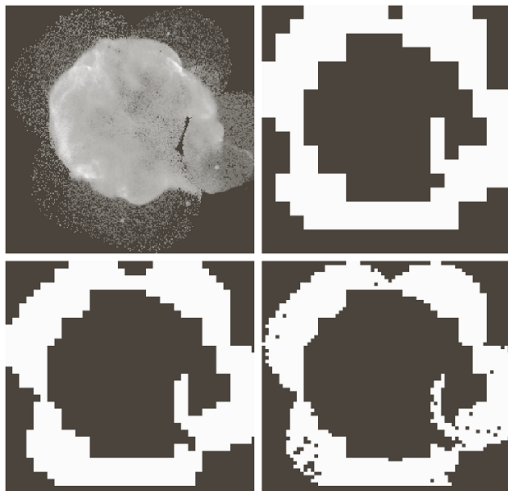


- Wymagane dalsze połączenie węzłów drzewa „w poprzek” jego struktury, zgodnie z sąsiedztwem kwadratów w obrazie.
- Szybki algorytm znajdowania sąsiadów „topologicznych” drzewa czwórkowego
-> Pavlidis

Segmentacja obrazu supernowej Cygnus Loop (pasma Rentgenowskie)
zorientowana na wydzielenie obszarów o znacznej wariancji.

Nietypowy test jednolitości: $\sigma > a \wedge \bar{f} < b$, gdzie a i b to parametry.

Obraz oryginalny i wynik segmentacji z ograniczeniem na rozmiar regionu 32x32, 16x16, i 8x8 pikseli.



Zakłada że dany jest pewien zbiór punktów startowych (ziaren, *seeds*), od których zaczyna się rozrost obszarów.

Dany punkt przyłączany jest do ziarna s jeżeli:

- spełnia test jednolitości,
- istnieje ścieżka punktów spełniających test jednolitości, łącząca punkt p z s

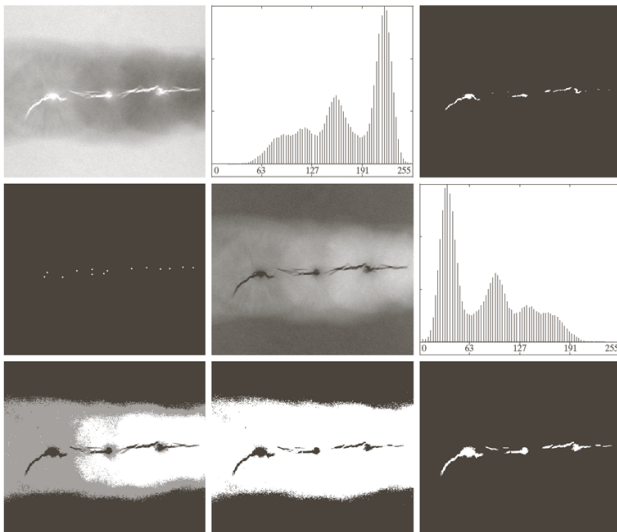
Legenda do następnego slajdu:



FIGURE 10.51 (a) X-ray image of a defective weld. (b) Histogram. (c) Initial seed image. (d) Final seed image (the points were enlarged for clarity). (e) Absolute value of the difference between (a) and (c). (f) Histogram of (e). (g) Difference image thresholded using dual thresholds. (h) Difference image thresholded with the smallest of the dual thresholds. (i) Segmentation result obtained by region growing. (Original image courtesy of X-TEK Systems, Ltd.)

Wykorzystany test jednolitości: $|f(p) - f(s)| \leq T$

Algorytm z 'ziarnem' – przykład



Motywacje: wady wcześniej wymienionych algorytmów, tj.:

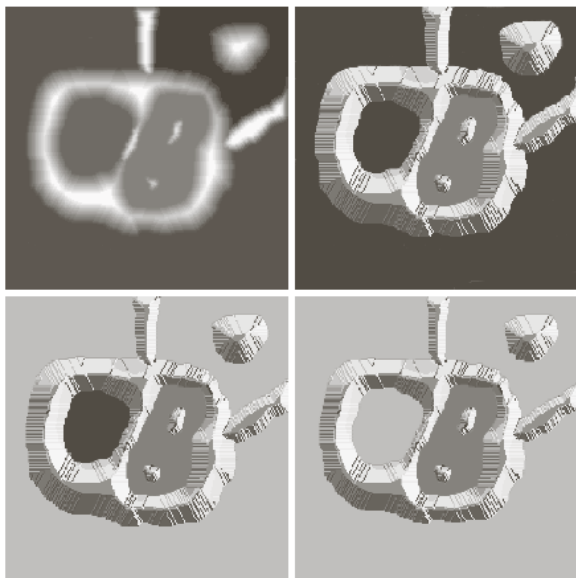
- algorytmy łączenia, startując z pojedynczych pikseli, silnie zależą od właściwości tych pikseli
 - ponadto, dla pojedynczych pikseli nie da się policzyć żadnych statystyk opisowych,
- z drugiej strony, algorytmy podziału mogą nie 'odkryć' pewnych detali obrazu.

Rozwiązanie: Algorytmy hybrydowe, tj. podziału i łączenia.

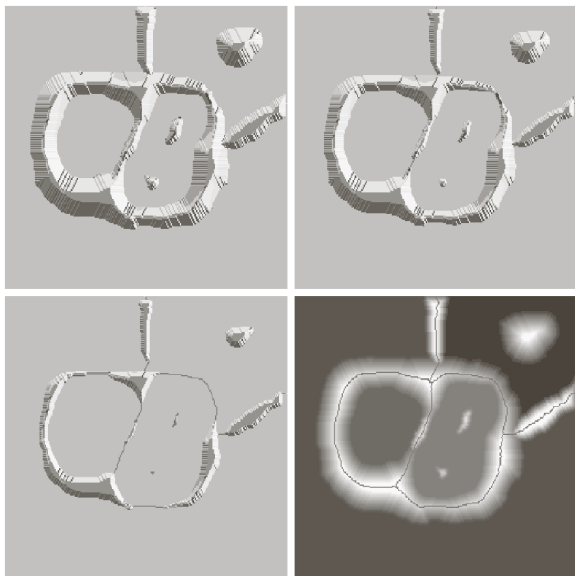
Idea: start z pośredniego poziomu drzewa (np. bloki pikseli 4x4).

Ang. *watershed segmentation*

- Klasa algorytmów segmentacji stosująca analogię map reliefowych:
 - obraz wejściowy interpretowany jako powierzchnia topograficzna (w 3 wymiarach)
 - obszar odpowiada *zlewisku*
- Wszystkie punkty obrazu można podzielić na 3 kategorie:
 - punkty reprezentujące lokalne minima,
 - punkty leżące na 'zbozczach'
 - punkty leżące na 'graniach'
- Punkty w których spotykają się rozlewiska wyznaczają granice pomiędzy obszarami.
- Punkt przynależy do tego zlewiska (obszaru) do którego spłynęłaby kropla wody upuszczona na obraz gradientowy.



Obraz oryginalny, interpretacja topograficzna (rendering 3D, z cieniem), oraz dwa pierwsze etapy zalewania zlewisk (począwszy od nainiżej położonych dolin



Woda wypełniająca poszczególne zlewiska zaczyna spotykać się na granicach;
budujemy 'tamy' aby nie dopuścić do połączenia zlewisk.

- Istotne: Obraz wejściowy dla algorytmu segmentacji wododziałowej jest zazwyczaj obrazem gradientowym otrzymanym z oryginalnego obrazu.
- Możliwe realizacje praktyczne:
 - „Rozlewanie” (flood fill) startujące z lokalnych minimów gradientu i rejestrowaniem punktów gdzie gdzie spotykają się zlewiska
 - Wykorzystanie odległości od minimów lokalnych
- (dają różne wyniki)

- Przepuść obraz przez filtr gradientowy
- Przypisz unikalne etykiety wszystkim punktom leżącym w lokalnych minimach jasności i umieść je w kolejce priorytetowej Q posortowanej po jasnościach
- Powtarzaj do opróżnienia Q :
 - Pobierz punkt p z Q
 - Dla każdego sąsiada q punktu p
 - jeżeli żaden sąsiad punktu q nie ma etykiety różnej od etykiety p , skopiuj etykietę z p do q i wstaw q do Q
 - w przeciwnym razie pomiń q (nie dostaje żadnej etykiety)
- Po zakończeniu działania punkty bez etykiet stanowią granice zlewisk.

Dane i oznaczenia:

- g - obraz gradientowy, z najmniejszą jasnością min i największą max
- R - liczba lokalnych minimów w obrazie gradientowym
- $M_i, i = 1, \dots, R$ - zbiór punktów obejmujących i -te minimum (w ogólności region)
- $T[n]$ - zbiór punktów obrazu o jasności mniejszej niż n ,

$$T[n] = \{(x, y) : g(x, y) < n\}$$

Cel: znalezienie $C(M_i)$, tj. zbiorów punktów stanowiących zlewisko związane z minimum M_i

- Do ostatecznego wyniku, tj. $C(M_i)$, będziemy zbliżać się budując kolejne przybliżenia, $C_n(M_i)$, gdzie n jest 'poziomem' wody zwiększanym o dyskretne przyrosty.
- n zmieniamy od pewnej wartości początkowej $min + 1$ do końcowej $max + 1$

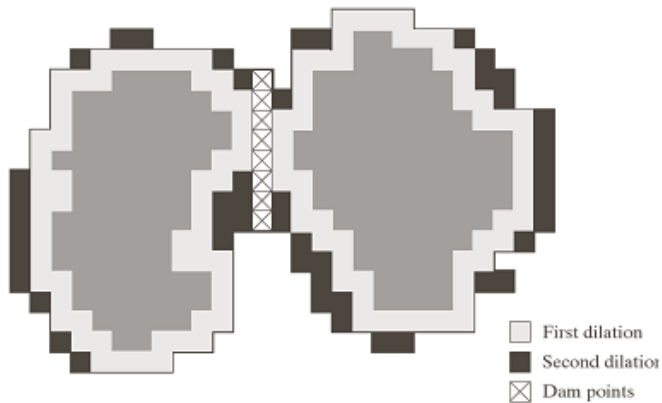
Sukcesywnie liczymy kolejne C :

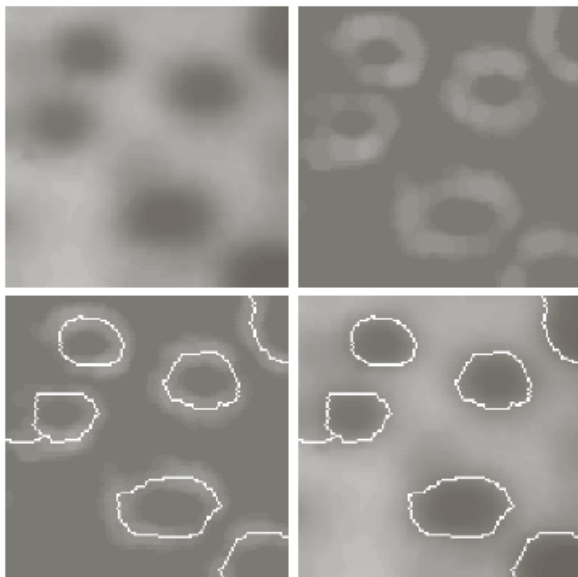
```

C[min+1]:=T[min+1] // Inicjalizacja
for n=min+1 to max+1
  Q:=ConnectedComponents(T[n])
  C[n]:=∅
  foreach q∈Q
    S:=q∩C[n-1]
    if S=∅ then
      // napotkalismy nowe lokalne minimum
      C[n]:=C[n-1]∪{q}
    elseif |S|=1 then
      // q jest w zlewisku lokalnego minimum
      // (napotkanego już wcześniej)
      C[n]:=C[n-1]∪{q}
    else // |S|>1
      // S zawiera 2 lub więcej zlewisk które się łączą
      // => konieczność zbudowania 'tamy' =>

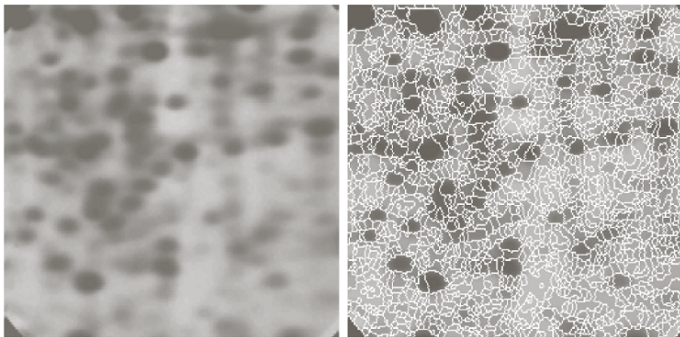
```

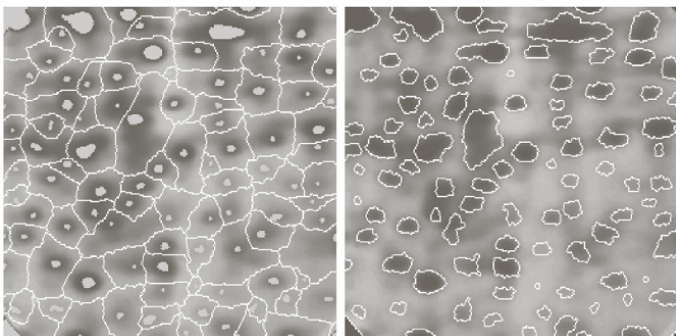
- Budowanie tamy ogranicza się do obszaru S (tj. nie dotyczy pozostałych elementów Q).
 - jedna z konsekwencji: pracujemy tylko z pikselami o jasności n
- Startując z S , przeprowadzamy dylatację elementem strukturalnym 3×3 .
- Dylatacje powielamy tak długo aż dojdzie do zetknięcia się zlewisk. Nie dopuszczamy do zlania się zlewisk.
- Punkty w których fronty procesu dylatacji spotykają się ze sobą definiują tamę.
- Punktom tamy przypisujemy jasność $\max+1$ w g . Dzięki temu punkty te nie zostaną naruszone do końca działania algorytmu.
- Wynikowe obszary (po dylatacji) dodajemy do $C[n]$





- Przed obliczeniami warto policzyć histogram obrazu g :
 - z histogramu natychmiast dostajemy wartości \min i \max
 - być może nie wszystkie jasności występują w obrazie; jeśli tak jest to niewystępujące poziomy jasności warto pominąć (tj. zwiększyć n o odpowiednio większe przyrosty).
- Częsty problem: oversegmentation. Przyczyna: duża liczba minimów lokalnych.
 - Możliwe rozwiązanie: wykorzystanie znaczników (markers)
 - Znaczniki definiują nowe punkty startowe dla procesu budowania zlewisk (czyli $T[\min+1]$).





Znaczniki: lokalne minima w *oryginalnym* obrazie przepuszczonym przez filtr dolnoprzepustowy. Wynik w prawym obrazie uzyskany przez zastosowanie algorytmu wododziałowego do każdego z regionów z lewego obrazu z osobna.

- Segmentacja wododziałowa da się wprost zastosować dla obrazów z gradacją stopni szarości.
- Co z obrazami binarnymi?
 - Dla 'płaskich' obszarów, zlewiska w segmentacji wododziałowej przyrastają bardzo szybko.
- Czy istnieje możliwość takiego 'uciąglenia' obrazu binarnego aby segmentacja wododziałowa dała sensowny wynik?
 - Odpowiedź: Distance transform.

Distance transform (transformacja odległościowa): (wymaga obrazu binarnego)

Dla każdego piksela należącego do obiektu:

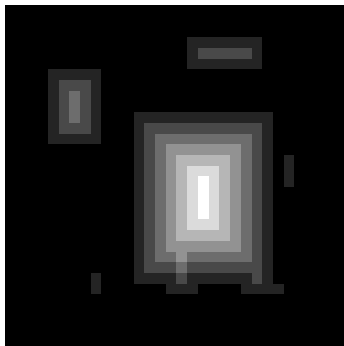
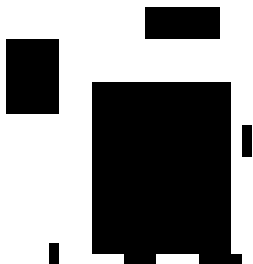
- Znajdź najbliższego sąsiada należącego do tła
- Oblicz odległość względem tego sąsiada
- Przypisz pikselowi wynikowemu wartość tej odległości.

Stopnie szarości w obrazie wynikowym wizualizują powyższą odległość.

	1	1	1	1	1	
	1	1	1	1	1	
	1	1	1	1	1	
	1	1	1	1	1	
	1	1	1	1	1	

	1	1	1	1	1	
	1	2	2	2	1	
	1	2	3	2	1	
	1	2	2	2	1	
	1	1	1	1	1	

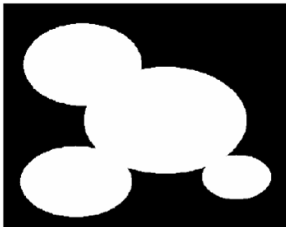
Obraz oryginalny i transformacja odległościowa przy metryce miejskiej (dla powyższego obrazu efekt dla metryki Euklidesowej byłby taki sam).



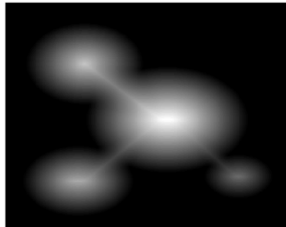
Uwagi:

- Powiązanie ze szkieletem obiektu (zobacz rozdział Opis obrazów i obiektów).
- Realizacja techniczna: obraz wynikowy musi być typu rzeczywistego (float, double).
- Szybka implementacja z wykorzystaniem erozji.
 - Kształt elementu strukturalnego determinuje metrykę transformaty.
 - Możliwe usprawnienia?

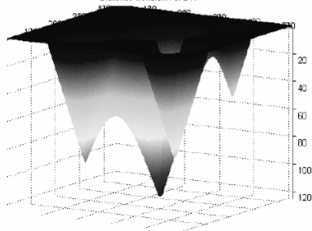
BW



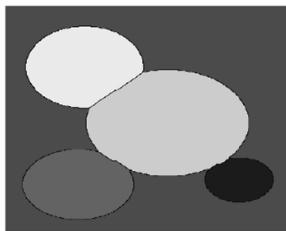
Distance transform of BW



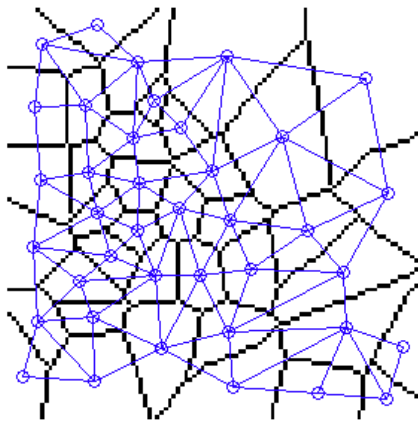
Distance transform of BW



Watershed transform of BW



- Mapa pokolorowanych obszarów
- Graf przyległości obszarów (*region adjacency graph*, RAG), graf nieskierowany taki że:
 - węzły = obszary
 - krawędzie = relacja sąsiedztwa pomiędzy obszarami.



Graf czy multigraf?

- Planarny, a zatem w konsekwencji:
 - Spełnia wzór Eulera: $vertices - edges + facets = 2$
 - Da się pokolorować 4 kolorami
- Jeżeli piksele granic pomiędzy obszarami wraz z relacją ich sąsiedztwa potraktować jako graf, to RAG jest grafem dualnym względem takiego grafu
- Mała liczba krawędzi a duża liczba węzłów: dogodne reprezentowanie w postaci listy,
- Cenna informacja dla etapu wnioskowania z obrazu; łatwo zdefiniować wiele informatywnych cech RAG, np.:
 - v, e, f
 - Średni stopień wierzchołka
- ... i/lub pojedynczych obszarów/węzłów w RAG:
 - Stopień wierzchołka (np. przypadek szczególny: stopień=1)
 - Średnia odległość regionu przyległego

Analiza wyników działania algorytmów segmentacji 'na oko' nie jest dobrym pomysłem. Potrzeba obiektywnych miar.

Pytania:

- Jak odnieść wynik działania algorytmu segmentacji do rzeczywistych regionów w obrazie?
- Jak porównać wyniki działania dwóch algorytmów segmentacji?

Jedna z często wykorzystywanych miar: Dice statistics:

$$S(R_A, R_B) = \frac{2|R_A \cap R_B|}{|R_A \cup R_B|}$$

gdzie R_A i R_B to

- Obszary znalezione przez dwa algorytmy segmentacji A i B, lub
 - Obszar zdefiniowane przez eksperta (*ground truth*, A) i obszar znaleziony przez algorytm B
- Bardzo zbliżony do Jaccard index: $|R_A \cap R_B| / |R_A \cup R_B|$
- Można uogólnić na wiele obszarów

- Wiele algorytmów segmentacji da się uogólnić na przypadek obrazów wielokanałowych.
- Wymagana modyfikacja: redefinicja kryterium (-ów) jednolitości obszarów.
- W ogólności po segmentacji obrazów wielokanałowych można często spodziewać się lepszych wyników (niż dla obrazów monochromatycznych), ponieważ dysponujemy bogatszą informacją o poszczególnych pikselach.

Segmentacja obrazów kolorowych – przykład





[D. Comaniciu and P. Meer]

Czy segmentacja bez analizy „semantycznej” obrazu jest zawsze możliwa ?

- Nie. W wielu zastosowaniach wykorzystanie uniwersalnych algorytmów segmentacji (nawet bardzo wyrafinowanych) prowadzi do powstania granic pomiędzy obszarami tam, gdzie ich nie ma.
- Pełna interpretacja obrazu (image understanding) wymaga dodatkowej wiedzy dziedzinowej.