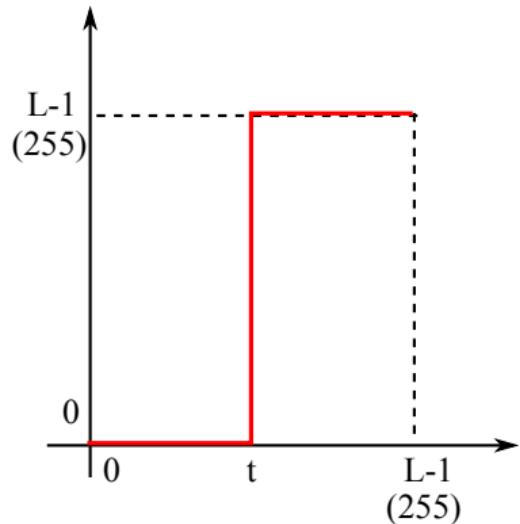


Procesarea Imagineilor Digitale

Curs - Thresholding

Universitatea "Transilvania" din Brașov

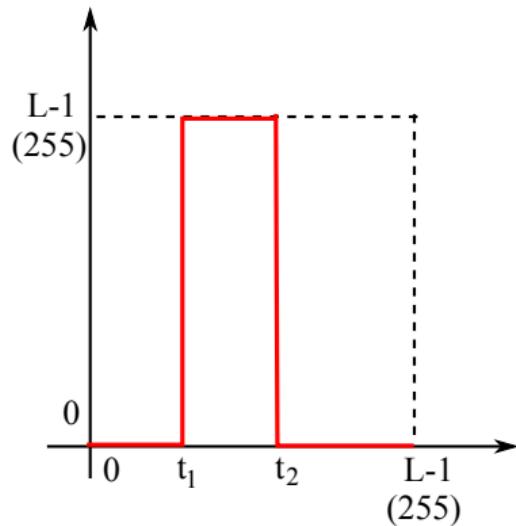
Binarizarea



- transformă o imagine în tonuri de gri într-o imagine binară în funcție de un prag - *threshold* - t
- partionează imaginea în obiecte și fundal

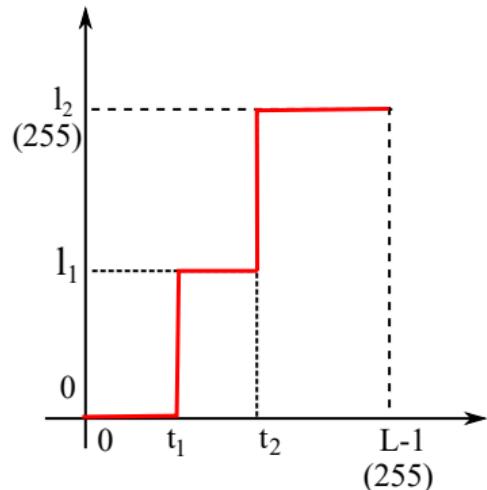
$$T(r) = \begin{cases} 0, & r \leq t \\ 255, & r > t \end{cases}$$

Segmentare cu două praguri



$$T(r) = \begin{cases} 0, & r \leq t_1 \\ 255, & t_1 < r \leq t_2 \\ 0, & r > t_2 \end{cases}$$

Segmentare cu două praguri



$$T(r) = \begin{cases} 0, & r \leq t_1 \\ l_1, & t_1 < r \leq t_2 \\ l_2, & r > t_2 \end{cases}$$

Thresholding - Prag global automat

Cele mai simple metode de selectare a unui prag global automat:

- Pe baza mediei aritmetice: $t = \text{mean}(f)$
- Prag **mid-range**:

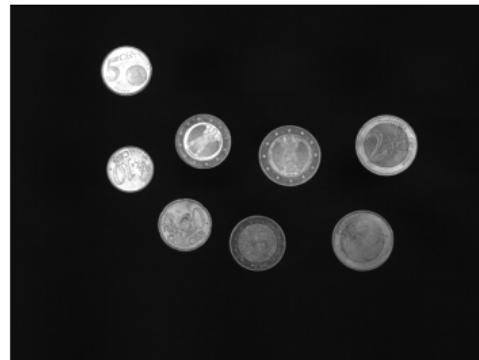
$$t = \text{round} \left(\frac{\max(f) + \min(f)}{2} \right)$$

- **Quantile threshold** - presupune faptul ca se poate estima procentul b al pixelilor de fundal în raport cu numărul total de pixeli

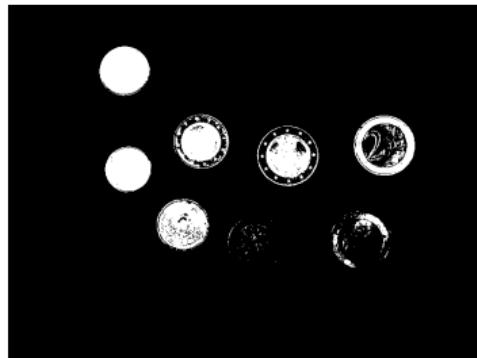
$$t = \min \left\{ j \left| \sum_{k=0}^{k=j} n_k \right\} \geq n * b \right.$$

unde n_k = nr de pixeli de culoarea k , n = nr de pixeli din imagine.

Thresholding - Prag global automat



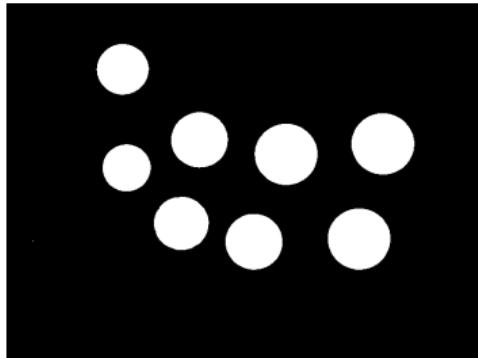
Imagine
originală



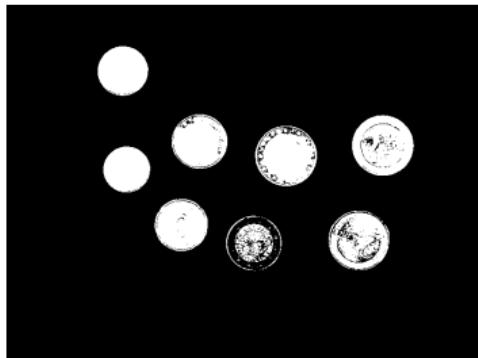
Mid-range
Threshold

Binarizare cu

Mean
Threshold



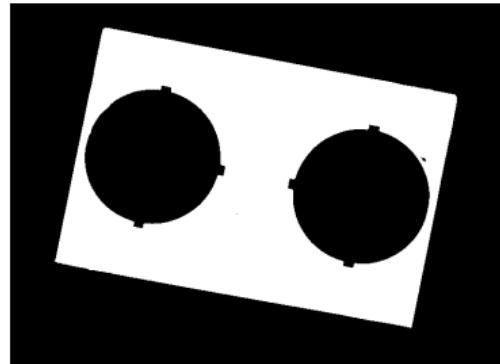
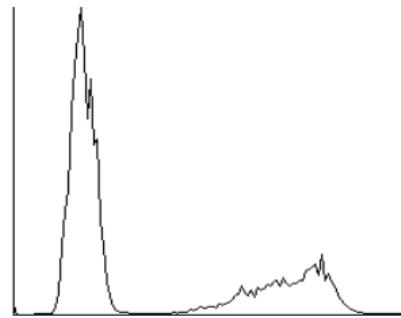
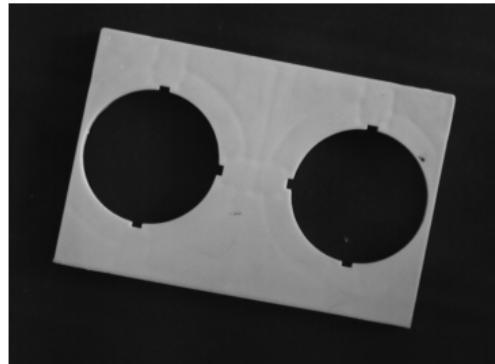
Quantile
Threshold



Praguri automate - histograma

- Calculul unui prag automat are la bază proprietăți statistice ale imaginii
- Histograma relativă: $h : \{0, 1, \dots, 255\} \rightarrow [0, 1]$, $h(k) = n_k/n$, n_k = numărul de pixeli de nivel k .
 - - poate fi considerată funcție de repartiție a nivelurilor de gri
 - - Notăm $p(k) := h(k)$ - probabilitatea de apariție a unui anumit nivel de gri în imagine

Thresholding - Prag global automat



Proprietăți statistice ale unui set de date

Se consideră un set de date $\{x_1, x_2, \dots, x_k\}$

- **Media:**

$$\mu = \frac{1}{k} \sum_{j=1}^k x_j$$

- **Abaterea medie pătratică / varianța:**

$$\sigma^2 = \frac{1}{k} \sum_{j=1}^k (x_j - \mu)^2$$

- **Abaterea medie:**

$$var = \sigma = \sqrt{\frac{1}{k} \sum_{j=1}^k (x_j - \mu)^2}$$

Calculul proprietăților statistice pe baza histogramei

Fie p funcția de probabilitate pentru nivelurile de gri ale imaginii, atunci pentru pixelii din intervalul $[t_1, t_2]$:

- **Media:**

$$\mu_{[t_1, t_2]} = \frac{\sum_{k=t_1}^{t_2} kp(k)}{\sum_{k=t_1}^{t_2} p(k)} \Rightarrow \mu = \sum_{k=0}^{255} kp(k)$$

- **Varianța:**

$$\sigma_{[t_1, t_2]}^2 = \frac{\sum_{k=t_1}^{t_2} (k - \mu_{[t_1, t_2]})^2 p(k)}{\sum_{k=t_1}^{t_2} p(k)} \Rightarrow \sigma^2 = \sum_{k=0}^{255} (k - \mu)^2 p(k)$$

Prag global - *ISODATA clustering - Intermeans*

Algoritm iterativ: Partiționarea pixelilor în două clase C_1 (regiunea de interes) și C_2 (fundalul) pe baza pragului global t

- ① Selectează un prag inițial t_0 - (ex. media între nivelul maxim și nivelul minim);

Prag global - *ISODATA clustering - Intermeans*

Algoritm iterativ: Partiționarea pixelilor în două clase C_1 (regiunea de interes) și C_2 (fundalul) pe baza pragului global t

- ① Selectează un prag inițial t_0 - (ex. media între nivelul maxim și nivelul minim);
- ② La pasul k partiționează imaginea pe baza lui t_k în C_1 - toți pixelii $> t_k$ - și C_2 - toți pixelii $\leq t_k$;

Prag global - *ISODATA clustering - Intermeans*

Algoritm iterativ: Partiționarea pixelilor în două clase C_1 (regiunea de interes) și C_2 (fundalul) pe baza pragului global t

- ① Selectează un prag inițial t_0 - (ex. media între nivelul maxim și nivelul minim);
- ② La pasul k partiționează imaginea pe baza lui t_k în C_1 - toți pixelii $> t_k$ - și C_2 - toți pixelii $\leq t_k$;
- ③ Calculează mediile μ_1 și μ_2 pentru C_1 și respectiv C_2 ;

Prag global - *ISODATA clustering - Intermeans*

Algoritm iterativ: Partiționarea pixelilor în două clase C_1 (regiunea de interes) și C_2 (fundalul) pe baza pragului global t

- ① Selectează un prag inițial t_0 - (ex. media între nivelul maxim și nivelul minim);
- ② La pasul k partiționează imaginea pe baza lui t_k în C_1 - toți pixelii $> t_k$ - și C_2 - toți pixelii $\leq t_k$;
- ③ Calculează mediile μ_1 și μ_2 pentru C_1 și respectiv C_2 ;
- ④ Calculează $t_{k+1} = \frac{\mu_1 + \mu_2}{2}$;

Prag global - *ISODATA clustering - Intermeans*

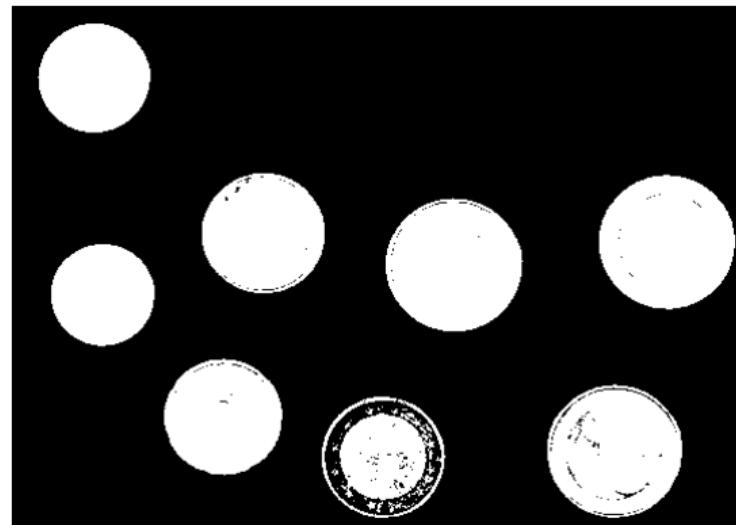
Algoritm iterativ: Partiționarea pixelilor în două clase C_1 (regiunea de interes) și C_2 (fundalul) pe baza pragului global t

- ① Selectează un prag inițial t_0 - (ex. media între nivelul maxim și nivelul minim);
- ② La pasul k partiționează imaginea pe baza lui t_k în C_1 - toți pixelii $> t_k$ - și C_2 - toți pixelii $\leq t_k$;
- ③ Calculează mediile μ_1 și μ_2 pentru C_1 și respectiv C_2 ;
- ④ Calculează $t_{k+1} = \frac{\mu_1 + \mu_2}{2}$;
- ⑤ Repetă pașii 2 - 4 până când $t_{k+1} = t_k$.

Algoritmul - *Intermeans* - Exemple



$t = 83$



Prag global optimal

Prag global optimal - prag t care optimizează o anumită funcție obiectiv
Pragul partiziionează pixelii în două clase: fond - obiect(e)

Metoda Otsu - minimizează eroarea de clasificare a unui pixel prin maximizarea raportului varianță inter-clasă / varianță intra-clasă

Algoritmul pragului de eroare minimă -Minimum error threshold - minimizează o funcție de eroare de clasificare

Algoritmul de maximizare a entropiei - maximizează funcția de entropie calculată pe baza celor două clase considerate

Prag global optimal - Metoda Otsu

Problematică partiționarea pixelilor în două clas C_1 (fundal) și C_2 (obiect) cu pragul t

Prag global optimal - Metoda Otsu

Problematică partitōnarea pixelilor în două clas C_1 (fundal) și C_2 (obiect) cu pragul t

Prag bun

- “indivizii” undei clase sunt cât mai asemănători \Rightarrow varianța mică
- cele două clase - cât mai diferite \Rightarrow “centrele” (date prin medii) cât mai depărtate

Prag global optimal - Metoda Otsu

Problematică partiționarea pixelilor în două clas C_1 (fundal) și C_2 (obiect) cu pragul t

Prag bun

- “indivizii” undei clase sunt cât mai asemănători \Rightarrow varianță mică
- cele două clase - cât mai diferite \Rightarrow “centrele” (date prin medii) cât mai depărtate

Ce ne trebuie: proprietăți statistice ale celor două clase (depind de t).

Prag global optimal - Metoda Otsu

Proprietățile statistice clasa C_1	Proprietățile statistice clasa C_2
Probabilitatea apartenenței la C_1	Probabilitatea apartenenței la C_2
$P_1(t) = \sum_{k=0}^t p(k)$	$P_2(t) = \sum_{k=t+1}^{255} p(k)$
Media	
$\mu_1(t) = \frac{1}{P_1(t)} \sum_{k=0}^t kp(k)$	$\mu_2(t) = \frac{1}{P_2(t)} \sum_{k=t+1}^{255} kp(k)$
Varianța	
$\sigma_1^2(t) = \frac{1}{P_1(t)} \sum_{k=0}^t (k - \mu_1(t))^2 p(k)$	$\sigma_2^2(t) = \frac{1}{P_2(t)} \sum_{k=t+1}^{255} (k - \mu_2(t))^2 p(k)$

$p(k)$ - probabilitatea nivelului k , dată prin histograma normalizată

Obs: $P_1(t) + P_2(t) = 1 \quad \forall t \in [0, 255]$

Prag global optimal - Metoda Otsu

Varianță inter-clase - *between-class variance* este definită prin:

$$\begin{aligned}\sigma_B^2(t) &= P_1(t)(\mu_1(t) - \mu)^2 + P_2(t)(\mu_2(t) - \mu)^2 = \\ &= P_1(t)P_2(t)(\mu_1(t) - \mu_2(t))^2\end{aligned}$$

unde

$$\mu = \sum_{k=0}^{255} kp(k) = P_1(t)\mu_1(t) + P_2(t)\mu_2(t)$$

este media nivelerelor de gri pentru întreaga imagine.

Prag global optimal - Metoda Otsu

Varianța inter-clase - *between-class variance* este definită prin:

$$\begin{aligned}\sigma_B^2(t) &= P_1(t)(\mu_1(t) - \mu)^2 + P_2(t)(\mu_2(t) - \mu)^2 = \\ &= P_1(t)P_2(t)(\mu_1(t) - \mu_2(t))^2\end{aligned}$$

unde

$$\mu = \sum_{k=0}^{255} kp(k) = P_1(t)\mu_1(t) + P_2(t)\mu_2(t)$$

este media nivelelor de gri pentru întreaga imagine.

Varianța intra-clase - *within-class variance* este definită prin:

$$\sigma_W^2(t) = P_1(t)\sigma_1^2(t) + P_2(t)\sigma_2^2(t)$$

Prag global optimal - Metoda Otsu

Prag optimal: - prag care maximizează raportul

$$\eta = \frac{\sigma_B^2(t)}{\sigma_W^2(t)}.$$

Varianță totală pentru imagine:

$$\sigma^2 = \sum_{k=0}^{255} (k - \mu)^2 p(k) = \sigma_B^2(t) + \sigma_W^2(t)$$

Rezultă: $\sigma_B^2(t)$ crește $\Rightarrow \sigma_W^2(t)$ scade.

Prag global optimal - Metoda Otsu

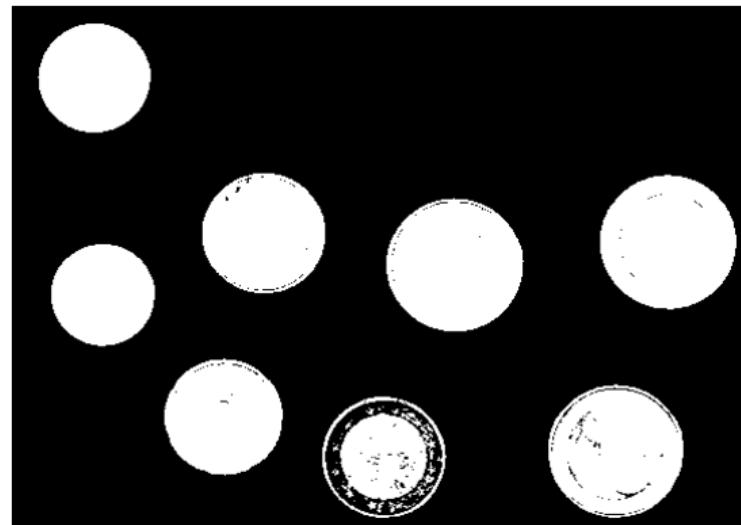
Practic:

- calculează $\sigma_B^2(t)$ pentru fiecare prag $0 < t < 255$
- selectează acel t care maximizează $\sigma_B^2(t)$.
- Dacă maximul se obține pentru mai mult de o singură valoare, atunci pragul optim se consideră a fi media diferențelor valori pentru care s-a obținut acest maxim.

Prag global optimal - *Otsu* - Exemple



$t = 83$

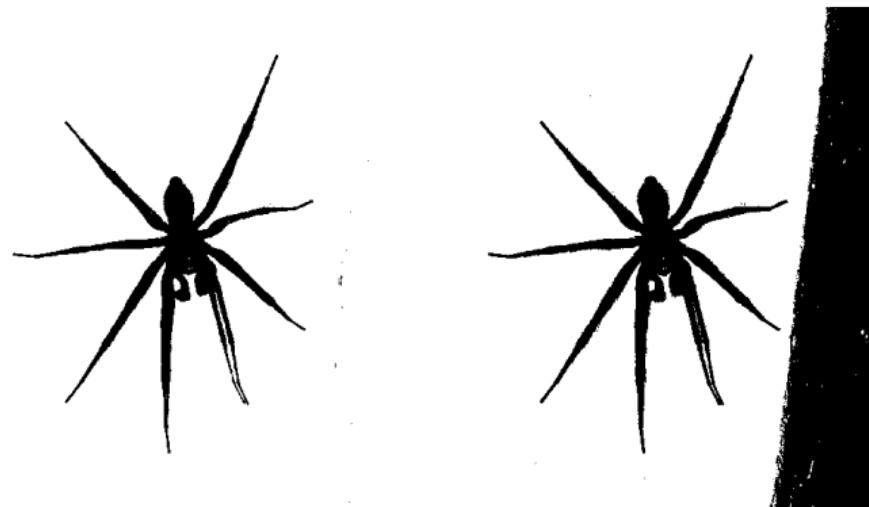


Prag global optimal - *Otsu* - Exemple



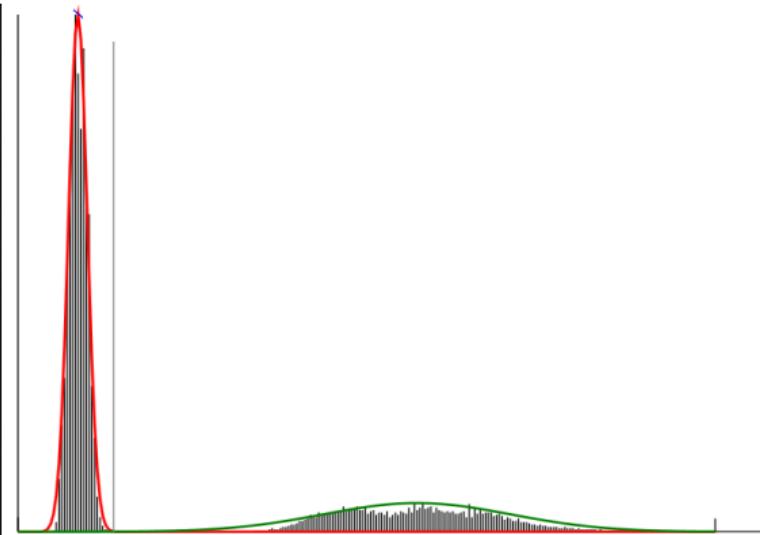
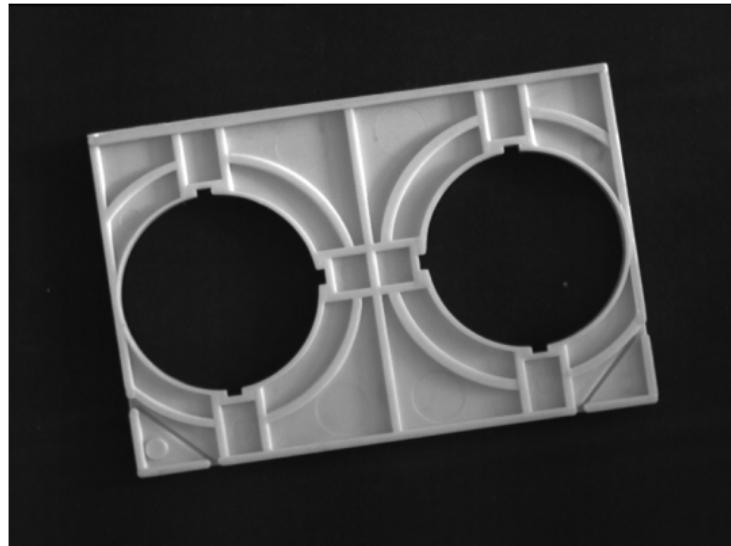
$t_{Intermeans} = 118,$

$t_{Otsu} = 150$



Prag global optimal - Eroare minimă - *Minimum Error Threshold*

Idee: se presupune cele două clase C_1 și C_2 pot fi reprezentate prin 2 repartiții gaussiene.



Prag global optimal - Eroare minimă - *Minimum Error Threshold*

Idee: se presupune cele două clase C_1 și C_2 pot fi reprezentate prin 2 repartiții gaussiene.

Deci: $P(k|C_i)$ = probabilitatea ca valoarea k să apară în clasa C_i e dată de repartitia Gauss

$$P(k|C_i) = \frac{1}{\sqrt{2\pi}\sigma_i} e^{\frac{-(k-\mu_i)^2}{2\sigma_i^2}}$$

unde μ_i - media valorilor și σ_i^2 - varianța din clasa C_i , $i = 1, 2$

Prag global optimal - Eroare minimă - *Minimum Error Threshold*

Scop: clasificarea pixelilor - dacă găsesc valoarea $k \Rightarrow$ face parte din C_1 sau din C_2 ?

Ne interesează:

- $P(C_1|k)$ - probabilitatea ca să fie clasa C_1 , dacă s-a găsit valoarea k
- $P(C_2|k)$ - probabilitatea ca să fie clasa C_2 , dacă s-a găsit valoarea k

$$P(C_i|k) = \frac{P(k|C_i)P(C_i)}{p(k)}$$

Decizie Bayesiană: dacă $P(C_1|k) > P(C_2|k) \Rightarrow$ atribui k clasei C_1 , altfel atribui clasei C_2

Se minimizează riscul de a face o clasificare greșită.

Prag global optimal - Eroare minimă - *Minimum Error Threshold*

$$P(C_1|k) > P(C_2|k) \Leftrightarrow \frac{P(k|C_1)P(C_1)}{p(k)} > \frac{P(k|C_2)P(C_2)}{p(k)}$$

Putem renunța la termenul $p(k)$ în inegalitate. Înlocuim $P(k|C_i)$ și obținem:

$$P(k|C_i)P(C_i) = \frac{1}{\sqrt{2\pi}\sigma_i} e^{\frac{-(k-\mu_i)^2}{2\sigma_i^2}} P(C_i)$$

Vrem:

- $P(C_1|k)$ cât mai mare pentru valorile din clasa C_1 și cât mai mică pentru valorile din clasa C_2 și similar

Prag global optimal - Eroare minimă - *Minimum Error Threshold*

$$P(C_1|k) > P(C_2|k) \Leftrightarrow \frac{P(k|C_1)P(C_1)}{p(k)} > \frac{P(k|C_2)P(C_2)}{p(k)}$$

Putem renunța la termenul $p(k)$ în inegalitate. Înlocuim $P(k|C_i)$ și obținem:

$$P(k|C_i)P(C_i) = \frac{1}{\sqrt{2\pi}\sigma_i} e^{\frac{-(k-\mu_i)^2}{2\sigma_i^2}} P(C_i)$$

Vrem:

- $P(C_1|k)$ cât mai mare pentru valorile din clasa C_1 și cât mai mică pentru valorile din clasa C_2 și similar
- $P(C_2|k)$ cât mai mare pentru valorile din clasa C_2 și cât mai mică pentru valorile din clasa C_1

Prag global optimal - Eroare minimă - *Minimum Error Threshold*

$$P(k|C_i)P(C_i) = \frac{1}{\sqrt{2\pi}\sigma_i} e^{\frac{-(k-\mu_i)^2}{2\sigma_i^2}} P(C_i)$$

Funcția Logaritm este crescătoare \Rightarrow putem să o aplicăm formulei fără să schimbe inegalitatea:

$$\begin{aligned} \ln(P(k|C_i)P(C_i)) &= \ln\left(\frac{1}{\sqrt{2\pi}\sigma_i}\right) + \ln\left(e^{\frac{-(k-\mu_i)^2}{2\sigma_i^2}}\right) + \ln(P(C_i)) = \\ &= -\frac{1}{2} \left[\ln(2\pi) + \frac{(k-\mu_i)^2}{\sigma_i^2} + \ln(\sigma_i^2) - 2\ln(P(C_i)) \right] \end{aligned}$$

Prag global optimal - Eroare minimă - *Minimum Error Threshold*

$$-\frac{1}{2} \left[\ln(2\pi) + \frac{(k - \mu_i)^2}{\sigma_i^2} + \ln(\sigma_i^2) - 2\ln(P(C_i)) \right]$$

- Valoarea acestei expresii trebuie să fie cât mai mare pentru un pixel k din clasa C_i

$\epsilon_i(k)$ = măsură potențialei erori de clasificare a valorii observate k în clasa C_i .

Prag global optimal - Eroare minimă - *Minimum Error Threshold*

$$-\frac{1}{2} \left[\ln(2\pi) + \frac{(k - \mu_i)^2}{\sigma_i^2} + \ln(\sigma_i^2) - 2\ln(P(C_i)) \right]$$

- Valoarea acestei expresii trebuie să fie cât mai mare pentru un pixel k din clasa C_i .
- $\ln(2\pi)$ este constantă, deci poate fi ignorată din relația care trebuie maximizată.

$\epsilon_i(k) =$ măsură potențialei erori de clasificare a valorii observate k în clasa C_i .

Prag global optimal - Eroare minimă - *Minimum Error Threshold*

$$-\frac{1}{2} \left[\ln(2\pi) + \frac{(k - \mu_i)^2}{\sigma_i^2} + \ln(\sigma_i^2) - 2\ln(P(C_i)) \right]$$

- Valoarea acestei expresii trebuie să fie cât mai mare pentru un pixel k din clasa C_i
- $\ln(2\pi)$ este constantă, deci poate fi ignorată din relația care trebuie maximizată.
- În loc să maximizăm relația de mai sus este suficient să minimizăm:

$$\epsilon_i(k) = \frac{(k - \mu_i)^2}{\sigma_i^2} + \ln(\sigma_i^2) - 2\ln(P(C_i))$$

$\epsilon_i(k)$ = măsură potențialei erori de clasificare a valorii observate k în clasa C_i .

Prag global optimal - Eroare minimă - *Minimum Error Threshold*

Valorile $\mu_1, \mu_2, \sigma_1, \sigma_2$ sunt determinate de pragul de binarizare t , care împarte valorile de gri în clasele C_1 și C_2 .

Calitatea clasificării pe baza pragului t poate fi estimată pe baza funcției

$$e(t) = \sum_{k=0}^t p(k)\epsilon_1(k) + \sum_{k=t+1}^{255} p(k)\epsilon_2(k)$$

Din calcule matematice, formula de mai sus devine:

$$e(t) = 1 + P_1(t) \ln(\sigma_1^2(t)) + P_2(t) \ln(\sigma_2^2(t)) - 2P_1(t) \ln(P_1(t)) - 2P_2(t) \ln(P_2(t))$$

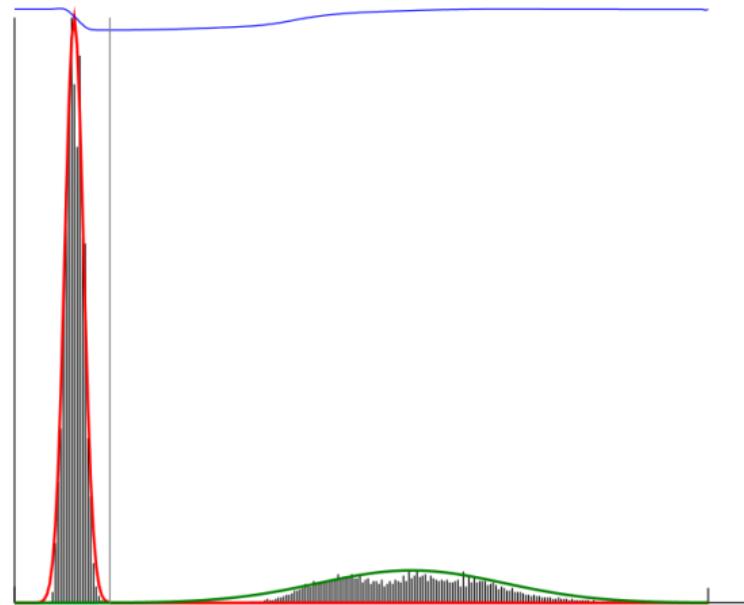
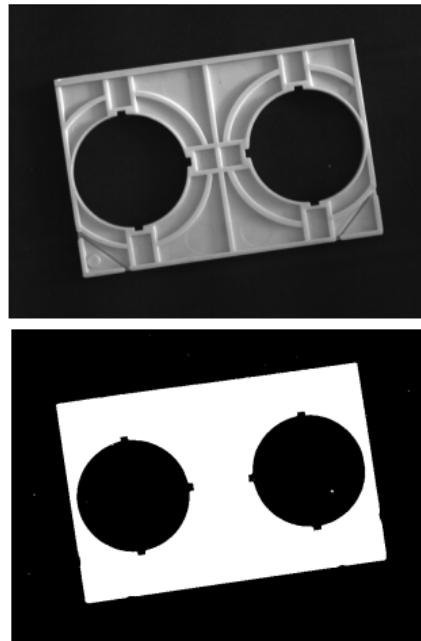
Care trebuie minimizată.

Prag global optimal - Eroare minimă - *Minimum Error Threshold*

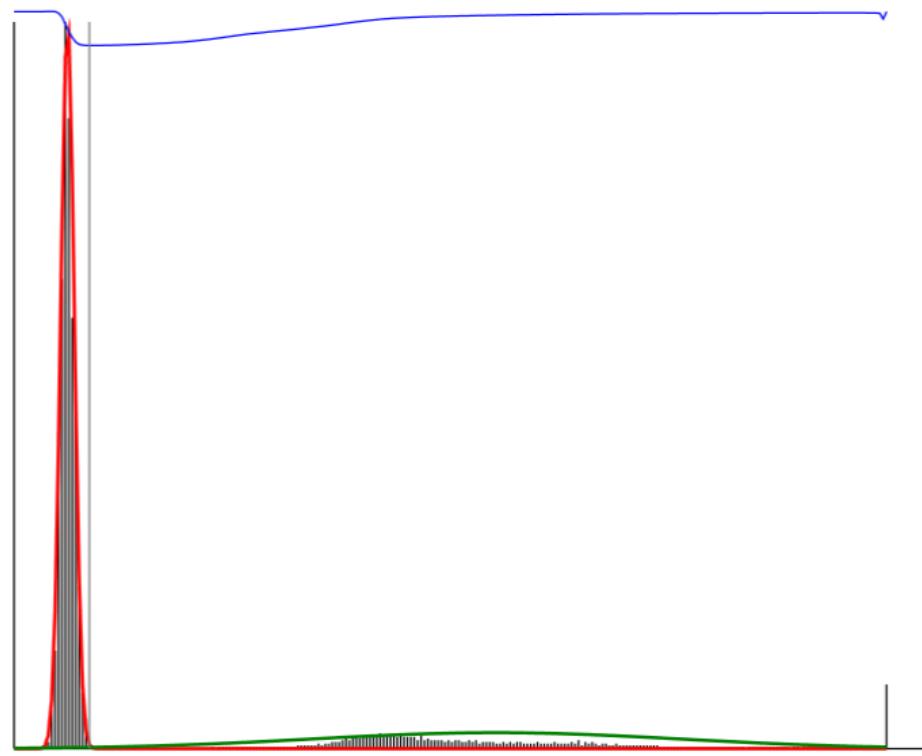
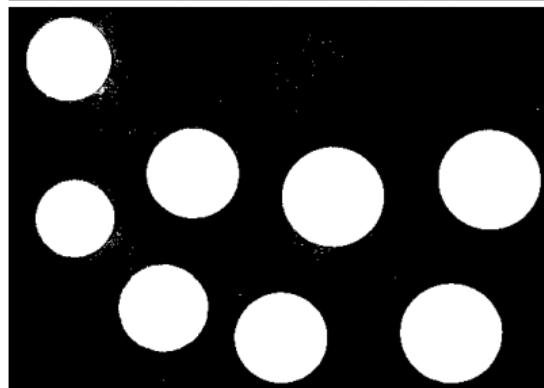
Algoritm

- ① Inițializează ϵ_{min}
- ② Pentru fiecare prag $t \in [0, 255]$
 - calculază $P_1, P_2 = 1 - P_1, \mu_1, \mu_2, \sigma_1, \sigma_2$ pe baza histogramei relative
 - calculează $\epsilon = 1 + P_1 \ln(\sigma_1^2) + P_2 \ln(\sigma_2^2) - 2P_1 \ln(P_1) - 2P_2 \ln(P_2)$
- ③ dacă $\epsilon < \epsilon_{min}$ atunci $\epsilon_{min} \leftarrow \epsilon$ și reține pragul curent
- ④ returnează pragul pentru care s-a obținut cea mai mică valoarea a lui ϵ

Prag global optimal - Eroare minimă - *Minimum Error Threshold*



Prag global optimal - Eroare minimă - *Minimum Error Threshold*



Prag global optimal - Variante

- **Histograma pixelilor de contur:** pentru determinarea pragului (optimal) în loc de toți pixelii din imagine, se utilizează doar aceia aflați pe sau în apropierea unui contur, dat prin gradient.
- **Praguri multiple:** În cazul histogramelor multi-modale se pot utiliza mai multe praguri pentru separarea zonelor de interes

Metoda Otsu - Praguri multiple

Metoda Otsu poate fi extinsă pentru un număr variabil - $(K - 1)$ - de praguri: t_1, t_2, \dots, t_{K-1} care determină K clase C_1, C_2, \dots, C_K

Variația inter-clasă devine:

$$\sigma_B^2 = \sum_{i=1}^K P_i(\mu_i - \mu)^2$$

unde C_i = clasa pixelilor din intervalul $(t_{i-1}, t_i]$

$$P_i = \sum_{k=t_{i-1}+1}^{t_i} p(k)$$

și

$$\mu_i = \frac{1}{P_i} \sum_{k=t_{i-1}+1}^{t_i} kp(k)$$

Se vor alege acele praguri $t_1 < t_2 < \dots < t_{K-1}$ care maximizează σ_B

Metoda Otsu - Praguri multiple

Exemplu Pentru 3 clase - 2 praguri t_1 și t_2

Varinația inter-clasă devine:

$$\sigma_B^2 = P_1(\mu_1 - \mu)^2 + P_2(\mu_2 - \mu)^2 + P_3(\mu_3 - \mu)^2$$

Pentru clasa C_1	Pentru clasa C_2	Pentru clasa C_3
$P_1 = \sum_{k=0}^{t_1} p(k)$	$P_2 = \sum_{k=t_1+1}^{t_2} p(k)$	$P_3 = \sum_{k=t_2+1}^{255} p(k)$
$\mu_1 = \frac{1}{P_1} \sum_{k=0}^{k_1} kp(k)$	$\mu_2 = \frac{1}{P_2} \sum_{k=k_1+1}^{k_2} kp(k)$	$\mu_3 = \frac{1}{P_3} \sum_{k=k_2+1}^{255} kp(k)$

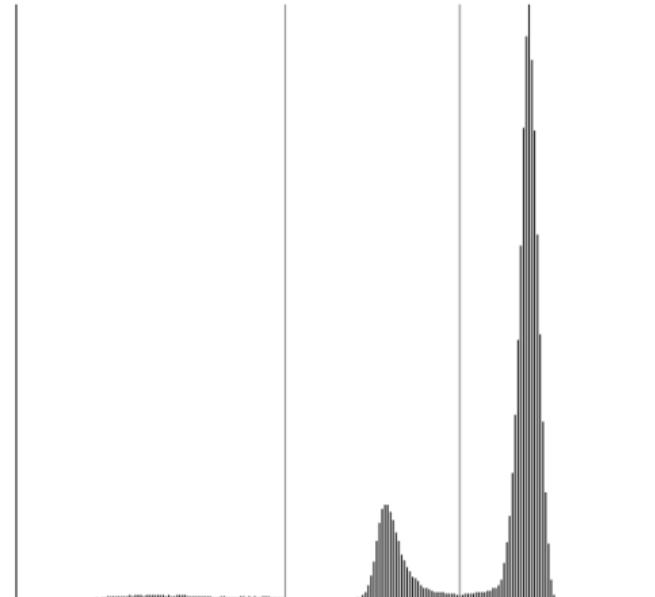
Se vor alege acele praguri $0 < k_1 < k_2 < 255$ care maximizează σ_B

Metoda Otsu - Praguri multiple

Algoritm - pentru 2 praguri

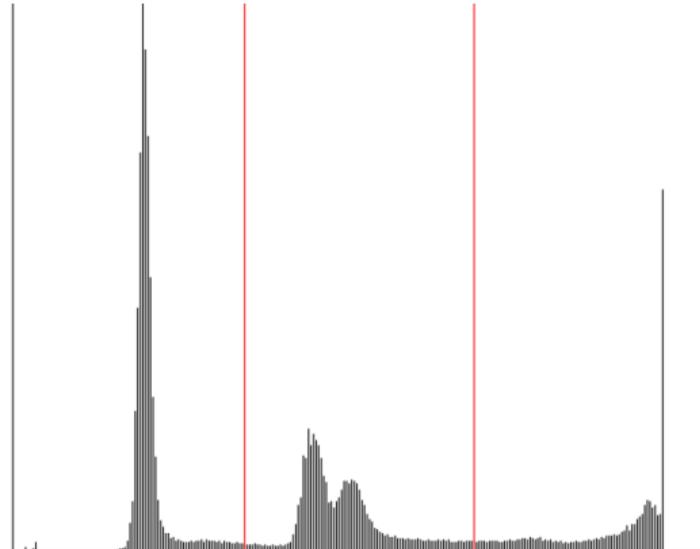
- ① se pornește cu $\sigma_B_{max} = 0$
- ② $k_1 = 1$
- ③ pentru $k_2 = k_1 + 1 \dots 254$ execută pașii
- ④ calculează $P_1, P_2, P_3, \mu_1, \mu_2, \mu_3$ și σ_B
- ⑤ dacă $\sigma_B > \sigma_B_{max}$ atunci $\sigma_B_{max} \leftarrow \sigma_B$
- ⑥ $k_1 = k_1 + 1$
- ⑦ dacă $k_1 < 253$ mergi la pasul 3

Metoda Otsu - Praguri multiple - Exemplu



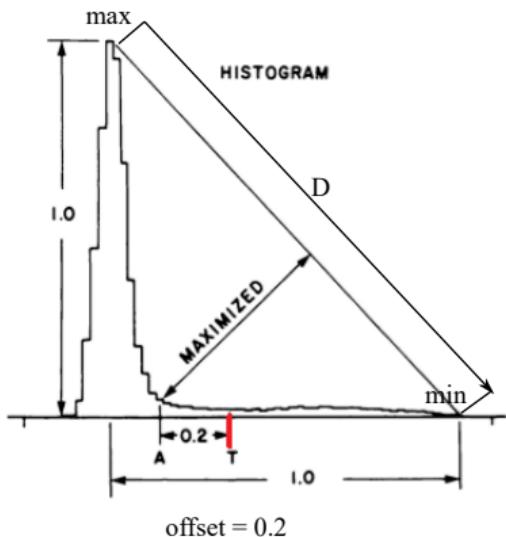
$$t_1 = 97, \quad t_2 = 160$$

Metoda Otsu - Praguri multiple - Exemplu



$$t_1 = 91, \quad t_2 = 181$$

Extragerea fundalului - Metoda triunghiului



- Se caută valoarea maximă din histogramă
- Se caută o valoare minimă cât mai departe de poziția maximului (poate fi de o parte sau de alta a maximului).
- Se consideră latura D care "unește" maximul de minim
- Se alege ca prag t acea valoare k pentru care se obține cea mai mare distanță dintre $H[k]$ și $D + \text{offset}$

Prag adaptabil

- se utilizează în cazul iluminării neuniforme.
- calculează pentru diferite regiuni praguri diferite prin
 - partiționarea imaginii în subimagini disjuncte și calculul unui prag pentru fiecare subimagine
 - calcularea pentru fiecare pixel (x, y) a unui prag pe baza proprietăților statistice ale unei vecinătăți.

Prag adaptabil pentru fiecare pixel - variante

- ① $t_{xy} = a\sigma_{xy} + b\mu_{xy}$, a, b constante pozitive apropriate de 1

Dacă notăm cu S_{xy} o vecinătate centrată în pixelul (x, y) atunci:

- t_{xy} = pragul pentru pixelul (x, y) ,
- μ_{xy} - media pixelilor din S_{xy}
- \min_{xy}, \max_{xy} , valoarea minimă respectiv maximă din S_{xy}
- σ_{xy} - varianța pixelilor din S_{xy}

Prag adaptabil pentru fiecare pixel - variante

1 $t_{xy} = a\sigma_{xy} + b\mu_{xy}$, a, b constante pozitive apropriate de 1

2 $t_{xy} = a \max_{xy} + (1 - a) \min_{xy}$,
Metoda lui *Bernsen* propune
 $a = (1 - a) = 1/2$

Dacă notăm cu S_{xy} o vecinătate centrată în pixelul (x, y) atunci:

- t_{xy} = pragul pentru pixelul (x, y) ,
- μ_{xy} - media pixelilor din S_{xy}
- \min_{xy}, \max_{xy} , valoarea minimă respectiv maximă din S_{xy}
- σ_{xy} - varianța pixelilor din S_{xy}

Prag adaptabil pentru fiecare pixel - variante

- ① $t_{xy} = a\sigma_{xy} + b\mu_{xy}$, a, b constante pozitive apropriate de 1
- ② $t_{xy} = a \max_{xy} + (1 - a) \min_{xy}$,
Metoda lui *Bernsen* propune
 $a = (1 - a) = 1/2$
- ③ $t_{xy} = b\mu_{xy}$

Dacă notăm cu S_{xy} o vecinătate centrată în pixelul (x, y) atunci:

- t_{xy} = pragul pentru pixelul (x, y) ,
- μ_{xy} - media pixelilor din S_{xy}
- \min_{xy}, \max_{xy} , valoarea minimă respectiv maximă din S_{xy}
- σ_{xy} - varianța pixelilor din S_{xy}

Prag adaptabil pentru fiecare pixel - Exemplu

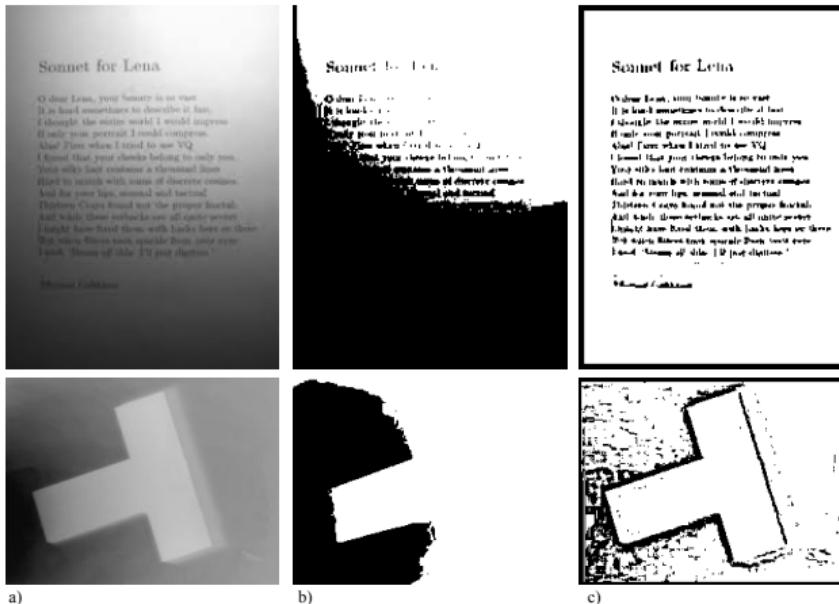


Figure: a) Imagini de test cu iluminare neuniformă; b) Imaginele binarizate cu metoda Otsu; c) Imaginele binarizate utilizând prag adaptabil, cu $a = 0.5$, $b = 0.9$ pentru prima imagine și $a = 0.4$, $b = 0.98$ pentru a doua imagine.

Prag adaptabil pentru fiecare pixel - Exemplu

It seems to have some difficult and ambiguous situations. The situation in the darker regions is insufficient and unnatural, and the bright artifacts are introduced which caused blurring. In the bright regions, it was not enough brightness, thereby losing information in highlight regions to have "overexposure" artifact and enlarge the block effect. However, if selected processes more details with high visual quality, the image quality will be improved. As shown in Figure 10(a), the histogram shows the characteristics of the image visibility should be preserved before after LFR. Figure 10(b) shows the histogram of the image visibility after LFR, the better histogram range, and the sky region distribution changed, for more display resulted, the proposed method restore local details and global contrast. Especially, the histogram of the image visibility after our proposed method can identify the block effect fully. Besides the subjective quality values of effective standards are listed in Table 1.

It seems to have some difficulties in reading the images in the darker regions. The histogram in the darker regions is insufficient and unnatural and halo artifacts are introduced which caused biased gray levels. In addition, it is only a bright region that is highlighted in the original image, so the image has "wing-out" artifact and enlarge the block effect. However, method possesses more details with high visual quality than the histogram matching and global histogram. Just as the histogram matching and global histogram, the histogram in the brighter regions shows the characteristics of the image. The visibility should be preserved before and after LR. Figure 1(d) shows that the shadow in the original image shape is changed for more degraded images in form and the sky region is also changed for more degraded images in form. The method can restore local detail and global contrast. Especially for the images with low visual quality, our method can not only apply the block effect better. Besides the subjective evaluation, values of objective standards are listed in Table 1.

3.1. Faint highlights In this section, we discuss the highlights in the original image and the results obtained by the proposed method. Figure 7 shows the original image and the results obtained by the proposed method. The original image contains a large number of highlights, which are mostly located in the darker regions. The highlights in the darker regions are insufficient and uncertain, and halo artifacts are introduced which caused bigger, more intense highlights in highlight regions to have the “ringing” effect and enlarge the black effect. However, the proposed method uses details with high visual quality to identify lighting highlights, and the shadow region is enlarged, for more detailed results, the proposed method makes the highlights appear in the original image. Figure 8 shows the histogram shows the characteristics of the image visibility should be preserved before and after LRF. Figure 8(a) shows the histogram in the original image, while Fig. 8(b) the better histogram, indicating that the shadow region is enlarged, for more detailed results, the proposed method makes the highlights appear in the original image. Figure 8(c) shows the histogram shows the characteristics of the image visibility should be preserved before and after LRF. Figure 8(d) shows that the shadow region in the original image should be enlarged, for more detailed results, the proposed method makes the highlights appear in the original image. Figure 8(e) shows the histogram shows the characteristics of the image visibility should be preserved before and after LRF. Figure 8(f) shows that the shadow region in the original image should be enlarged, for more detailed results, the proposed method makes the highlights appear in the original image.

Figure: Thresholding adaptiv utilizând formula $t_{xy} = b\mu_{xy}$ cu $b = 0.8$.



Calcul rapid al mediei într-o vecinătate - Imaginea integrală

Problemă: pentru fiecare pixel (x, y) al unei imaginei $M \times N$ trebuie calculată media pixelilor într-o vecinătate $dim \times dim$.

Ce complexitate avem?

Calcul rapid al mediei într-o vecinătate - Imaginea integrală

Problema: pentru fiecare pixel (x, y) al unei imaginii $M \times N$ trebuie calculată media pixelilor într-o vecinătate $dim \times dim$.

Ce complexitate avem?

$$\text{Complexitatea} = O(M * N * dim^2)$$

Calcul rapid al mediei într-o vecinătate - Imaginea integrală

Imaginea integrală: Considerăm imaginea $f(x, y)$, atunci $I(x, y)$ este dată prin:

$$I(x, y) = \sum_{s=0}^x \sum_{t=0}^y f(x, y)$$

1	2	2	4	3	2	4
4	4	1	2	3	4	4
2	2	2	3	1	1	2
1	2	1	1	3	3	4
3	2	2	3	3	1	2
3	3	3	4	3	3	4
1	2	1	1	2	2	4

Imaginea originală

1	3	5	9	12	14	18
5	11	14	20	26	32	40
7	15	20	29	36	43	53
8	18	24	34	44	54	68
11	23	31	44	57	68	84
14	29	40	57	73	87	107
15	32	44	62	80	96	120

Imaginea integrală

Imaginea integrală - calcul bazat pe formule de recurență

$$I(x, y) = \begin{cases} f(0, 0), & \text{pentru } x = 0, y = 0 \\ I(0, y-1) + f(0, y), & \text{pentru } x = 0 \\ I(x-1, 0) + f(x, 0), & \text{pentru } y = 0 \\ I(x-1, y) + I(x, y-1) - I(x-1, y-1) + f(x, y), & \text{altfel} \end{cases}$$

1	2	2	4	3	2	4
4	4	1	2	3	4	4
2	2	2	3	1	1	2
1	2	1	1	3	3	4
3	2	2	3	3	1	2
3	3	3	4	3	3	4
1	2	1	1	2	2	4

Imaginea originală

1	3	5	9	12	14	18
5	11	14	20	26	32	40
7	15	20	29	36	43	53
8	18	24	34	44	54	68
11	23	31	44	57	68	84
14	29	40	57	73	87	107
15	32	44	62	80	96	120

Imaginea integrală

Calculul sumei unei vecinătăți cu imaginea integrală - complexitate $M \times N$

$$sum(x_0, y_0, x_1, y_1) = \begin{cases} A, & \text{pentru } x_0 = 0, y_0 = 0 \\ A - B, & \text{pentru } y_0 = 0 \\ A - D, & \text{pentru } x_0 = 0 \\ A + C - B - D, & \text{altfel} \end{cases}$$

unde $A = I(x_1, y_1)$, $B = I(x_0 - 1, y_1)$, $C = I(x_0 - 1, y_0 - 1)$, $D = I(x_1, y_0 - 1)$.

Imaginea originală

	x ₀		x ₁				
y ₀	1	2	2	4	3	2	4
y ₁	4	4	1	2	3	4	4
	2	2	2	3	1	1	2
	1	2	1	1	3	3	4
	3	2	2	3	3	1	2
	3	3	3	4	3	3	4
	1	2	1	1	2	2	4

Imaginea integrală

	x ₀	x ₁					
y ₀	1	3	5	9	12	14	18
y ₁	5	11	14	20	26	32	40
	7	15	20	29	36	43	53
	8	18	24	34	44	54	68
	11	23	31	44	57	68	84
	14	29	40	57	73	87	107
	15	32	44	62	80	96	120

	x ₀	x ₁				
y ₀	C					
y ₁	B					

Binarizarea imaginilor color

Problematică:

- Obținerea unui obiect / zone de interes pornind de la o imagine color
- Depinde de cerința dată

Binarizarea imaginilor color

Obținerea unui obiect de interes pe un fundal relativ uniform:



Grayscale



Otsu

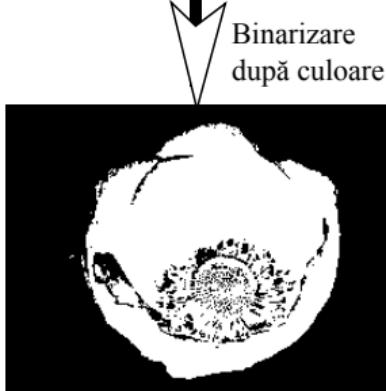


Binarizarea imaginilor color

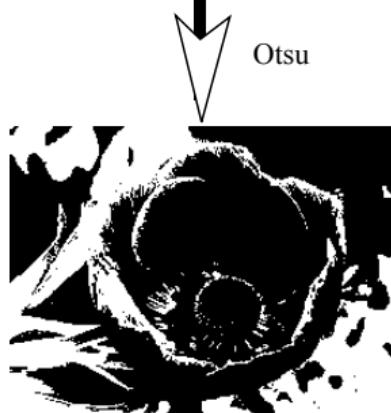
Binarizarea pe baza culorii = identificarea regiunilor / obiectelor cu o anumită culoare



Grayscale

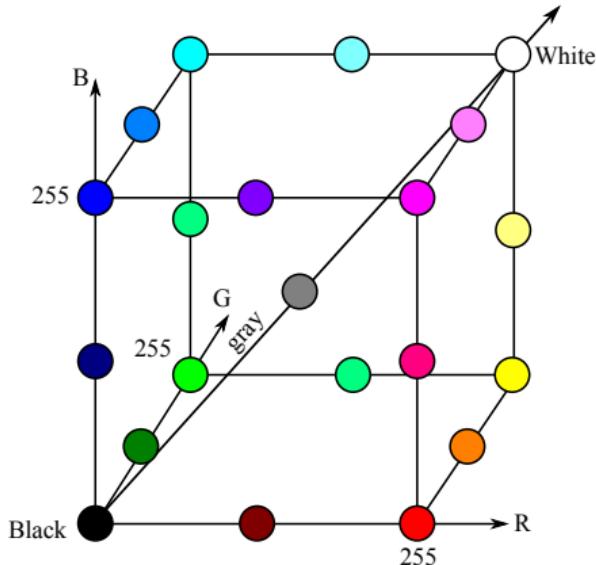


Binarizare
după culoare



Otsu

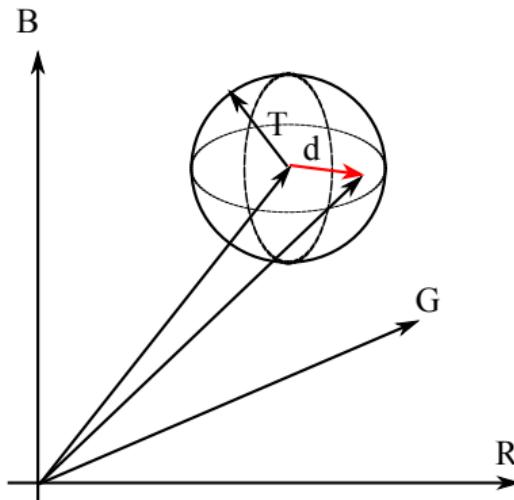
Binarizarea imaginilor color



Fiecare culoare: un vector (R, G, B)

Ce înseamnă că două culori (R_1, G_1, B_1) și (R_2, G_2, B_2) sunt asemănătoare?

Culori similare - binarizarea imaginilor color - abordare 3D



Culori asemănătoare

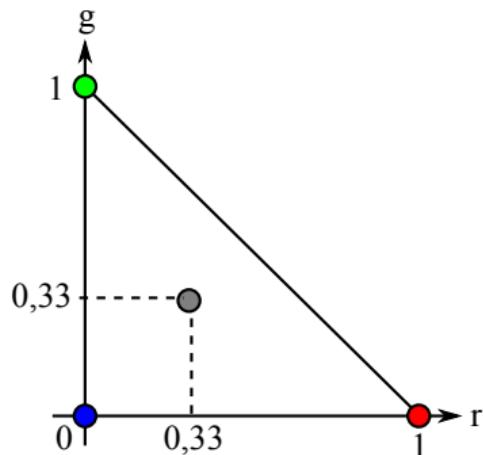
- $c(x, y) = (R(x, y), G(x, y), B(x, y))$ - culoarea pixelului (x, y) ;
- $c_0 = (R_0, G_0, B_0)$ - culoarea de referință;
- t - pragul care definește asemănărea.

$c(x, y)$ este asemănătoare cu c_0 dacă:

$$d := \sqrt{(R(x, y) - R_0)^2 + (G(x, y) - G_0)^2 + (B(x, y) - B_0)^2} \leq t$$

Diagrama cromatică

Reprezentarea 2D a cubului de culoare normalizat



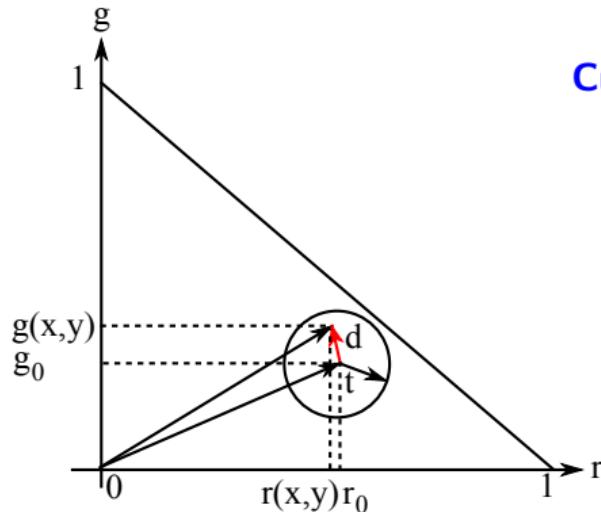
Pentru fiecare pixel (x, y) de culoare $(R(x, y), G(x, y), B(x, y))$:

$$r(x, y) = \frac{R(x, y)}{R(x, y) + G(x, y) + B(x, y)}$$

$$g(x, y) = \frac{G(x, y)}{R(x, y) + G(x, y) + B(x, y)}$$

$$\begin{aligned} b(x, y) &= \frac{B(x, y)}{R(x, y) + G(x, y) + B(x, y)} = \\ &= 1 - r(x, y) - g(x, y) \end{aligned}$$

Culori similare - binarizarea imaginilor color - abordare 2D



Culori asemănătoare utilizând tabloul cromatic

- $r(x, y), g(x, y)$ - valorile normalize ale componentelor R și G ale pixelui (x, y) ;
- (r_0, g_0) - valorile normalize ale componentelor R și G pentru culoarea de referință;
- t - pragul care definește asemănărea.

$c(x, y)$ este asemănătoare cu c_0 dacă:

$$d := \sqrt{(r(x, y) - r_0)^2 + (g(x, y) - g_0)^2} \leq t$$

Exemple de binarizare



Imagine originală

Imagine binarizată 3D
 $T=30$

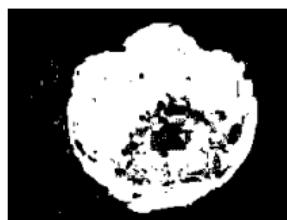
Imagine binarizată 2D
 $t=0,1$

Binarizarea în spațiul de culoare HSV

Alternativă - binarizarea în spațiul de culoare HSV

- Se transformă imaginea din spațiul RGB în spațiul HSV
- Se realizează binarizarea pe canalul H în raport cu culoarea de referință

Binarizarea în spațiul de culoare HSV



Rezumat

- Binarizare.
- Thresholding cu prag global automat
 - Praguri globale simple
 - Intermeans
 - Otsu
 - Minimum Error threshold
- Thresholding cu praguri multiple - ex Otsu
- Thresholding cu prag adaptabil
- Binarizarea imaginilor color
 - Binarizare3D
 - Diagrama cromatică
 - Binarizare2D
 - Binarizarea in HSV