

## Restaurarea frontierelor unei imagini: abordare genetica

Prof.dr. Luminita STATE, Universitatea Pitesti  
 Conf.dr. Doina FUSARU, Universitatea Spiru Haret Bucuresti  
 Asist. Catalina-Lucia COCIANU  
 Catedra de Informatica Economica, A.S.E. Bucuresti

*Lucrarea prezinta un algoritm de restaurare a frontierelor unei imagini pe baza puterii spectrale a unui pixel virtual frontiera, calculata la nivelul unui bloc al imaginii de intrare. Algoritmul cuprinde doua componente: una de natura statistica, utilizata pentru a decide daca pixelul procesat la momentul curent este de tip frontiera si cea de-a doua de natura genetica, în care este realizata modificarea nivelului de gri al pixelului curent printr-un proces de selectie repetata bazat pe puterea spectrala corespunzatoare punctului prelucrat.*

**Cuvinte cheie:** *putere spectrala, nivel de gri, algoritm genetic, selectie, mutatie, crossover, imagine-bloc.*

### 1. Algoritmi genetici; consideratii generale

Principiile genetice si de natura evolutionista dezvoltate în ultimii ani pentru rezolvarea unor clase largi de probleme complexe s-au impus datorita performantelor lor legate atât de timpul redus de executie implicat, cât si datorita posibilitatilor de determinare a unor solutii apropiate de cele optimale. Multe dintre problemele tratate genetic sunt probleme de optimizare; aplicatii ale acestora au condus la rezolvarea unor tipuri diverse de probleme, printre care: probleme de rutare, control adaptiv, jocuri, modelare de tip cognitiv, probleme de transport, de control optimal etc. În cele ce urmeaza, vor fi tratate modele de rezolvare ale unor probleme de optim, de altfel cele în care programarea genetica are cele mai bune rezultate.

Algoritmii genetici apartin clasei algoritmilor probabilistici; ei difera de algoritmii aleatori prin combinarea elementelor de cautare directa si stochastica. De aceea algoritmii genetici sunt mai robusti decât cei de cautare directa. În plus, programarea genetica permite mentinerea unei întregi populatii de solutii potentiale, fata de metodele directe, care, uzual, proceseaza la

un moment de timp un singur punct din spatiul de cautare. Evolutia cautare directa-cautare stochastica cuprinde doua tehnici care reduc din dezavantajele cautarilor directe, si anume metode de tip "hillclimbing" si "simulated annealing".

Metodele de tip "hillclimbing" utilizeaza o tehnica de iterativitate îmbunatatita. Aceasta se aplica unui singur punct din spatiul de cautare. La o iteratie este selectat un nou punct aflat într-o vecinatate a punctului curent procesat. Daca acest punct determina o valoare mai buna (din punct de vedere al criteriului de optim considerat) pentru functia obiectiv, el devine punct curent. În caz contrar, este selectata o alta vecinatate a punctului curent, procesul desfasurându-se ulterior similar. Algoritmul se încheie când nici un punct vecin celui curent nu aduce îmbunatatiri valorilor functiei obiectiv. Metodele de acest tip conduc de obicei la valori de optim local, depinzând de punctul de start. În plus, nu se pot furniza informatii referitoare la eroarea relativa a solutiei calculate. Pentru a creste performantele unor astfel de modele, acestea se utilizeaza pentru un numar mare de punct de start.

Metodele de tip "simulated annealing" elimina o mare parte din dezavantajele

algoritmilor "hillclimbing", în sensul ca soluțiile nu depind de punctul de start și sunt de obicei apropiate de punctul de optim global. Pentru aceasta, este considerată o probabilitate de acceptare a punctului selectat drept următor punct curent, egală cu 1 dacă noul punct furnizează o valoare mai bună pentru funcția obiectiv considerată. În unele situații, probabilitatea de a accepta un nou punct este o funcție cu valori corespunzătoare funcției obiectiv pentru punctul curent și noul punct selectat. De asemenea, fata de tehnica "hillclimbing", este considerat un parametru de tip temperatura sistemului, care influențează probabilitatea de acceptare a unui nou punct ca punct curent: cu cât acest parametru este mai scăzut, cu atât șansele de acceptare sunt mai mici. Pe parcursul executiei algoritmului, temperatura sistemului scade; algoritmul se încheie pentru o temperatura mică, pentru care nu se mai accepta nici o modificare a soluției (probabilitatea de acceptare a unui nou punct este 0).

Algoritmii genetici reprezintă o altă abordare a problemei de căutare a unei soluții optimale. La fiecare iteratie  $t$ , algoritmul menține o populație de soluții potențiale, numite cromozomi,  $P(t) = \{x_1^t, x_2^t, \dots, x_n^t\}$ .

Cromozomii sunt reprezentări binare ale soluțiilor considerate la momentul respectiv. Fiecare cromozom  $x_i^t, t = \overline{1, n}$  este evaluat în scopul măsurării performanțelor sale din punct de vedere al funcției obiectiv considerate, aplicate pentru soluția a cărei reprezentare o constituie. La iteratia  $t+1$  este selectată o nouă populație, pe baza calității indivizilor populației de la momentul anterior. O parte a membrilor acestei populații suferă modificări de tip mutație și crossover (încrucisare) și determină noi indivizi.

Operația crossover combină caracteristicile a doi cromozomi părinți și creează doi indivizi noi, copii ai cromozomilor ce intra în operație, prin interschimbarea unor seg-

mente ale reprezentării binare corespunzătoare. De exemplu, dacă părinții sunt reprezentați prin șiruri binare de dimensiune 5  $(a_1, b_1, c_1, d_1, e_1)$ , respectiv  $(a_2, b_2, c_2, d_2, e_2)$ , atunci combinarea cromozomilor după a doua genă (a doua poziție de la stânga la dreapta în vectorul de reprezentare) determină apariția copiilor  $(a_1, b_1, c_2, d_2, e_1)$ , respectiv  $(a_2, b_2, c_1, d_1, e_1)$ . Aplicabilitatea intuitivă a unei operații crossover o constituie schimbul de informații între indivizii populației soluțiilor potențiale.

Operația mutație afectează arbitrar una sau mai multe gene ale unui cromozom selectat, printr-o modificare aleatoare cu probabilitatea de execuție egală cu rata prestabilă a mutației. Biții corespunzători pozițiilor selectate astfel pentru mutație sunt modificați (din 0 în 1, respectiv din 1 în 0). Intuitiv, un operator mutație are rolul de a introduce o variabilitate suplimentară în cadrul indivizilor populației.

Un algoritm genetic, deci, trebuie să cuprindă următoarele componente:

1. o reprezentare genetică (la nivel de șiruri de biți) a soluțiilor potențiale ale problemei;
2. o modalitate de creare a populației inițiale;
3. o funcție de evaluare care furnizează calitatea unui cromozom din cadrul populației;
4. operatori genetici pentru modificarea structurii cromozomiale;
5. valori ale parametrilor utilizați în algoritm: dimensiunea populației, probabilitatea de aplicare a operatorilor genetici etc.

## **2. Algoritm genetic pentru restaurarea frontierelor unei imagini dirijat de puterea spectrală a unui pixel calculată la nivel de bloc**

Algoritmul genetic propus se bazează pe o funcție de evaluare sugerată de o măsură globală de informație, și anume puterea spectrală calculată în punctele frontiera.

Problema considerata pentru rezolvare este maximizarea puterii spectrale a unui punct de tip contur.

Fie o imagine  $X \in M_{M \times N}(LS)$ , în care  $LS$  reprezintă mulțimea nivelelor de gri ale imaginii (considerată în continuare alb-

$$\begin{aligned} P_X(k,l) &= \frac{I}{(MN)^2} \left| \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} X_{u,v} e^{-2\mathbf{p} \left( \frac{uk}{M} + \frac{vl}{N} \right)} \right|^2 = \\ &= \frac{I}{(MN)^2} \left| \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} X_{u,v} \cos \left( -2\mathbf{p} \left( \frac{uk}{M} + \frac{vl}{N} \right) \right) + i \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} X_{u,v} \sin \left( -2\mathbf{p} \left( \frac{uk}{M} + \frac{vl}{N} \right) \right) \right|^2 = \\ &= \frac{I}{(MN)^2} \left[ \left( \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} X_{u,v} \cos \left( 2\mathbf{p} \left( \frac{uk}{M} + \frac{vl}{N} \right) \right) \right)^2 + \left( \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} X_{u,v} \sin \left( 2\mathbf{p} \left( \frac{uk}{M} + \frac{vl}{N} \right) \right) \right)^2 \right] \end{aligned}$$

Relatia definește puterea spectrală drept amplitudinea transformatei Fourier discrete bidimensionale.

Puterea spectrală corespunzătoare unui pixel al imaginii este redusă în cazul unui pixel fond și este mare în cazul unui pixel frontieră sau zgomet. Utilizând o funcție de evaluare pornind de la această remarcă, rezulta că algoritmul genetic corespunzător nu prezintă stabilitate la zgomet. În schimb, datorită măsurii globale considerate pentru evaluare, puterea de subliniere a conturilor obiectelor unei imagini este apreciabilă chiar și în contextul unei imagini cu contrast și luminozitate foarte reduse.

Pentru un pixel  $(k, l)$  al imaginii de intrare, algoritmul propus determină dacă este pixel frontieră și, în caz afirmativ, declanșează un mecanism genetic pentru maximizarea puterii spectrale a acestuia. Dacă evaluarea se face la nivelul întregii imagini, iar aceasta are o rezoluție foarte fină (matricea de reprezentare a imaginii este de dimensiuni mari), atunci, chiar și în condițiile aplicării unei metode FFT (Fast Fourier Transform), algoritmul necesită un timp foarte mare de execuție. În scopul eliminării acestui inconvenient, evaluarea puterii spectrale a unui pixel se face la

negru), iar  $M$  și  $N$  reprezintă numărul de linii, respectiv de coloane ale reprezentării matriceale a lui  $X$ . Puterea spectrală a pixelului  $(k, l)$  din imaginea considerată este calculată conform relației [19]:

nivelul unui bloc al imaginii care conține pixelul respectiv și care are dimensiuni mici.

Fie  $(k, l)$  un pixel aparținând unui bloc  $B$  al imaginii de intrare,  $X$ . Pentru a decide dacă pixelul este virtual frontieră, se procedează astfel:

- a) 1. Este calculată valoarea medie a nivelelor de gri corespunzătoare pixelilor blocului  $B$ ; fie aceasta  $m_B$ ;
2. Dacă  $|m_B - X_{k,l}| > val$ , atunci pixelul  $(k, l)$  este de tip frontieră. Valoarea  $val$  depinde invers proporțional de gradul de deteriorare (din punct de vedere al contrastului și luminozității) a imaginii.

sau

- b) 1. Pentru fiecare pixel  $(i, j)$  al blocului  $B$ , se calculează puterea spectrală corespunzătoare  $P_B(i, j)$ , relativ la subimaginea  $B$ , respectiv media acesteia, notată  $m_{P_B}$ ;
2. Dacă  $P_B(k, l) - m_{P_B}$  este suficient de mare (depinzând, ca și în cazul primei variante, de contrastul și luminozitatea imaginii de intrare), atunci  $(k, l)$  este pixel frontieră.

Prima modalitate de decizie, prezentată la punctul a), are calitatea că necesită un timp

de calcul redus. Cea de-a doua varianta, desi presupune un volum mult mai mare de calcule, induce o detectie mai fina. Se poate proceda, însa, la o combinatie a celor doua metode. Considerând  $(k, l)$  un pixel al blocului  $B$ , atunci:

1. Se calculeaza valoarea medie a nivelelor de gri corespunzatoare pixelilor blocului
2. Daca  $|m_B - X_{k,l}| > val_1$ , atunci:
  - 2.1. pentru fiecare pixel  $(i, j)$  al blocului  $B$ , calculeaza  $P_B(i, j)$ , respectiv  $m_{P_B}$
  - 2.2. daca  $P_B(k, l) - m_{P_B} > val_2$ , atunci  $(k, l)$  este pixel frontiera.

Fie  $B$  bloc de dimensiuni  $n \times n$  al imaginii intrare si  $(k, l)$  un pixel frontiera aparținând lui  $B$ . Algoritmul genetic care prelucreaza  $(k, l)$  în vederea maximizarii puterii spectrale  $P_B(k, l)$  este descris în continuare.

**Reprezentarea cromozomiala** a unui pixel este data de reprezentarea binara a numarului natural care indica intrarea respectiva din paleta de culori. Lungimea reprezentarii este de 8 biti, în cazul imaginilor cu 256 nivele de gri de la alb la negru, respectiv de 16 biti, pentru palete cu mai multe culori.

**Populatia initiala.** Pentru îndepartarea efectelor de tip frontiere cu nivel de gri scazut pentru obiecte cu nivel de gri ridicat, respectiv frontiere cu nivel de gri ridicat pentru obiecte cu nivel de gri mic, se procedeaza astfel:

- a) daca  $B(k, l) > med_B$ , atunci  $pop[i] = B(k, l) + 2(i-1)$ ,  $i=1, \dots, dim$ ;
- b) altfel,  $pop[i] = B(k, l) - 2(i-1)$ ,  $i=1, \dots, dim$ .

Sunt generate astfel multimi ale valorilor posibile pentru nivelul de gri  $B(k, l)$  corespunzatoare pixelului  $(k, l)$ .

**Funcția de evaluare.** Pentru fiecare  $m \in \{pop[1], pop[2], \dots, pop[dim]\}$ , evaluarea se face conform relatiei

$$eval(m) = \frac{1}{n^2} \left| \sum_{u=0}^{n-1} \sum_{v=0}^{n-1} B'_{u,v} e^{-2pi \left( \frac{uk+vl}{n} \right)} \right|^2,$$

în care  $B'_{u,v} = \begin{cases} B_{u,v}, & u \neq k \text{ si } v \neq l \\ m. & \text{altfel} \end{cases}$ , pen-

tru  $\forall u \in \{0, 2, \dots, n-1\}, \forall v \in \{0, 2, \dots, n-1\}$ .

**Selectia noii populatii.** Pentru populatia curenta  $pop_t$  se calculeaza:

1. performanta data de functia de evaluare  $eval(pop_t[i]), i=1, \dots, dim$ ;
2. performanta întregii populatii

$$E = \sum_{i=1}^{dim} eval(v_i), i=1, \dots, dim;$$

3. probabilitatea de selectie  $p_i = \frac{eval(v_i)}{E}$

si probabilitatea cumulata  $q_i = \sum_{j=1}^i p_j$ ,  $i=1, \dots, dim$ ;

4. se genereaza aleator un numar  $r \in [0, 1], i=1, \dots, dim$ ;
5. daca  $r < q_1$ , este selectat  $pop_t[1]$ ; altfel, este selectat cromozomul  $pop_t[i]$ , cu proprietatea  $q_{i-1} < r < q_i$ .

**Modificarea populatiei selectate** are loc prin aplicarea operatorilor crossover si de tip mutatie, astfel.

*Aplicarea operatorului crossover*

- a) Pentru  $i=1, \dots, dim$ 
  1. genereaza aleator  $r \in [0, 1]$ ;
  2. daca  $r < p_c$  selecteaza  $pop_t[i]$  pentru crossover
- b) Operatorul crossover este cel standard: pentru o dimensiune  $d$  generata între 1 si 7, în cazul reprezentarii pe un octet, respectiv între 1 si 15, pentru reprezentarea pe 2 octeti:
  1. selecteaza aleator doi cromozomi dintre cei determinati la a);
  2. realizeaza crossover.

*Aplicarea operatorului de tip mutatie*

Algoritmul presupune mutatii cu probabilitatea  $p_m$  în general mica. Pentru problema considerata, o mutatie revine la:

Pentru  $i=1, \dots, dim$

1. genereaza aleator  $r \in [0, 1]$ ;
2. daca  $r < p_m$  selecteaza  $pop_t[i]$  pentru mutatie, astfel:

2.1. genereaza aleator  $d \in \{1, \dots, 7\}$ , respectiv  $d \in \{1, \dots, 15\}$ ;

2.2. efectueaza schimbarea celui de-al  $d$ -lea bit al lui  $pop_i[i]$ .

**Conditia de oprire.** Întreruperea calculului inițiat de algoritmul genetic are loc când nu se mai realizează îmbunătățiri semnificative asupra puterii spectrale corespunzătoare pixelului considerat.

Pe baza acestor considerente, algoritmul pentru sublinierea conturilor obiectelor prezente într-o imagine de intrare este:

**Intrare**  $X$ , imagine de dimensiuni  $M \times N$ .

Pentru toate blocurile  $B$  de dimensiuni  $n \times n$ , respectiv  $n \times (M - kn)$ ,  $0 \leq M - kn < n$  și  $(N - kn) \times n$ ,  $0 \leq N - kn < n$  (date printr-o partiție a imaginii  $X$ ). Pentru toți  $(k, l)$  pixeli din  $B$ :

**Pas 1** Determina dacă pixelul  $(k, l)$  este frontieră. Dacă nu, consideră următorul pixel;

**Pas 2 (algoritmul genetic)**

1.1. Construiește populația inițială  $pop_0$ ;

1.2. Evaluează populația inițială

2.3. Cât timp  $P_{B(k,l)}^{curent} - P_{B(k,l)}^{curent-1} > \nu$  executa

2.4-2.6

2.4. Selectează noua populație

2.5. Modifică noua populație

2.6. Evaluează noua populație

**Iesire**  $X'$ , imaginea obținută în urma procesului de subliniere a frontierelor

### 3. Experimente și concluzii

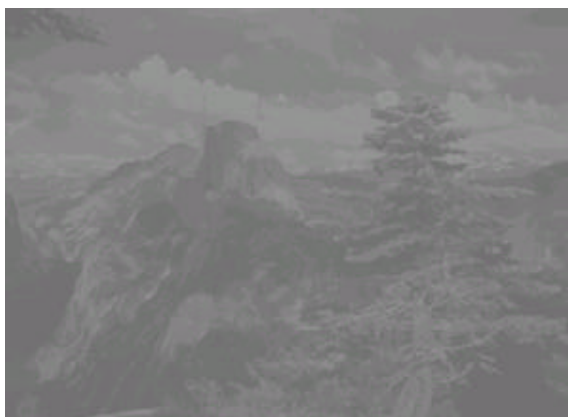
Au fost efectuate experimente pentru imagini cu 256 nivele de gri de la alb la negru, cu contrast și luminozitate foarte reduse; implementarea algoritmului descris conduce la imagini superioare din punct de vedere al clarității conturilor obiectelor prezente în imagine. Dimensiunea unui bloc variază între 3 și 7, ca atare timpul de calcul este mic. Probabilitățile de crossover și mutație sunt în jurul valorilor de 0.25, respectiv 0.15. Valorile medii ale nivelelor de gri variază între 5 și 7 (funcție de dimensiunile blocurilor considerate și de calitatea slabă a imaginii intrare). Sunt prezentate în continuare rezultatele furnizate de acest algoritm în cazul a trei intrări, în condițiile specificate anterior.



Imaginea de intrare



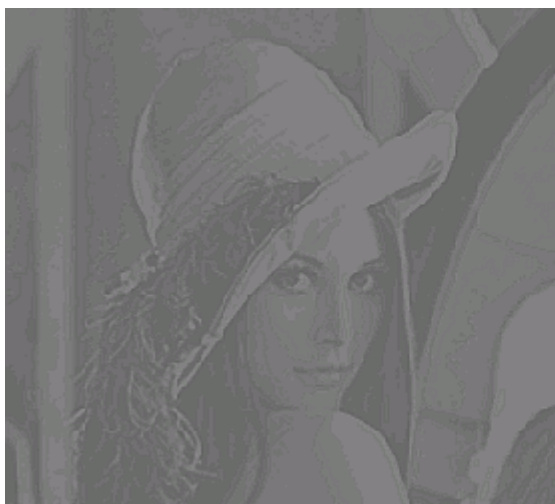
Rezultatul aplicării algoritmului de subliniere a conturilor



Imaginea de intrare



Rezultatul aplicarii algoritmului de subliniere a conturilor



Imaginea de intrare



Rezultatul aplicarii algoritmului de subliniere a conturilor

### Bibliografie

- [1] **Andrews H., Hunt B.,** *Digital Image Restoration*, Prentice-Hall, Englewood Cliffs, 1977
- [2] **Anderson T.W.,** *An Introduction to Multivariate Statistical Analysis*, John Wiley & Sons, 1958
- [3] **Brockwell P., Richard A.,** *Time Series: Theory and Methods*, Springer Verlag 1985
- [4] **Chellappa R., Kashyap R.L.,** *Digital Image Restoration Using Spatial Interaction Models*, Proc. Intl. Conf. on Acoustic, Speech and Signal Processing, vol. ASSP-30, 1982
- [5] **Cocianu C.,** *Restaurarea imaginilor perturbate normal prin matrice de împrastiere*, *Informatica Economica*, nr. 5/1998
- [6] **Cocianu C.,** *One Pixel Based Technique for Image Enhancement*, The Proceedings of the 3<sup>rd</sup> International Symposium of Economic Informatics, May 1997
- [7] **Cocianu C.,** *Îmbunătățirea contrastului unei imagini utilizând procedee de modificare a histogramei*, *Informatica Economica*, Vol. 1, Nr. 2, 1997

- [8] **Devijver P.A., Kittler J.,** *Pattern Recognition: A Statistical Approach*, Prentice-Hall Int.Inc.,1982
- [9] **Gonzales R., Woods R.,** *Digital Image Processing*, Addison Wesley, 1993
- [10] **Fukunaga K.,** *Introduction to Statistical Pattern Recognition*, Academic Press, Inc. 1990
- [12] **Pratt W.K.,** *Digital Image Processing*, Wiley, New York, 1978
- [13] **State L., Cocianu C.,** *Restaurarea imaginilor perturbate normal utilizand coeficientul Bhattacharyya*, Informatica Economica, Vol. 2, Nr.8, 1998
- [14] **State L., Cocianu C.,** *A Regressive Technique of Image Restoration*, Studii si cercetari de calcul economic si cibernetica economica, nr.4/1998
- [15] **State L., Cocianu C.,** *Statistical Based Approaches of Image Restoration*, Analele Universitatii Bucuresti, in curs de aparitie
- [16] **State L., Cocianu C.,** *New Approach of Image Restoration*, The Proceedings of the 4rd International Symposium of Economic Informatics, May 1998
- [17] **State L.,** *Analiza in componente principale pentru compresia/ restaurarea datelor* Informatica Economica, Nr, 2/1997
- [18] **State L., Cocianu C.,** *“Determinarea caracteristicilor lineare optime din punct de vedere informational in compresia si decompresia datelor”*, Informatica Economica, Nr. 4/1997
- [19] **Woods, J.W., Ingle, V.K.** *“Kalman Filtering in Two Dimensions: Further Results”*, IEEE Trans. Acoustics, Speech Processing, vol. ASSP-29, 1981