

Proiect Procesarea Semnalelor

Image Enhancement Using Genetic Algorithm

Calmîș Alina, 343C2

1. INTRODUCERE

În acest proiect am implementat algoritmul cu numărul 2 din cercetarea "A survey on Image Contrast Enhancement Using Genetic Algorithm", publicat în International Journal of Scientific and Research Publications, Volume 2, Issue 7, July 2012, cu numele : " CHROMOSOME STRUCTURE BASED ON RANDOM SELECTION OF INTEGER'S".

2. DESCRIERE ALGORITM

Algoritmii genetici sunt pe larg folosiți pentru a genera cele mai optime soluții. În abordarea problemei de îmbunătățire a contrastului imaginii, algoritmi genetici oferă un rezultat mai natural decât metoda de egalizare a histogramei, de exemplu. Un algoritm genetic funcționează în următorul fel, o populație inițială este creată. Acești indivizi se numesc cromozomi. Toți indivizii sunt evaluați folosind o funcție de fitness. Cu cât mai mare este valoarea de fitness, cu atât mai mare șansa să fie selectat individul pentru reproducerea ulterioară. Doar cei mai potriviți pot supraviețui în generația următoare. Operația de crossover se efectuează pentru a recombină informația.

În algoritmul vizat în acest proiect, se presupune îmbunătățirea calității imaginii gri. Dimensiunea cromozomului este egală cu numărul de niveluri de gri din imagine. Pentru a crește calitatea imaginii fiecare nivel de gri din imagine este înlocuit cu nivelul de gri din cromozomul îmbunătățit.

Dimensiunea populației este egală cu nivelul cel mai mare de gri din imagine. După construirea întregii populații, este analizat fitnessul fiecărui individ, conform formulei (1).

$$fitness(x) = \log(\log(E(I(x)))) * n_{edges(I(x))} \quad (1)$$

Unde x este cromozomul, $fitness(x)$ – valoarea de fitness a cromozomului, $I(x)$ este imaginea îmbunătățită, E – intensitatea calculată după formula (2), iar n_edge este numărul de margini calculate prin intermediul operatorului Sobel.

$$E(I) = \sum_x \sum_y \sqrt{\delta h_1(x, y)^2 + \delta V_1(x, y)^2} \quad (2)$$

$$\delta h_1 = g_1(x+1, y-1) + 2g_1(x+1, y) + g_1(x+1, y+1) - g_1(x-1, y-1) - 2g_1(x+1, y) - g_1(x-1, y+1)$$

$$\delta V_1 = g_1(x-1, y+1) + 2g_1(x, y+1) + g_1(x+1, y+1) - g_1(x-1, y-1) - 2g_1(x, y-1) - g_1(x+1, y-1)$$

În generația următoare trec doar $P_s - P_s * P_c$ indivizi, dintre care $P_s * P_c$ indivizi sunt generați în fiecare generație, unde P_s este numărul total de indivizi, iar P_c – rata de încrucișare. Selecția cromozomilor se efectuează random, iar crossover-ul folosit pentru generarea descendenților este în două puncte.

3. IMPLEMENTARE ALGORITM

Cum zice și titlul, acest concept al algoritmului genetic folosit pentru îmbunătățirea contrastului imaginii este bazat pe cromozomi generați random.

3.1. ETAPELE INTEGRATE ÎN CADRUL IMPLEMENTĂRII ALGORITMULUI:

- Generarea populației inițiale
- Selecție
- Crossover
- Fitness
- Analiză

Populația inițială de cromozomi a fost generată folosind un generator random de întregi. Dimensiunea fiecărui cromozom fiind egală cu numărul de niveluri de gri din imaginea care urmează să fie procesată. Vectorul generat trebuie sortat crescător.

Pentru îmbunătățirea contrastului imaginii este necesar să înlocuim fiecare nivel de gri din imaginea inițială cu nivelul de gri din cromozomul generat random folosind o funcție de transformare.

$$T(G(k)) = C_i(k), \quad k = 1, 2, 3 \dots n \quad (3)$$

Unde T este transformarea propriu-zisă, $G(k)$ este nivelul de gri din imaginea inițială, iar $C(k)$ = valoarea cromozomului. Etapă implementată de funcția *enhance*.

Se calculează fitness-ul pentru fiecare imagine nou formată. Pentru calcularea acestei valori avem nevoie de "marginile" (edges), acestea sunt punctele imaginii în care contrastele diferă esențial, punctele de trecere între obiectele din imagine și fundal, imaginii calculate cu operatorul sobel. Ulterior se folosesc formulele din articol pentru calcularea valorii de fitness. Funcția fitness.

Crossover-ul se realizează în două puncte. Fiecare generație de indivizi generează în următoarea generație doar $PS \cdot PC$ noi indivizi, unde PS este numărul de indivizi, iar PC rata de efectuare a crossover-ului.

Se continuă crearea noilor generații până nu este îndeplinită una din condițiile de terminare necesare. Deci generațiile se creează la infinit. O nouă generație este creată de funcția *perform_crossover*, unde se calculează și valoarea de fitness pentru fiecare nou individ. Din această etapă se selectează doar indivizii cu un rezultat al fitness-ului mai mare și doar în număr de $PS \cdot PC$. Fiecare individ este adăugat în lista celor existenți deja. PS -ul scade o dată cu numărul de indivizi.

3.2. CONDIȚII DE TERMINARE

Procesul se încheie în cazul împlinirii a uneia dintre următoarele condiții:

- Diferența dintre două generații consecutive este mai mică decât epsilon. Epsilon a fost definit drept $0.02 \cdot \text{cea_mai_mare_val_de_fitness}$;
- $PS = 1$

3.3. DETALII IMPLEMENTARE

Limbaj de programare:

- *Python*

Biblioteci folosite:

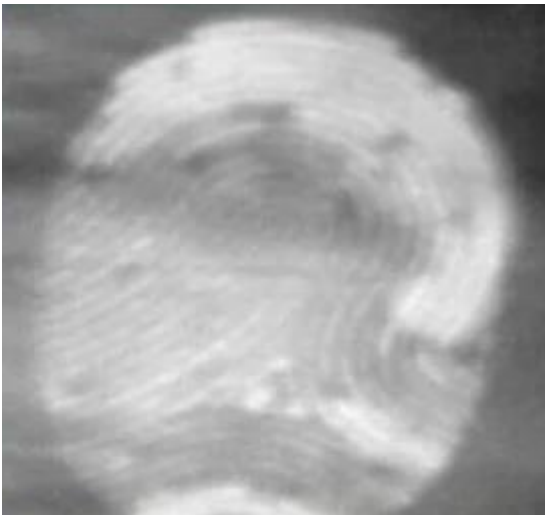
- *OpenCV*, pentru procesarea imaginilor
- *Numpy*, pentru operațiile de transformare a matricilor
- *Math*

Valori cu care am operat:

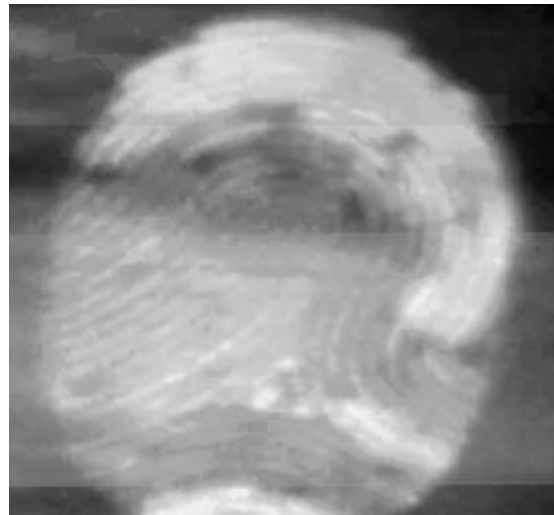
- $PS = 50$
- $PC = 0.2$
- $Num_crossovers = 20$
- $Eps = 0.02 * greatest_fit$

3.4. REZULTATE OBTINUTE

Imagine originală



Imagine prelucrată



Imagine originală



Imagine prelucrată



4. CONCLUZII

În acest proiect am implementat al treilea concept descris în articolul de cercetare "A survey on Image Contrast Enhancement Using Genetic Algorithm", realizat de Akhilesh Vema, Archana: "Chromosome

structure based on random selection of integers". Algoritmul se bazează pe crearea populației inițiale folosind un generator random de întregi de dimensiunea egală cu nivelurile de gri din imagine. Astfel, acest algoritm poate fi cu ușurință aplicat și folosit pentru imaginile în domenii precum produsele electronice de larg consum și alte tipuri de sisteme optice.