Documentație

În acest proiect am implementat algoritmul cu numarul 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*”.

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

**Etapele integrate în cadrul implementăriii 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 egala cu numărul de niveluri de gri din imainea care urmeaza să fie procesată.

Pentru îmbuătățirea contrastului imaginii este necesar să înlocuim fiecare nivel de gri din imaginea inițială cu nivelul de gri din cromozomul generat random folsind o funcție de transormare.

T(G(k)) = Ci(k), k = 1,2,3…n

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 calculeaza fitness-ul pentru fiecare imagine nou formată. Pentru calcularea acestei valori avem nevoie de ”marginile” (edges), acestea sunt punctele imaginii in 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 realizeaza în două puncte. Fiecare generație de indivizi genereaza în următoarea generație doar PS\*PC noi indivizi, unde PS este numărul de indivizi, iar PC rata de efectuare a crossover-ului.

Se continua crearea noilor generații până nu este îndeplinită una din condițiile de terminație necesare. Deci generațiile se creează la infinit. O nouă generație este creată de funcția perform\_crossover, unde se calculeaza și valoarea de fitness pentru fiecare nou individ. Din aceasta etapa se selecteaza doar indivizii cu u rezultat al fitness-ului mai mare și doar în număr de PS\*PC. Fiecare individ este adăugat in lista celor existenși deja. PS-ul scade o dată cu numărul de indivizi.

**Condiții de terminare**

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

* Diferența dintre două generașii consecutive este mai mică decât epsilon. Epsilon a fost definit drept 0.02\*cea\_mai\_mare\_val\_de\_fitness;
* PS = 1

**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*

**Rezultate obținute**

 Imagine originală Imagine prelucrată

 Imagine originală Imagine prelucrată

