

Documentatie

Proiect-PP

Proiectul meu e compus din 2 parti.

Prima parte se ocupa de criptarea unei imagini BMP prin permutarea si XOR-area pixelilor din ea si de decriptarea ei aplicand criptarea invers. Si afisarea testului chi-patrat pentru ambele imaginii.

A doua parte se ocupa de gasirea unor numere intr-o imagine printr-un algoritm de pattern-matching, eliminarea non-maximelor suprapuse si desenarea unui contur numerelor ramase.

Funcitiile folosite, si ce fac, sunt urmatoarele:

XORSHIFT32

- Este algoritmul de xorshift32, folosit pentru a genera un numar aleatoriu;
- Returneaza elemental generat

Generare

- Este algoritmul care genereaza secventa de numere pseudo-aleatori, plecand de la un seed value (R0) si folosit xorshift32 pentru a genera restul elementelor;
- Returneaza un vector alocat dinamic reprezentand sirul generat aleatoriu;

Incarcare_BMP

- Aceasta functie incarca o imagine din memorie externa, prin calea specificata in parametru, si o salveaza sub forma liniarizata print-o structura care contine un header, si un vector alocat dinamic de pixeli;
- Returneaza imaginea v;

Salvare_BMP

- Aceasta functie primeste ca parametru o imagine si o cale, si salveaza imaginea respectiva din memorie interna in memorie externa;

Durstenfeld

- Este algoritmul de generare aleatorii a unei permutarii, folosind ca numere aleatorii elementele sirului $R(n+1)....R(2n)$;
- Returneaza un vector alocat dinamic, reprezentand permutarea;

XOR_pixel_int

- Aceasta functie aplica operatia XOR intre un pixel si primi 3 octeti ai unui intreg;
- Returneaza pixelul XOR-at;

XOR_image

- Primește ca parametri un vector de pixeli c , un vector de numere aleatorii R , un starting value SV și un număr n reprezentând numărul de pixeli;
- XOR-ează cei n pixeli din vectorul c , folosind numerele aleatorii din $R(n+1) \dots R(2n)$ și cu valoarea precedentă, salvând rezultatul în vectorul c ; Prima valoare este XOR-ata cu SV ;
- Returnează vectorul c , după XOR-are;

XOR_image_inv

- Este inversul subprogramului `XOR_image`, folosit pentru decriptare (folosind proprietățile XOR-ului);

Permutare_pixel

- Primește ca parametri un vector de pixeli v , un vector permutare p și un număr n reprezentând numărul de pixeli;
- Aplică permutarea p celor n pixeli din v ;
- Returnează v_{perm} , reprezentând vectorul permutat;

Criptare_BMP

- Primește ca parametri calea imaginii inițiale, calea imaginii criptate și alea cheii secrete;
- Încarcă imaginea v , generează sirul de numere pseudoaleatorii $R(1) \dots R(2n)$;
- Generează o permutare p folosind primele n numere din sirul R ;
- Permute pixeli din v folosind permutarea p ;
- Aplică `XOR_image` pe imaginea v , folosind elementele rămase din sirul R ;
- Salvează imaginea v pe calea specificată în parametric;
- Eliberează memoria alocată dinamic;

Decriptare_BMP

- Primește ca parametri calea imaginii initiale, calea imaginii criptate și alea cheii secrete;
- Încarcă imaginea v , generează sirul de numere pseudoaleatorii $R(1) \dots R(2n)$;
- Generează o permutare p folosind primele n numere din sirul R ;
- Aplică XOR_image_inv pe imaginea v , folosind elementele rămase din sirul R ;
- Generează permutarea inversă permutării p ;
- Permutează pixeli din v folosind permutarea inversă;
- Salvează imaginea v pe calea specificată în parametric;
- Eliberează memoria alocată dinamic;

Chi_patrat

- Primește ca parametru calea unei imagini;
- Calculează frecvența fiecărui canal de culoare;
- Calculează valorile testului Chi-patrat și le afișează pe ecran

Functii folosite in partea a doua:

Grayscale_image

-Este algoritmul de grayscale dat ca material pentru proiect;

Intensitate_medie

-Calculeaza intensitatea medie a n pixeli din vectorul de pixeli v;

-Returneaza rezultatul calculului;

Deviatie

-Calculeaza deviatia celor n pixeli din vectorul de pixeli v;

-Foloseste intensitatea medie ca parametru, pentru a nu se calcula de mai multe ori

-Returneaza rezultatul calculului;

Corr

-Calculeaza corelatia dintre 2 vectori de pixeli, S si f;

-Foloseste ca parametru intensitatea medie a lui S (pentru ca reprezinta intensitatea medie a sablonului, si ea nu se schimba pana cand nu trecem la urmatorul sablon);

-Returneaza rezultatul calculului;

Template_maching

- Primește ca parametri o imagine v, o imagine S (sablon) , un prag pS și un pixel culoare (reține culoarea într-o structură, pentru a putea colora mai târziu conturul);
- Pentru fiecare poziție din v (în care încap sablonul S), calculează corelația dintre v și S, iar unde avem o corelație peste pragul pS, reținem într-o structură de ferestre fer (reținem culoarea, poziția și scorul corelației);
- Returnăm ferestrele fer;

Template_maching2

- Primește ca parametri o imagine și calea tuturor sabloanelor;
- Apelează funcția Template_maching pentru fiecare sablon;
- Returnează o structură det, reprezentând detectiile pentru toate sabloanele;

Colorare_contur

- Primește ca parametri o imagine v, o imagine S (este necesar pentru aflarea lățimii și lungimii sablonului), o poziție f și o culoare cul;
- Colorează conturul de pe poziția f din imaginea v cu culoarea cul;
- Returnează imaginea colorată;

Comp_ferestre

- Funcția de comparare pentru algoritmul qsort pentru ferestre;

Suprapunere

- Primește ca parametri două poziții $f1$ și $f2$, și două header-uri de imagini $header_S$ și $header_i$ reprezentând header-ul unui șablon și header-ul unei imagini;
- Calculează coordonatele punctelor ferestrelor de pe pozițiile $f1$ și $f2$;
- Calculează Aria șablonului și Aria intersecției;
- Returnează raportul de suprapunere dintre cele două ferestre;

Eliminarea_non_maximelor

- Primește ca parametri o imagine I , o structură de ferestre det (detectii), o cale pentru un șablon S (avem nevoie de el pentru lungimea și lățimea șablonului);
- Sortează vectorul de detectii descrescător în funcție de scorul corelației;
- Verifică de la stânga la dreapta detectiile, și unde găsește un procent de suprapunere peste 20%, marchează detectia respectivă pentru a nu fi desenată;
- Desenează conturul detectiilor rămase (maxime);
- Returnează imaginea I desenată;