

Documentatie proiect

Programare Procedurala

Mititelu George-Antonio
Grupa: 131

Fisierul “Functions.h” contine structurile si anteturile functiilor folosite.

Structuri:

- CULOAREINDEX - contine variabilele necesare pentru salvarea culorii unui pixel (R, G, B);
- infoFereastră - contine informatii despre o fereastră (pozitia in imaginea mare, corelatia, culoarea);

Functii:

- void readBMP(char *cale_image, unsigned char *header, CULOAREINDEX **L);
- void storeBMP(char *cale_image, unsigned char *header, CULOAREINDEX *L);
- void grayscale_image(char *nume_fisier_sursa, char *nume_fisier_destinatie);
- unsigned int XORSHIFT32(unsigned int x);
- void generate(unsigned int **R, unsigned int **permutare, unsigned int nrElemR);
- void encryptBMP(char *cale_image, char *cale_image_criptata, char *cale_cheie, unsigned char *header, CULOAREINDEX *L);
- void decryptBMP(char *cale_image, char *cale_image_decriptata, char *cale_cheie, unsigned char *header, CULOAREINDEX *C);
- void test_chi2(char *cale_image);
- void mediePixeli(CULOAREINDEX **M, double *medie, int row, int col);
- void deviatieStandard(CULOAREINDEX **M, double medie, double *deviatie, int row, int col);
- void getColor(char *cale_sablon, CULOAREINDEX *c);
- void templateMatching(char *cale_image, char *cale_sablon, double prag, int *nrDetectii, infoFereastră **detectii);
- int overlap(infoFereastră a, infoFereastră b);
- int minim(int a, int b);
- int maxim(int a, int b);
- double overlapArea(infoFereastră a, infoFereastră b);
- void delete_nonmaxim(infoFereastră **detectii, int *nrDetectii);
- void contur(CULOAREINDEX ***M, CULOAREINDEX culoare, infoFereastră f);
- int cmpCorelatii(const void *a, const void *b);
- void colorareImg(char *cale_image);

Fisierul “BMP.c” contine definitiile functiilor pentru salvarea/modificarea unei imagini in formatul BMP.

- `void readBMP(...)` citeste o imagine in format BMP si o salveaza in forma liniarizata in vectorul L cu elemente de tip CULOAREINDEX
- `void storeBMP(...)` folosindu-se de elementele de tip CULOAREINDEX ale vectorului L creeaza o imagine in format BMP si o salveaza in memoria externa
- `void grayscale_image(...)` transforma o imagine color intr-o imagine alb-negru

Fisierul “Encrypt.c” contine definitiile functiilor pentru criptarea/decriptarea unei imagini BMP si testul χ .

- `unsigned int XORSHIFT32(...)` implementeaza generatorul de numere pseudo-aleatoare
- `void generate(...)` genereaza o secventa de numere pseudo-aleatoare de dim $2*W*H-1$ (W si H sunt dimensiunile unei imagini) si creeaza o permutarea de $W*H$ elemente conform algoritmului lui Durstenfeld
- `void encryptBMP(...)` cripteaza o imagine BMP si o salveaza in memoria externa
- `void decryptBMP(...)` decripteaza o imagine BMP si o salveaza in memoria externa
- `void test_chi2(...)` ruleaza testul χ pentru o imagine BMP

Fisierul “Template.c” contine definitiile functiilor pentru efectuarea operatiei de template matching.

- `void mediePixeli(...)` calculeaza media intensitatii pixelilor intr-o fereastră
- `void deviatieStandard(...)` calculeaza deviatia standard intr-o fereastră
- `void getColor(...)` obtine culoarea necesara pentru desenarea unui chenar
- `void templateMatching(...)` efectueaza operatia de template matching intre o imagine si un sablon
- `int overlap(...)` verifica daca doua detectii se suprapun
- `int minim(...)`
- `int maxim(...)`
- `double overlapArea(...)` calculeaza suprapunerea dintre doua ferestre
- `void delete_nonmaxim(...)` sterge non-maximele
- `void contur(...)` deseneaza conturul ferestrei f in imagine
- `int cmpCorelatii(...)` functie necesara pentru sortarea tabloului cu detectii folosita in qsort()
- `void colorareImg(...)` apeleaza celelalte functii si salveaza in memoria externa imaginea cu chenarele desenate

Citirea se face din fisierele text DateProiect.txt si Sabloane.txt astfel:

- in fisierul DateProiect.txt se afla pe fiecare linie calea catre un fisier in aceasta ordine: numele imaginii ce urmeaza a fi criptata, numele imaginii in care va fi stocata imagine criptata, numele imaginii in care va fi stocata imagine decriptata, numele fisierului text din care se va citi cheia secreta si numele imaginii supuse operatiei de template matching;
- in fisierul Sabloane.txt se afla pe prima linie numarul de sabloane, iar apoi pe fiecare linie numele unui sablon;