Pattern Matching

Proiectul are urmatoarea structura :

1. Zona bilbiotecilor, macrodefinitiilor si a definii de structuri

2. Functia int main()

3.Functia Eliminare()

4. Functia Sort()

5. Functia Cmp()

6. Functia Contur()

7. Functia TemplateMatching()

8.Functia DeviatiaStandardFereastra ()

9. Functia DeviatiaStandard ()

10. Functia MedieIntensitatiFereastra ()

11. Functia MedieIntensitati ()

12. Functia realocare ()

13. Functia grayscale ()

14. Functia truncate ()

15. Functia ScrieMatrice2 ()

16. Functia Citeste Matrice2()

17. Functia CitesteHeader()

1. Zona bilbiotecilor, macrodefinitiilor si a definii de structuri

- definesc ( #define ) dimensiunea Headerului - 54

- folosesc pragma pack (1) pentru a elimina paddingul din structura ce va definita in continuare

- definesc o structura (Header) in care se pastreaza toate tipurile pentru campurile din header ,precizandu-se rolul lor ( b1 )

- definesc o structura (BMPMatrix) pentru a pastra pixelii din imaginile ce urmeaza a fi prelucrate

- definesc o structura (Punct) in care se memoreaza:

- 2 coordonate : x si y

- indicele tabloului : val

- valoarea corelatiei in acel punct pentru sablonul cu valoarea val : corr

2. Functia int main()

- initializez si citesc din fisierul NumeFisiere.txt ( din care se vor face toate citirile de nume de fisiere din program ) numele fisierelor pe care urmeaza sa le folosesc in program

- daca nu sunt gasite opresc executia programului afisand un mesaj corespunzator

- initializez matrice de pixeli pentru imaginea care urmeaza sa fie prelucrata

- apelez functia CitesteHeader () (17)

- initializez dimensiunile matricei ( informatii care se gasesc in header )

- calculez nr de pixeli in dim si paddingul in padding

- apelez functia CitesteMatrice()(16) pentru I

-apelez functia grayscale ()(13) pentru I

- apelez functia CitesteMatrice()(16) pentru II

- toate calculele si masuratorile se vor face folosind matricea I

- matricea II va fi folosita doar pentru a colora conturul detectiilor si pentru salvare in memoria externa

- aloc memorie pentru vectorul de pixeli in care vor fi pastrate culorile si il initializez

- apelez functia CitesteHeader()(17) pentru primul Sablon pentru a calcula dimensiunile acestuia si paddingul ( se cunoaste faptul ca toate sabloanele folosite in cadrul proiectului au aceleasi dimensiuni )

- apelez realocare()(12) pentru I , marind matricea cu dimensiunile sablonului - 1, pentru ca nu iesi din ea in momentul in care fac calculele ce urmeaza

- aloc memorie pentru vectorul de puncte D ( va fi folosit pentru a retine detectiile )

- pentru fiecare dintre cele 10 sabloane :

- citesc numele fiecarui sablon

- apelez CiesteMatrice()(16)

- apelez grayscale()(13)

- apelez TemplateMatching()(7) si creez ,astfel, vectorul fI care este un axuliar in formarea lui D

- Realoc memoria pentru D in functie de nr de elemente din fI

- copiez elementele lui fI in memoria nou alocata

- adaug la valorile copiate indexul sablonului

- maresc dimensiunea lui D

- eliberez memoria pentru fI, deoarece se va realoca la urmatorul apel TemplateMatching()(7), iar valorile lui sunt deja copiate in D

- eliberez memoria pentru sablonul S, cum nu mai este utilizat

- apelez functia Sort()(4) pentru D

- apelez Eliminare() (3)pentru D

- pentru fiecare valoare din D apelez Contur()(6) pentru matricea II

- apelez ScrieMatrice()(15) pentru II

- eliberez toata memoria alocata dinamic

3.Functia Eliminare()

- verifica daca 2 sabloane di si dj se intersecteaza

- daca da ,calculeaza suprapunerea dintre doua sabloane di si dj dupa formula data

- daca suprapunerea >= 0.2 s elimina sablonul cu corelatie mai mica ( i<j, se elimina j, pentru ca tabloul este sortat descrescator in functie de corelatii )

4. Functia Sort()

- sorteaza descrescator un tablou unidimensional de tip Punct , in functie de corr, x,y, in aceasta ordine, folosind functia qsort din biblioteca stdlib si utilizand comparatorul Cmp ()(5)

5. Functia Cmp()

- comparator pentru qsort

- retuneaza o valoare de tip int comparand cei 2 parametrii ai sai

- b < a => sunt soratate

- a = b =>ma uit la urmatorul camp

- a < b=> trebuie sortate

6. Functia Contur()

- folosita pentru a desena conturul unui sablon dreptunghic incepand din punctul dat cu o culoare data

-conturul este echivalent cu rama care incadreaza cifra

7. Functia TemplateMatching()

- returneaza un tablou unidimensional, pentru care aloc memorie la inceput

- apelez MedieIntensitati()(11)

- apelez DeviatiaStandard ()(9)

- pentru fiecare punct din imagine :

- apelez MedieIntensitatiFereastra()(10)

- apelez DeviatiaStandardFereastra()(8)

- calculez corelatia corr dupa formula data

- daca corr > pragul , salvez punctul pe care ma aflu in vectorul fI,

marind dimensiunea lui

- in vector salvez si valoarea corelatiei pentru acel punct ( imi va trebui la sortare)

- returnez tabloul obtinut

8.Functia DeviatiaStandardFereastra ()

- calculeaza deviatia standard pentru tabloul bidimensional trasnmis ca parametru ,dupa formula data, tinand cont de faptul ca tabloul transmis este o fereastra dintr-o imagine mai mare

9. Functia DeviatiaStandard ()

- calculeaza deviatia standard pentru sablonul trasnmis ca parametru ,dupa formula data

10. Functia MedieIntensitatiFereastra ()

- calculeaza media intensitatii pixelilor pentru un tablou bidimensional transmis ca parametru , tinand cont de faptul ca se afla intr-o imagine mai mare => calculeaza media intensitatii pixelilor pentru o fereastra

11. Functia MedieIntensitati ()

- calculeaza media intensitatii pixelilor pentru un tablou bidimensional transmis ca parametru

12. Functia realocare ()

- rol de bordare

- realoca memorie pentru o zona data de tip BMPMatrix

- initializeaza ce a fost adaugat cu pixelul ( 0, 0, 0)

13. Functia grayscale ()

- transforma o imagine din color in grayscale

- apelez in cadrul ei functia truncate()(14) pentru fiecare pixel pentru a evita depasirile tipului de date unsigned char

- cele trei canale ale fiecarui pixel devin egale

-este de tip void, matricea fiind transmisa ca un pointer triplu

14. Functia truncate ()

-returneaza o valoare intre 0 si 255

- se foloseste doar in cadrul functiei grayscale

15. Functia ScrieMatrice2 ()

-salveaza in memoria externa, la o destinatie data, un dublu pointer de tip BMPMatrix ( matrice), precedat de o variabila de tip Header ( headerul imaginii si imaginea in pixeli )

- scrie imaginea invers

- tine cont de padding

16. Functia Citeste Matrice2()

- parseaza headerul unei imagini cu vectorul buff[]

- citeste pixel cu pixeli matricea, luand in calcul paddingul

- o rastoarna si o incarca in memoria interna

- returneaza un pointer dublu catre matricea de pixeli citita

17. Functia CitesteHeader()

- citeste o variabila de tip header ( headerul unei imagini )

- returneaza variabila citita

Bibliografie :

(b1) https://engineering.purdue.edu/ece264/17au/hw/HW15

(b2) EnuntProiect.pdf