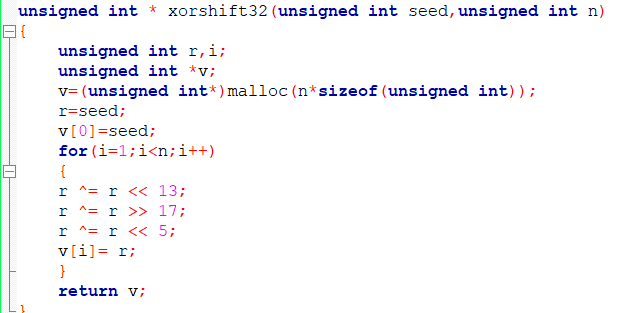
Prezentare proiect

Functiile pe care le-am implementat pentru a realiza acest proiect sunt urmatoarele:

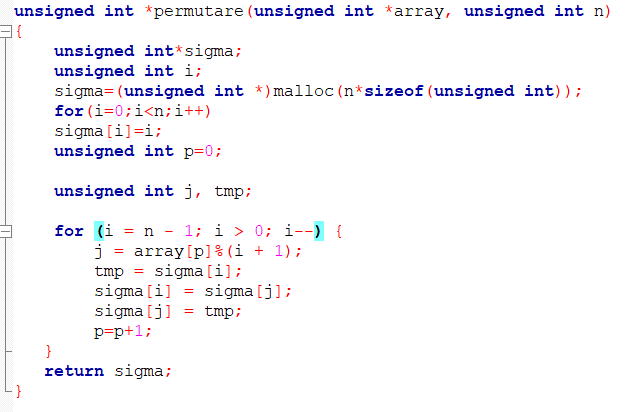
Pentru criptare /decriptare

Functia xorshift32



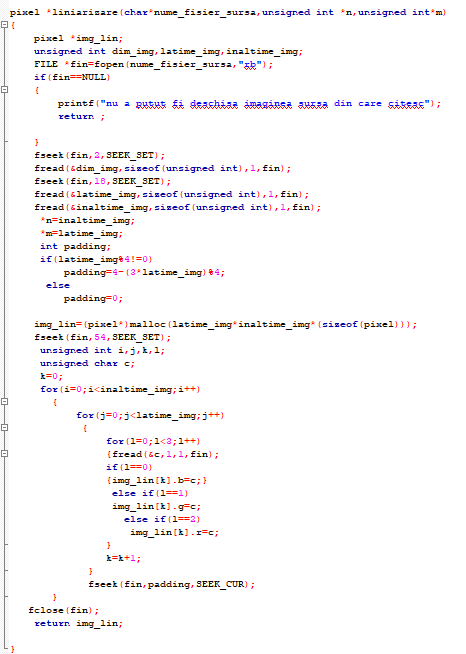
Aceasta functie genereaza un numar n de numere pseudo-aleatoare, adica o secventa de numere pentru care probabilitatea de aparitie a unei anumite valori este independenta de toate valorile generate anterior, plecand de la o valoare initiala seed. Valorile generate sunt salvate in vectorul v, care va fi returnat la final.

Functia permutare



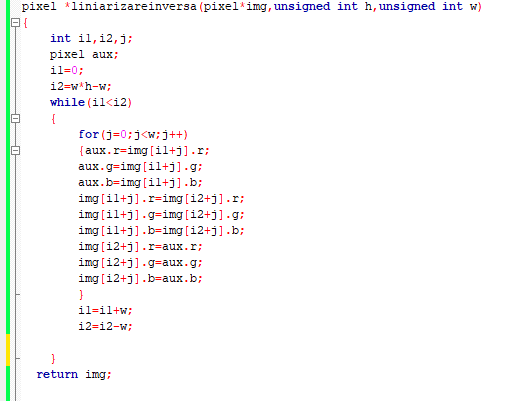
Aceasta functie este utilizata pentru a genera o permutare de dimensiune n. Pentru a fi realizata aceasta, este folosit algoritmul lui Durstenfeld. Astfel se prelucreaza vectorul array cu acest algoritm, iar permutarea este returnata prin vectorul sigma.

Functia liniarizare



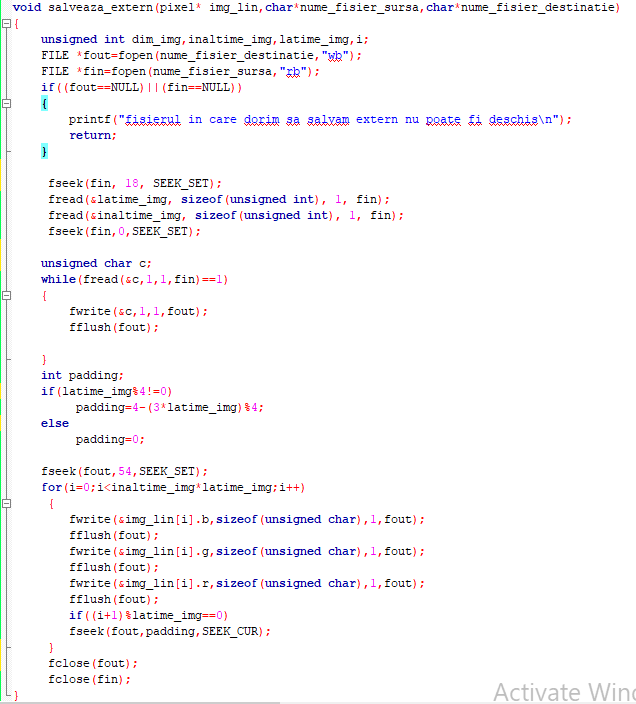
Aceasta functie este utilizata pentru a liniariza o imagine. Functia are 3 parametri, fisierul de unde va fi citita imaginea si n , m doua variabile in care vor fi salvate ulterior inaltimea si respectiv latimea imaginii. Se va crea un vector de tip pixel (o structura ce cuprinde 3 campuri, r,g,b, acestea reprezentand culoarea unui pixel) , care va fi returnat la final prin vectorul img\_lin.

Functia liniarizareinversa



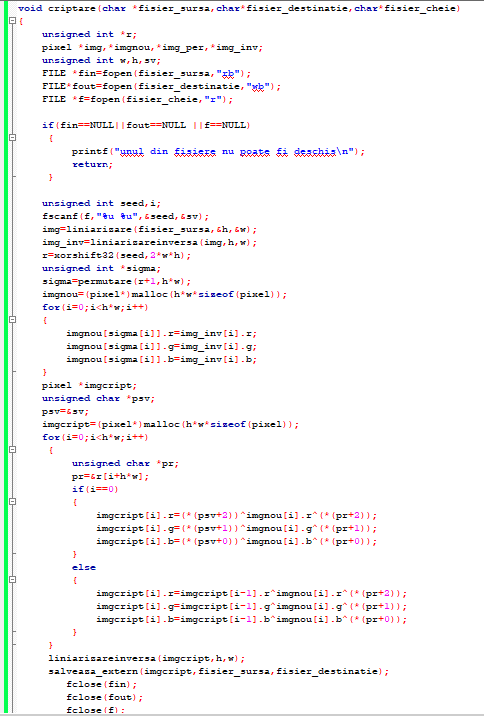
Aceasta functie este folosita pentru a realiza inversarea liniilor unei imagini ,intr-un vector img (ce reprezinta liniarizarea unei imagini). Deorece in liniarizarea propriu-zisa imaginea apare ca fiind rasturnata. Rezultatul vectorului va fi returnat la final.

Functia salveaza\_extern



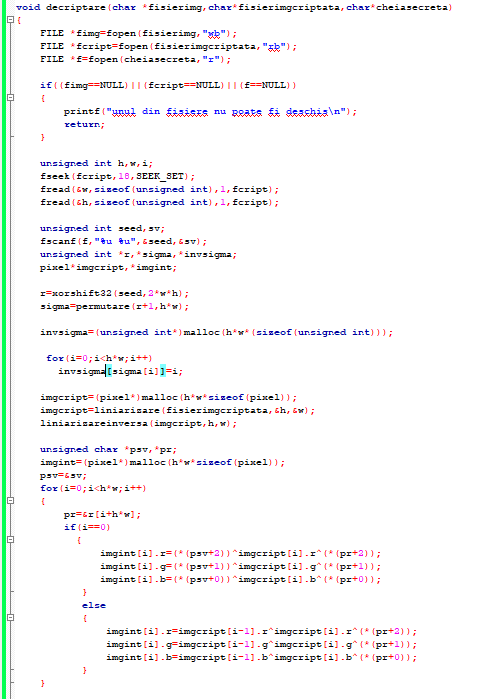
Aceasta functie este folosita pentru a salva in memoria externa o imagine BMP stocata in forma liniarizata in memoria interna . Functia are 3 parametri , primul reprezentand imaginea liniarizata iar urmatoarele doua reprezinta cele doua fisiere, in care se afla imaginea si in care se va salva liniarizarea acesteia . Astfel, este copiat fisierul cu imaginea ,in fisierul destinatie , pentru a fi copiat header-ul , ulterior ne deplasam cu 54 de octeti fata de pozitia de inceput a fisierului si copiem in fisierul destinatie vectorul ce reprezinta liniarizarea imaginii . Avem grija sa trecem peste octetii care reprezinta padding-ul.

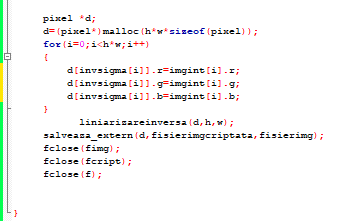
Functia criptare



Aceasta functie se uitilizeaza pentru criptarea unei imagini. Functia are 3 parametri, primul parametru reprezinta fisiereul din care se citeste imaginea pe care urmeaza sa o criptam, al doilea parametru reprezinta fisierul in care va fi salvata imaginea criptata, iar ultimul este un fisier text din care va fi citita cheia secreta. Astfel, sunt deschise toate fisierele, sunt citite din fisierul text cheia secreta (variabila sv) si valoarea de inceput utilizata pentru generarea numerelor aleatorii (variabila seed). Se foloseste functia xorshift32 pentru a genera 2\*latimea imaginii (reprezentata in program de variabila w)\*inaltimea imaginii (in program fiind variabila h) numere . Se apeleaza functia permutare , folosind elemete din vetorul generat anterior. Vectorul cu permuarea (sigma) va fi aplicat pe liniarizarea inversata a imaginii, creandu-se astfel un nou vector( imgnou). Se aplica formula de criptare utilizand operatia XOR.( Pointerii pvs si pr sunt utilizati pentru a lucra mai usor cu octetii numerelor intregi , cheia si respectiv elementele vetorului r) . In urma aplicari formulei se va crea un nou vector (imgcript) care la final va fi liniarizat invers si va fi salvat extern.

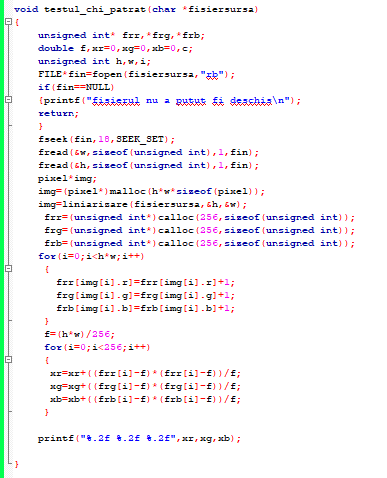
Functia decriptare





Aceasta functie este folosita pentru decriptarea imaginilor. Parametrii sai sunt reprezentati de, fisierul in care se va salva decriptarea , fisierul ce contine imaginea criptata, si fisierul in care se afla cheia secreta . Si in aceasta functie sunt apelate functiile xorshift32 si permutare . Se liniarizeaza imaginea care trebuie decriptata , se aplica fomula de decriptare si apoi , pe vectorul creat se aplica imversa permuatrii sigma realizate la apelarea functiei pemuatre (invsigma).

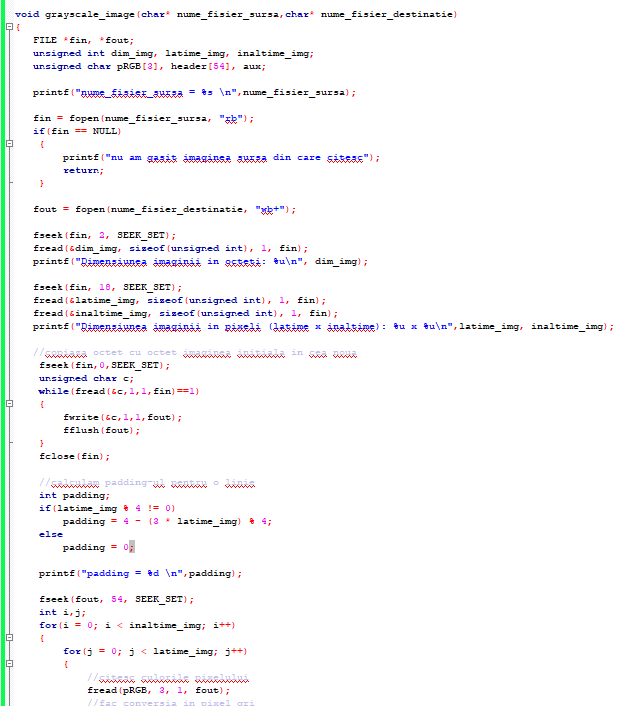
Functia testul\_chi\_patrat



Aceasta functie este utilizata pentru a masura uniformitatea distributiei valorilor dintr-un sir. Se liniarizeaza imaginea si se afla frecventa ficarei valori pe fiecare canal de culore , in algoritm aceasta frecventa este reprezentata de cei 3 vectori frr, frg, frb. La final se aplica formula pentru realizarea testului chi\_patrat.

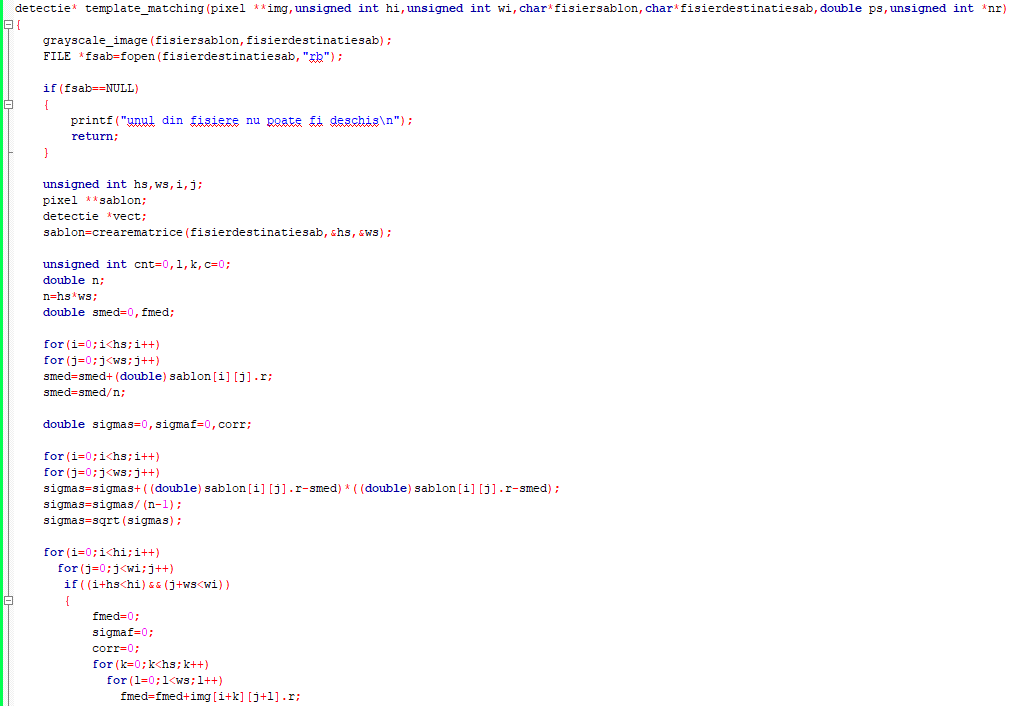
Pentru recunoasterea de pattern-uri intr-o imagine

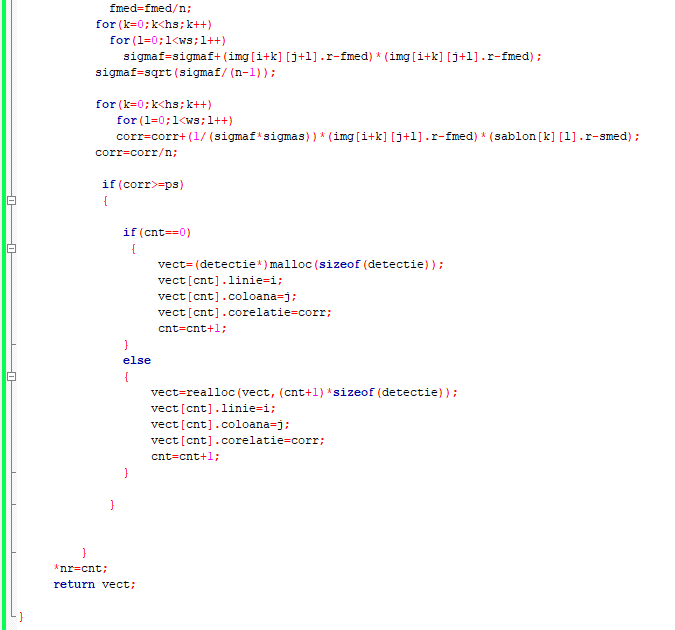
Functia grayscale\_image



Aceasta functie este utilizata pentru a transforma o imagine color intr-o imagine in tonuri de gri.

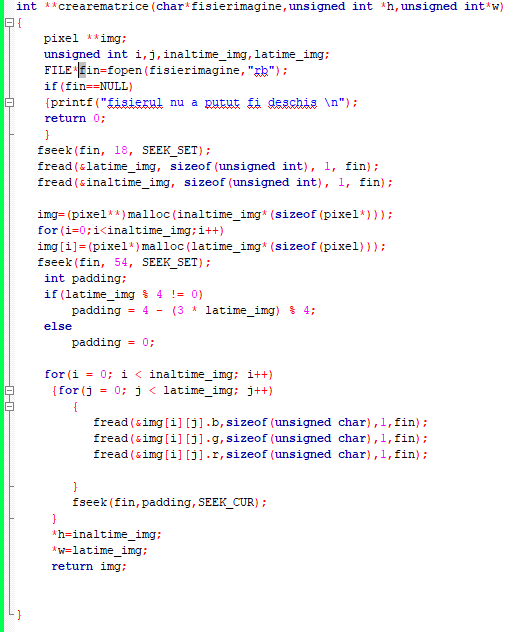
Functia template\_matching



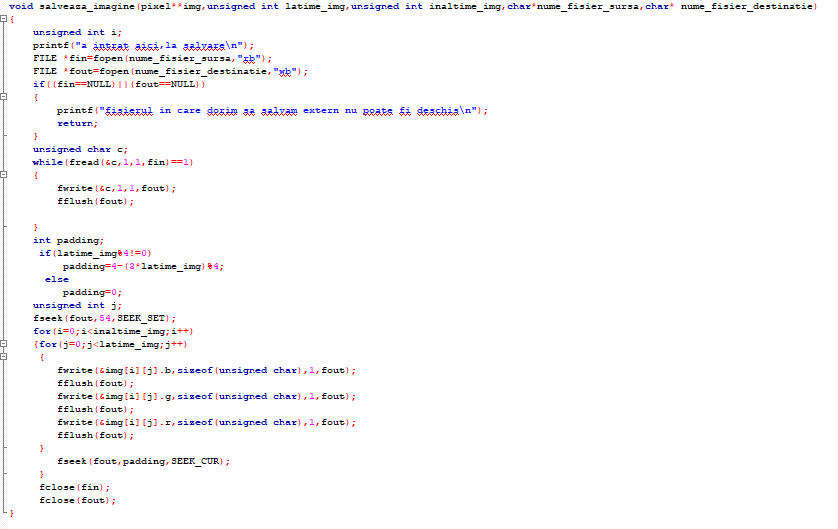


Aceasta functie determina detectiile unui sablon dintr-o imagine. Functia are ca parametrii: imaginea (data sub forma de matrice), inaltimea acesteia , latimea ei, fisierul din care se citeste sablonul , fisierul in care va fi pusa imaginea grayscale a sablonului , pragul ps, si un numar reprezentand numarul de detectii care au cel putin pragul ps (care vor fi gasite in aceasta functie). Sablonul dupa ce este transformat grayscale si reprezentat in forma matriceala este glisat pe intreaga imagine. Cu ajutorul formulei de calcul al corelatiei, pentru fiecare pixel (cu exceptia celor care se afla la margine, iar sblonul nu mai incape in imagine pentru a fi glisat) se calculeaza o corelatie. Daca aceasta corelatie este mai mare decat pragul dat , se va considera gasita o detectie care va fi salvata in vetorul vect, care este de tipul detectie (o structura care are campul linie, coloana, culoare si corelatie). Vectorul creat va fi returnat la final. (Linia si coloana reprezinta indicii matricei de la care incepe detectia).

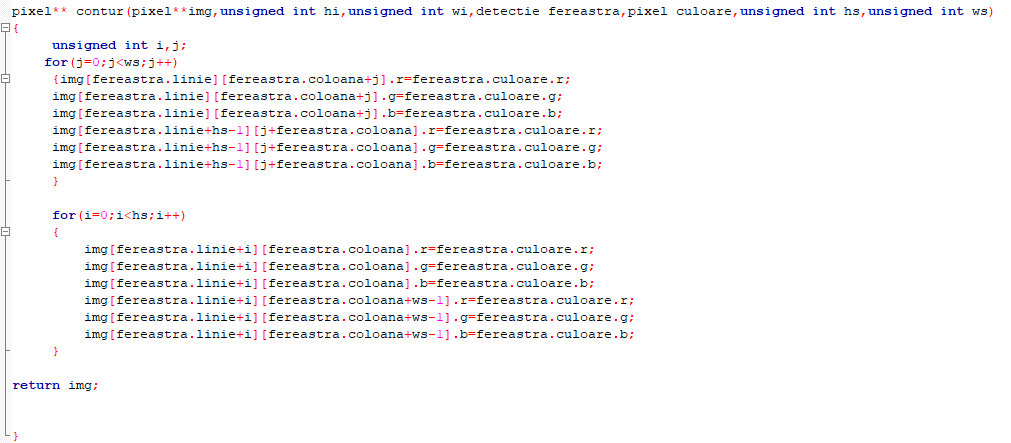
Functia crearematrice



Aceasta functie creeaza o matrice cu imaginea care se afla in fisierul primit ca parametru. Matricea va fi returnata la final. (Mai sunt 2 parametrii, ei reprezentand inaltimea si latimea imaginii).

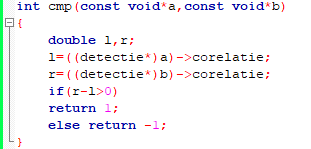
Functia salveaza\_imaginea 

Aceasta functie salveaza o imagine ce se afla in memeorie sub forma matriceala intr-un fisier. (imagine bmp)

Functia contur

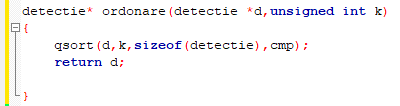
Aceasta functie este utilizata pentru a realiza conturul unei frestre,cu o anumita culoare , imaginea fiind data in forma matriceala .

Functia cmp



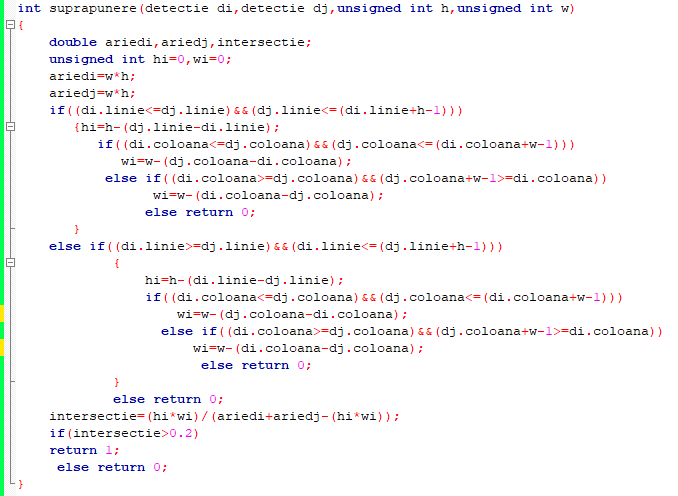
Functia cmp este functia folosita pentru compararea corelatiilor unor detectii , aceasta functie fiind folosita la qsort.

Functia ordonare



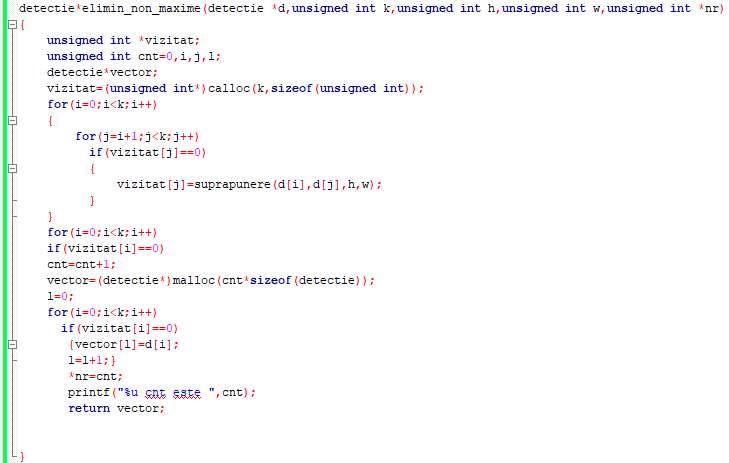
Aceasta functie este folosita pentru a ordona descrescator corelatiile unui vector de detectii.

Functia suprapunere



Aceasta functie calculeaza latimea si lungimea dreptunghiului creat prin suprapunerea celor doua detectii date ca parametrii. Daca aceste detecti au intersectia mai mare de 0,2 se va returna valoarea 1 iar daca nu valoarea 0.

Functia elimin\_non\_maxime



Aceasta functie elimina detectiile care au o suprapunere mai mare de 0.2. Functia primeste ca parametru un vector in care sunt incluse toate detectiile unei imagini, acesta fiind sortat descrescator. Functia foloseste un vector, numit vizitat, avand k (numarul de detecti din vectorul d) elemente cu valoarea 0 sau 1.Algoritmul functioneaza astfel,se ia fiecare detectie,si se verifica suprapunerea sa cu cele aflate dupa ea in vectorul de detectii. Daca suprapunerea este mai mare de 0,2, atunci valoarea vectorului vizitat corespunzatoare detectiei cu care se verifica va primii valoarea1, altfel va ramane 0.

La final, pe baza vectorului vizitat, se creeaza vectorul vector care cuprinde doar detectiile care au valoarea vetorului vizitat corespunzator indicelui lor egala cu 0.