1. **Introducere**

Formatul de fișier BMP (en. Bitmap - hartă de biți), cunoscut de asemenea și ca fișier imagine Bitmap sau formatul de fișier DIB (en. Device Independent Bitmap - „bitmap” independent de dispozitiv) sau pur și simplu Bitmap, este un format de fișier imagine de tip rastru folosit pentru a stoca imagini digitale independent de dispozitivul de afișare (cum ar fi un adaptor grafic) în special pe sistemele de operare Microsoft Windows și OS/2.

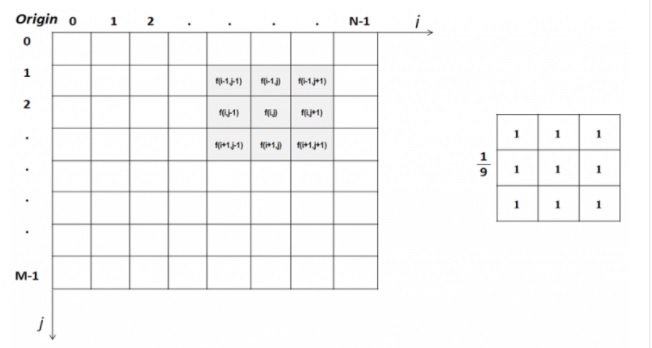
Formatul de fișier BMP este capabil de a stoca imagini digitale bidimensionale de lățime, înălțime și rezoluție arbitrare, monocrome sau color, de variate profunzimi ale culorii iar opțional cu comprimare de date, canale alfa și profile de culoare. Procesul de smoothing consta in crearea unei functii aproximative care pastreaza datele importante din fisierul bitmap, ignorand structurile neimportante din fisier.

1. **Descrierea aplicatiei cerute**

Scopul aplicatiei este de a blura o imagine, folosind bufferul rgb(red, green blue) al imaginii, si aplicand o masca de convolutie 3x3, facand media bitilor rgb ai imaginii, rezultatul fiind o imagine blurata fata de cea originala.

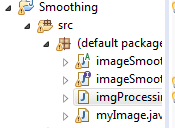
1. **Partea teoretica**

Algoritmul are la baza aplicarea unei masti de convolutie 3x3 pentru bitii rgb ai unei imagini. Pentru fiecare pixel se calculeaza valoarea aplicand masca de convolutie 3x3 pentru pixelii vecini.

1. **Descrierea aplicatiei**

Aplicatia contine clasa principala myImage, care contine metodele readImage, writeImage si smoothImage, mostenind clasa abstracta imageSmoothing care impune implementarea acestor metode.

Aplicatia contine doua clase, una abstracta, si una care o mosteneste pe cea abstracta si implementeaza metodele ei.



**public** **abstract** **class** imageSmoothing

**public** **class** myImage **extends** imageSmoothing

**Conceptele POO:**

**Mostenirea –** clasa myImage mosteneste toate atributele si metodele din clasa parinte imageSmoothing, acestea nefiind private, ci protected(fiind vizibile in clasa copil).

**Polimorfism –** clasa imageSmoothing este de tipul abstract, ea avand metodele abstracte writeImage() si smoothImage(). Aceste doua metode, implementate in clase copil, vor avea forme diferite in functie de clasa in care sunt implementate.

**Incapsulare –** toate atributele din clasa imageSmoothing sunt de tipul protected(pentru a putea fi vizibile in clasele copil). In clasa copil myImage, toate atributele sunt private, pentru a nu putea fi accesibile din exterior. Pentru aceasta se definesc setteri si getteri pentru atributele necesare.

**Abstractizare –** orice clasa mosteneste clasa imageSmoothing va trebui sa ii implementeze metodele abstracte

1. **Descrierea modulelor**

Metoda readImage citeste pixel cu pixel imaginea din calea sursa, introdusa de la tastatura, iar din headerul imaginii citeste width si height.

Metoda writeImage scrie pixel cu pixel imaginea, in calea introdusa de la tastatura.

Metoda readPixels citeste pixelii imaginii pentru a putea fi folositi mai departe de metoda smoothImage(folositi pentru a crea bufferul rgb al imaginii)

Metoda smoothImage aplica metoda descrisa mai sus, folosind bufferul rgb, asupra caruia aplica masca de convolutie, facand pentru fiecare pixel media vecinilor de pe aria 3x3 a mastii.

In main se apeleaza aceste functii si se citeste rata de smooth a imaginii(cu cat este mai mare, cu atat imaginea se va blura mai mult).

1. **Performante**

Algoritmul are o performanta liniara, timpul de procesare fiind direct proportional atat cu dimensiunea imaginii, cat si cu rata de smooth a ei. Cu cat acestea sunt mai mari, cu atat timpul de procesare(calculat in secunde pentru fiecare metoda) este mai mare.

1. **Concluzii**

Algoritmul de procesare pe care l-am implementat este unul simplu, care face o operatie aritmetica pentru fiecare bit in parte, si care se executa intr-un timp foarte mic(cateva secunde, in functie de rata de smooth). Metodele implementate au fost facute pentru a indeplini o singura functionalitate, dupa cum sugereaza si denumirea lor.

**BIBLIOGRAFIE**

<https://ro.wikipedia.org/wiki/BMP_(format_fi%C8%99ier)>

<https://en.wikipedia.org/wiki/Smoothing>

<https://en.wikipedia.org/wiki/Kernel_(image_processing)>

<http://praveentutor.com/image-blurring-weighted-average-filter-image-smoothing/>

<https://stackoverflow.com/questions/6852194/image-processing-smoothing>