

## Level Adjustment

### (Contrast and B/W adjustment)

#### Introducere

Scopul lucrării este prelucrarea de imagini prin ajustarea nivelului contrastului.

Contrastul este diferența dintre înnegrirea maximă și cea minimă care se pot vedea într-o imagine fotografică. În percepția vizuală a lumii reale, contrastul este determinat de diferența dintre culoarea și luminozitatea unui obiect și alte obiecte din interiorul aceleiași câmp vizual. (1)

#### Descrierea aplicației

Aplicația are la bază principiile programării orientate pe obiecte și secvențe de cod low-level.

Aplicația conține două pachete. Pachetul 1 are în componență doar clasa de TestClass în care se regăsește metoda main(). În cadrul metodei main calculăm timpul de execuție a procesării imaginii și aplicăm metoda ce implementează algoritmi de procesare. Clasa TestClass este derivată din clasa Modif.

```
//Nivel de mostenire 4
public class TestClass extends Modif {

    static public void main(String args[]) throws IOException {

        Modif m = new Modif(); //cream un obiect de tip Modif pentru a apela metodele din aceasta clasa
        long startTime = m.getCurrentTime(); //salvam timpul inaintea executiei metodei
        m.modific(); //apelam metoda de modificare
        long endTime = m.getCurrentTime(); //salvam timpul la care s-a terminat executia metodei
        long duration = (endTime - startTime); //calculam cat a durat executia metodei
        System.out.println("Procesare finalizata - timp (milisecunde)");
        System.out.print(duration); //afisam durata
        System.out.println("\n");
    }
}
```

Pachetul 2 conține toate celelalte clase din componența aplicației, în continuare voi încerca să descriu, pe rând, funcționalitatea acestora.

Clasa Interfata este de tip interface, am definit în cadrul acesteia două metode pe care le vom utiliza în cadrul claselor ce implementează interfața.

```
//Interfata
public interface Interfata {

    //Metoda care implementeaza procesarea imaginii
    public static void modific() {
    };

    //Metoda pe care vom ilustra principiul polimorfismului
    public static void level() {
    };
}
```

Clasa Time este de tip abstract și are în componență definiția metodei abstracte getCurrentTime() pe care o vom utiliza pentru a calcula timpul pentru fiecare etapă.

```
//Clasa abstracta
abstract class Time {

    //Metoda abstracta
    public abstract long getCurrentTime();
}
```

Clasa BlackAndWhite este derivată din clasa Time și implementează interfața. În cadrul acesteia găsim o primă varintă a metodei level() declarată în clasa de tip interface. Ceea ce am făcut de fapt aici a fost să ofer o opțiune suplimentară utilizatorului, aceea de a transforma imaginea în B/W înainte de a-i modifica contrastul. Ceea ce mai poate face este să modifice nivelul de B/W prin modificarea parametrilor din cod, adică în loc de 0.21, 0.72, 0.07 aceștia pot alege orice alte valori care respectă regula  $R+G+B=1$ . (2)

```
int red = (int) (c.getRed() * 0.21); //modificam nivelul de rosu
int green = (int) (c.getGreen() * 0.72); //modificam nivelul de verde
int blue = (int) (c.getBlue() * 0.07); //modificam nivelul de albastru
int greyscale = (red + green + blue); //greyscale=R+B+G=1(formula generala)
Color newColor = new Color(greyscale, greyscale, greyscale); //se creeaza o noua culoare
```

Tot în această clasă am implementat și metoda abstractă getCurrentTime().

```
public long getCurrentTime()
{
    return System.currentTimeMillis();
}
```

Clasa LevelAdjustment este derivată din clasa BlackAndWhite și implementează, la rândul ei, Interfața. Găsim în cadrul ei o altă implementare a metodei level() în care se modifică contrastul imaginii.

Ne vom folosi de următoarea formulă:

$$g(x) = f(x) * L1 + L2$$

Parametrii L1 și L2 sunt luminozitatea relativă a culorilor deschise, respectiv luminozitatea relativă a culorilor închise.  $g(x)$  reprezintă valoarea pixelului imaginii finale, iar  $f(x)$  valoarea pixelului imaginii de intrare. (3)

```
Color color2 = new Color(saturation(color.getRed() * L1+ L2),
    saturation(color.getGreen() * L1 + L2),
    saturation(color.getBlue() * L1 + L2)); //se modifica nivelul fiecarei culori
```

Clasa Modif este derivată din clasa LevelAdjustment și implementează Interfața. Această clasă este cea care conține etapele de citire din fișier și scriere în fișier, precum și partea de procesare a imaginii în funcție de preferințele utilizatorului și de tratare a excepțiilor. În definirea metodei modific am folosit varargs.

```
public void modific(BufferedImage ...im) { //varargs
```

Exemplu de utilizare

Imagine inițială:



1)

```
Introduceti path-ul fotografiei pe care doresti sa o editati si numele acesteia:  
C:\Users\ioana.boriceanu\Documents\workspace\Tema_AWJ\sample_images\imagine.bmp  
Introduceti path-ul si numele cu care vreti sa se salveze fotografia dupa modificari:  
C:\Users\ioana.boriceanu\Documents\workspace\Tema_AWJ\imagine.bmp  
Alegeti modificarea dorita: B/W:1 sau Contrast:2  
1  
Citire finalizata - timp (milisecunde)  
38164  
  
Scriere finalizata - timp (milisecunde)  
38164  
  
Precesare finalizata - timp (milisecunde)  
38505
```

Imagine după procesare:





2)

```
<terminated> TestClass [Java Application] C:\Program Files\Java\jre1.8.0_311\bin\javaw.exe (15 feb. 2022, 00:46:07)
Introduceti path-ul fotografiei pe care dori sa o editati si numele acesteia:
C:\Users\ioana.boriceanu\Documents\workspace\Tema_AWJ\sample_images\image.bmp
Introduceti path-ul si numele cu care vreti sa se salveze fotografia dupa modificari:
C:\Users\ioana.boriceanu\Documents\workspace\Tema_AWJ\image.bmp
Alegeti modificarea dorita: B/W:1 sau Contrast:2
2
Citire finalizata - timp (milisecunde)
22290

Introduceti valoarea pentru alpha:
0.5
Introduceti valoarea pentru beta:
5
Scriere finalizata - timp (milisecunde)
22290

Precesare finalizata - timp (milisecunde)
30346
```

Imagine după procesare:



## Referințe

- (1) <https://dexonline.ro/definitie/contrast>
- (2) <https://www.dynamsoft.com/blog/insights/image-processing/image-processing-101-color-space-conversion/>
- (3) <https://medium.muz.li/the-science-of-color-contrast-an-expert-designers-guide-33e84c41d156>