

# PROJET FINAL DE NSI EN CLASSE DE 1<sup>ère</sup> : Image Lab

## Description :

Image Lab est un logiciel de traitement d'image codé en Python avec les librairies PIL (pour la manipulation des images) et Tkinter (pour l'interface graphique).

Ce logiciel supporte uniquement le format .jpg et pour les tables de couleurs le format .pctby (format ajouté par nous les développeurs).

Au sein de ce logiciel vous avez accès à de nombreuses méthodes d'édérations, qu'elle soit de forme ou de couleur. Parmi les éditions de couleurs, vous pouvez faire un négatif, un niveau de gris ou ignorer (ou garder) des canaux, vous pouvez aussi utiliser une table de couleurs/une palette de couleurs. Parmi les éditions de forme vous avez accès à du rognage, du re-taillage et de la symétrie.

## Description de code par Maxime (algaunor) :

```
def getPalFile():
    pathi = askopenfilename(filetypes=[('Color Tables File', '*.pctby')])
    try:
        with open(pathi, 'r+') as pfile:
            pfile.write('')
            lines = pfile.readlines()
            pfile.close()
            global current_pal
            current_pal = []
            for i in lines:
                line = i.split(';')
                current_pal.append((int(line[0]),int(line[1]),int(line[2])))
    except:
        pass
```

Cette méthode permet d'ouvrir un fichier correspondant à une palette de couleur.

Au début, on demande à l'utilisateur d'ouvrir le fichier avec askopenfile (le format doit être en .pctby).

Dans le bloc try, with as permet d'ouvrir le fichier. Une fois le fichier ouvert, le programme edite current\_pal.

## Description de code par Lucas (LTHCTheMaster) :

```
def mdcl(self, img, color_list):
    """Return the color matrix of the image with only specified colors in color_list"""
    wd, hg = img.size
    out = []
    for i in range(hg):
        line = []
        for j in range(wd):
            colr, colg, colb = img.getpixel((j,i))
            line.append(color_list[self.closest((colr,colg,colb), color_list)])
        out.append(line)
    #
    return out
```

La méthode mdcl permet à partir d'une image de créer une matrice de couleurs tirés d'une palette de couleur,

Pour ce faire, le programme va parcourir l'image pixel par pixel et chercher la couleur de la palette de couleur la plus proche de la couleur d'origine. Pour ce programme les couleurs sont comme des vecteurs tridimensionnels, donc les couleurs les plus proches entre elles sont les couleurs séparées par la plus petite distance.

Distance entre deux vecteurs tridimensionnels :  $\sqrt{(x_b - x_a)^2 + (y_b - y_a)^2 + (z - z_a)^2}$  avec x pour le rouge, y pour le vert, z pour le bleu

## Extensions possibles :

De nombreuses extensions de ce programme sont possibles, une petite liste pour mieux le décrire :

- Ajouter des rotations
- Ajouter des histogrammes
- Ajouter la possibilité de dessiner sur le canvas pour dessiner une image
- Ajouter d'autres fonctions de manipulation de couleurs comme la possibilité d'intervertir deux canaux
- Ajouter des découpes diverses et variées basées sur des formes
- Ajouter de la génération d'image à l'aide de diverse méthodes, comme par exemple : des pixels aléatoires