PCA

Projet

Hai Nam TRAN

hai-nam.tran@univ-brest.fr

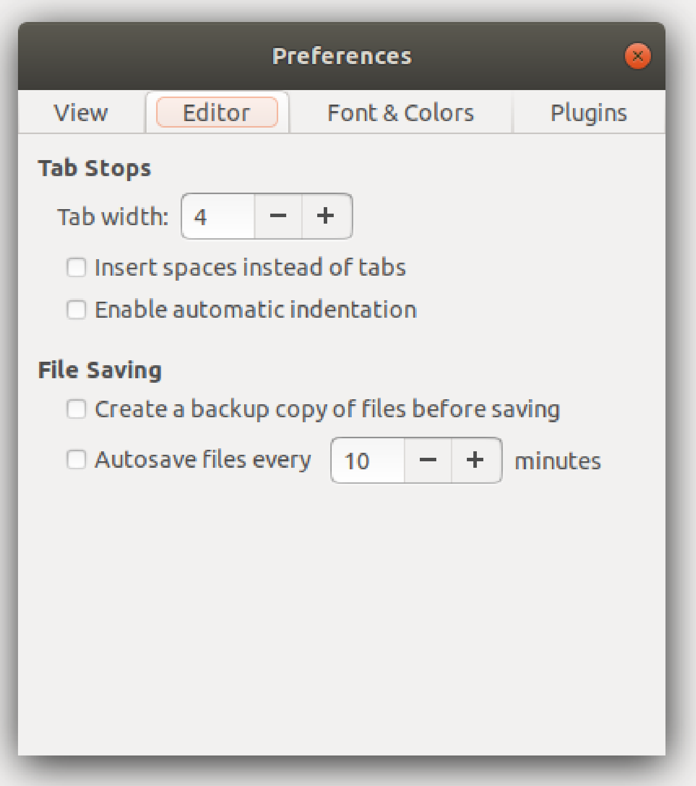
Matériels pédagogiques pour des TDs et TPs, peuvent contenir des erreurs intentionnelles et des termes non définis. À utiliser avec prudence.

Séance 1

- Faire une analyse (sur un papier ou un document):
  - Les attributes de la structure zpixel
    - type de donnée + nom
  - Les fonctions
    - output + nom + (type de donnée + nom, ...)
- Dossier: projetPCA
  - zpixel.c
  - zpixel.h
  - testzpixel.c (main)
  - gcc zpixel.c testzpixel.c -lm -o testzpixel
- Implanter et tester fonction par fonction

Makefile

```
CC=gcc
LIBS=-lm

testzpixel:
	$(CC) testzpixel.c zpixel.c $(LIBS) -o testzpixel
```



Tab Stops

Tab width: 4

□ Insert spaces instead of tabs

□ Enable automatic indentation

File Saving

□ Create a backup copy of files before saving

□ Autosave files every 10 minutes

```
//Créer un zpixel avec une position et une taille
struct zpixel * zpixelCreate(int x, int y, int size);

//Init un zpixel avec une couleur + dégradation
void zpixelInit(struct zpixel * p, int r, int g, int b, double d) ;
void zpixelInitBlack(struct zpixel * p) ;
void zpixelInitWhite(struct zpixel * p) ;

double zpixelLightness(struct zpixel *p) ;
double zpixelSaturation(struct zpixel *p) ;
double zpixelDistance(struct zpixel *p1, struct zpixel *p2) ;
int zpixelProjection(struct zpixel *p, struct image * image);
```

- **Distance =** $\sqrt{(r_1 - r_2)^2 + (g_1 - g_2)^2 + (b_1 - b_2)^2}$

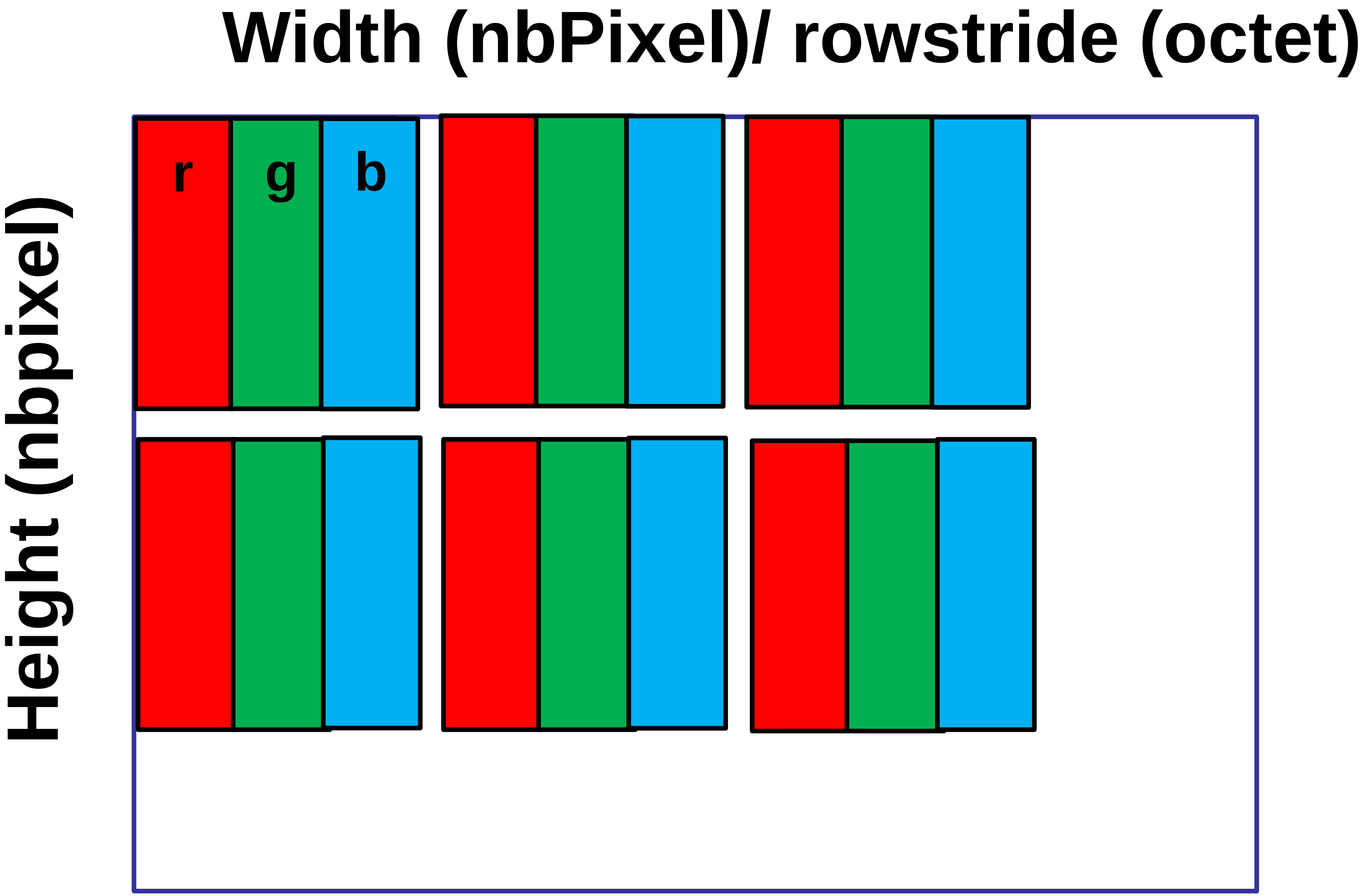
Test

- [Test number] - [Description]
- Input
- Expected Output
- Ouput

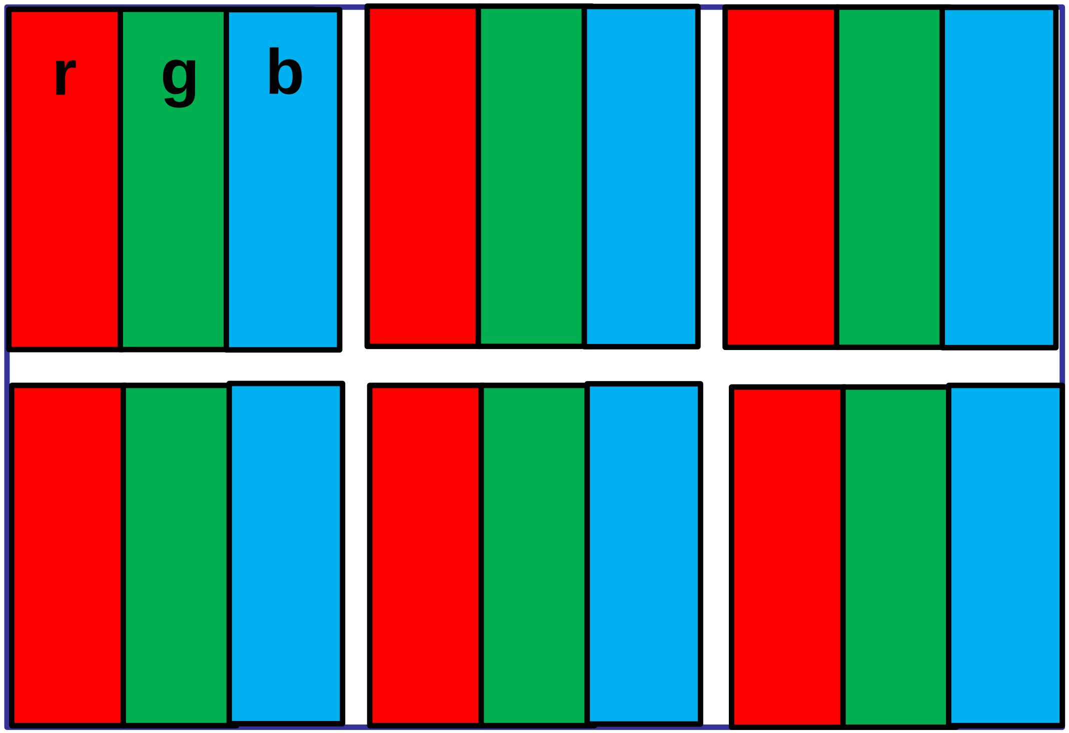
```
Test 01 : Function zpixelCreate
Input: x=4, y=5, size= 10
Expected Output: pixel p with x=4, y=5, size=10
Output: x=4, y=5, size=10
```

```
Test 01 : Function zpixelCreate
Input: x=4, y=5, size= 10
Expected Output: pixel p with x=4, y=5, size=10
Output: x=4, y=5, size=10

---

Test 02: Function zpixelInit
Input: r=1, y=2, b=3
Expected Output: pixel p with r=1, y=2, b=3
Output: r=1, y=2, b=3
```

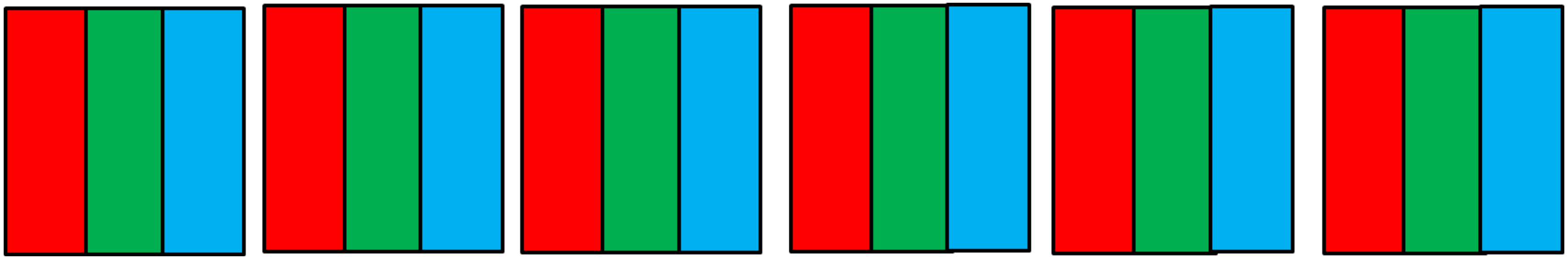
zpixelProjection : Image



zpixelProjection : Image



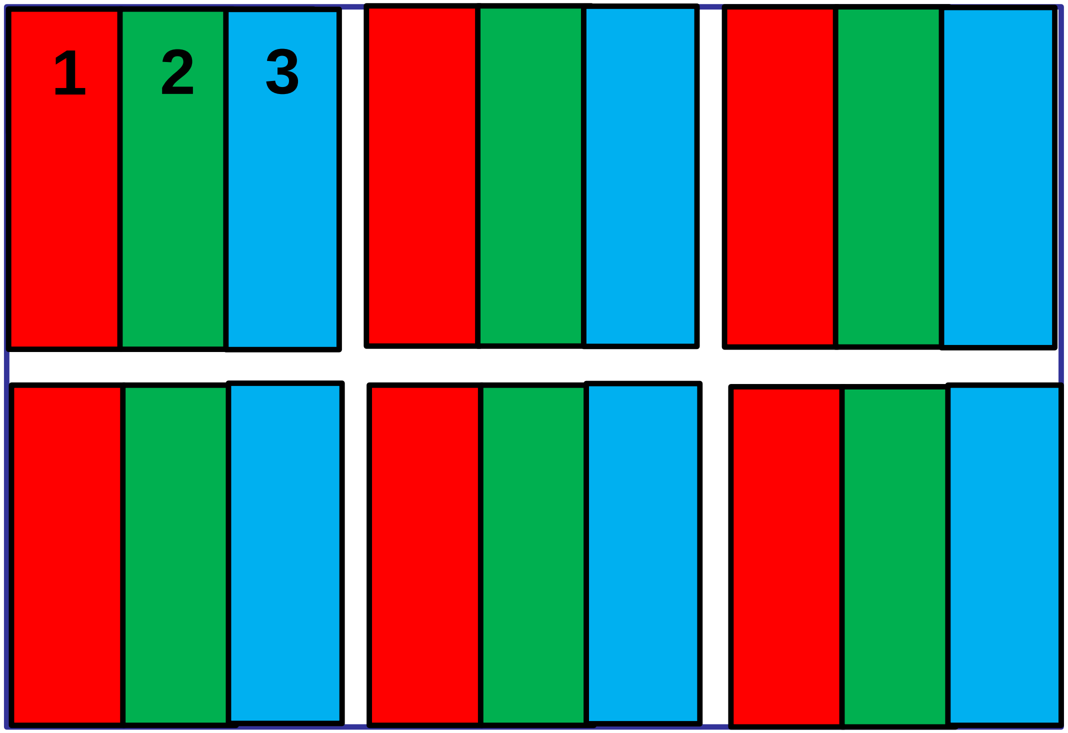
zpixelProjection : Image



r,g,b,r,g,b,r,g,b...

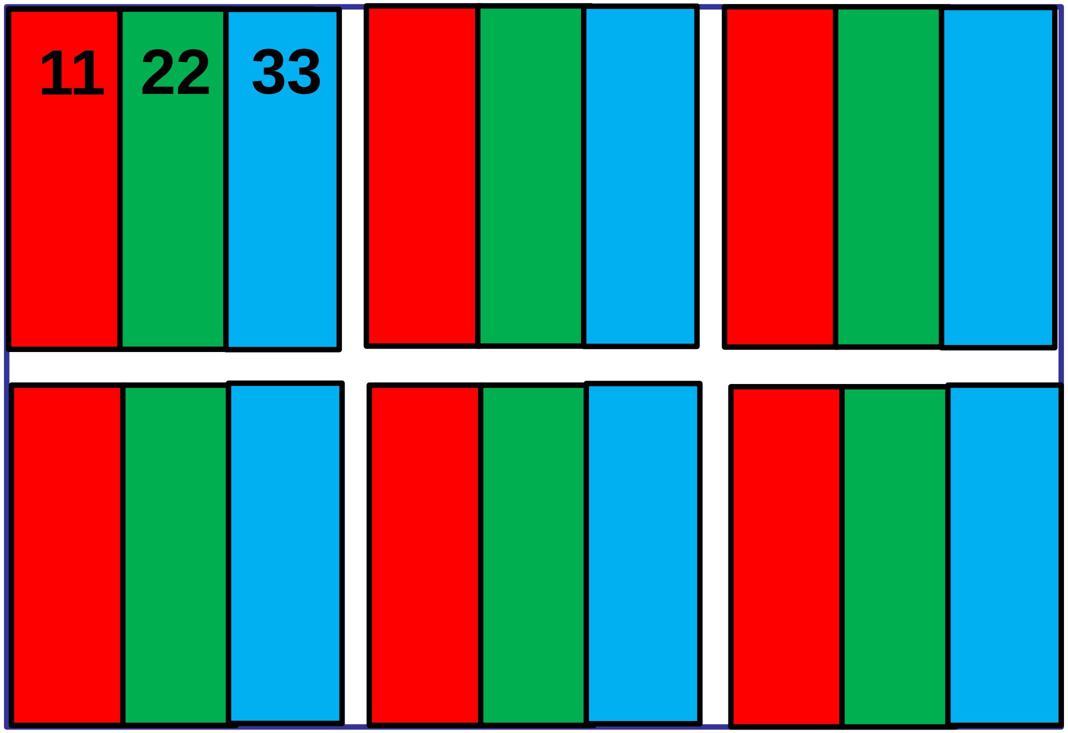
```
struct image {
        int width, height;
        int rowstride;
        unsigned char * pixelBuffer;
};
```

zpixelProjection

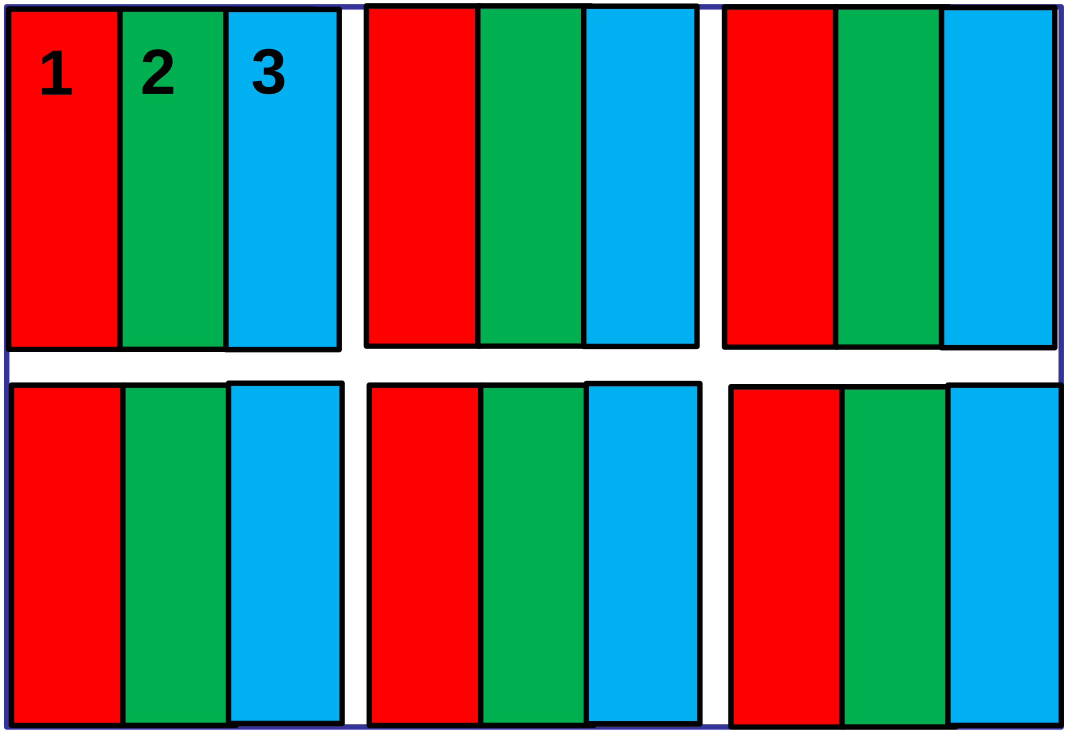


```
struct zpixel * p
p = zpixelCreate(0,0,1)
zpixelInit(p,11,22,33)
zpixelProjection(img,p)
```

zpixelProjection

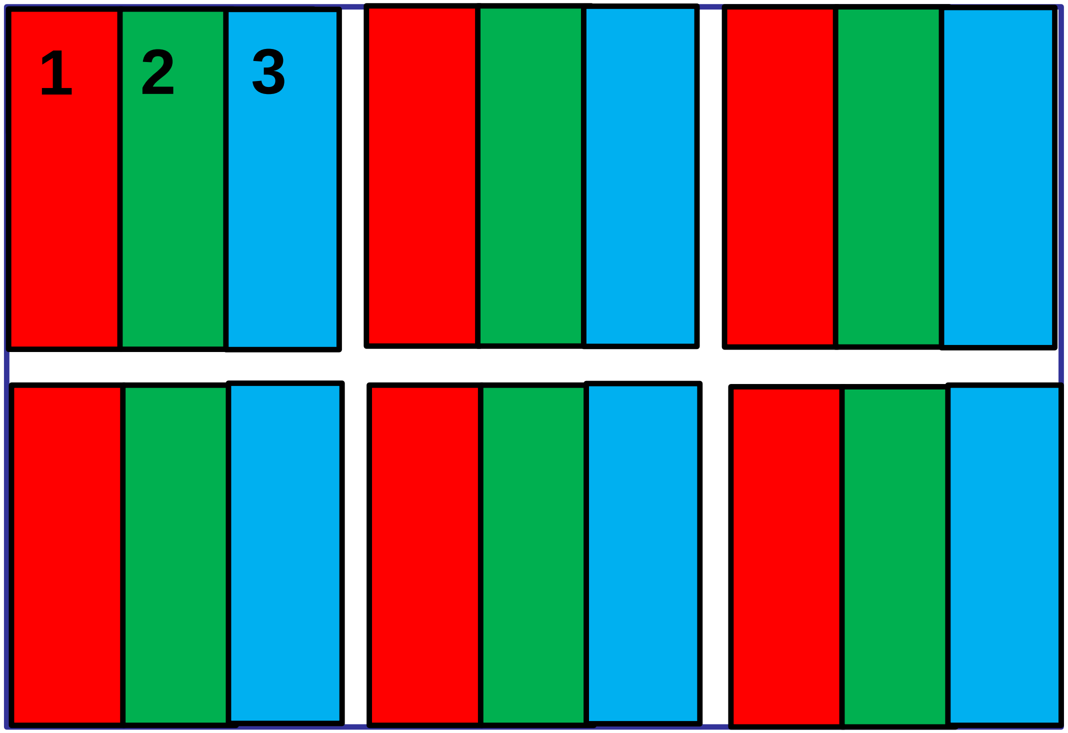


```
struct zpixel * p
p = zpixelCreate(0,0,1)
zpixelInit(p,11,22,33)
zpixelProjection(img,p)
```



?

```
struct zpixel * p
p = zpixelCreate(0,0,2)
zpixelInit(p,11,22,33)
zpixelProjection(img,p)
```



?

```
struct zpixel * p
p = zpixelCreate(1,1,2)
zpixelInit(p,11,22,33)
zpixelProjection(img,p)
```