

Bureau d'étude

TRAITEMENT D'IMAGE

TP 1 : Quantification de la couleur

I. Introduction

Lors de ce TP nous nous proposons de mettre en œuvre et de tester plusieurs méthodes de quantification complémentaires au fur et à mesure. Pour se faire, nous utiliserons les trois mêmes images telles que :



Figure 1 - Image 1



Figure 2 - Image 2



Figure 3 - Image 3

II. Exercice 1

1. Question 1

En utilisant le programme LutSubSamp.m fourni afin de générer automatiquement une LUT à partir d'un paramètre N représentant le nombre de couleurs souhaitées, nous obtenons pour N=8 couleurs la LUT L suivante :

$$L = \begin{bmatrix} 64 & 64 & 64 \\ 192 & 64 & 64 \\ 64 & 192 & 64 \\ 192 & 192 & 64 \\ 64 & 64 & 192 \\ 192 & 64 & 192 \\ 64 & 192 & 192 \\ 192 & 192 & 192 \end{bmatrix}$$

2. Question 2

Nous avons écrit une fonction quantification.m permettant de quantifier une image passée en paramètres en fonction d'une LUT donnée. Nous obtenons, après avoir passé les trois images présentées précédemment, les images quantifiées suivantes :

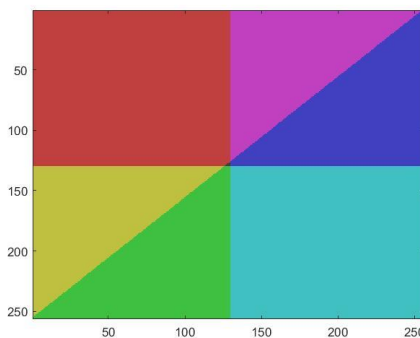


Figure 4 - Image 1 quantifiée avec N = 8

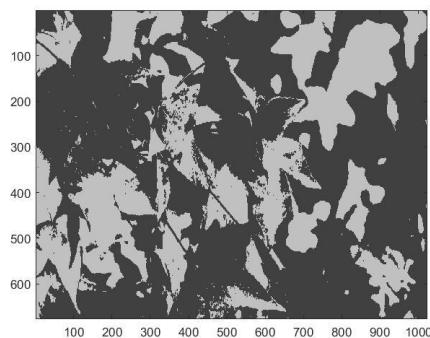


Figure 5 - Image 2 quantifiée avec N = 8

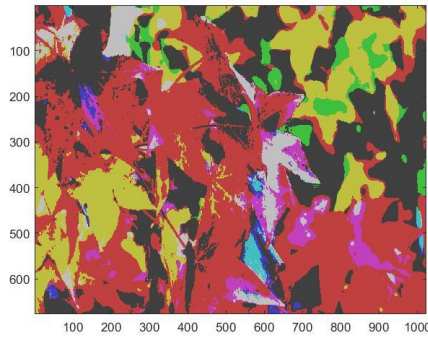


Figure 6 - Image 3 quantifiée avec $N = 8$

3. Question 3

La fonction `quantification.m` renvoie également l'erreur quadratique moyenne de chaque image. Ainsi, nous avons :

`ErreurQuadratiqueMoyenne1` = 1365.6

`ErreurQuadratiqueMoyenne2` = 1643.3

`ErreurQuadratiqueMoyenne3` = 1389.3

4. Question 4

En utilisant les commandes `tic` et `toc`, nous parvenons à avoir le temps de traitement pour chacune des images.

Nous avons donc :

`TempsTraitementImage1` = 0.0749s

`TempsTraitementImage2` = 0.1504s

`TempsTraitementImage3` = 0.0652s

5. Question 5

Le temps de traitement est extrêmement rapide. Les images quantifiées n'ont pas l'air de posséder 8 couleurs différentes.

6. Question 6

A partir de cette question pour cet exercice, nous refaisons les mêmes manipulations en changeant cependant le nombre de couleur. Ce dernier passe de 8 à 32. Ainsi, la nouvelle matrice LUT est telle que :

$$L = \begin{matrix} 64 & 32 & 32 \\ 64 & 96 & 32 \\ 64 & 32 & 96 \\ 64 & 96 & 96 \\ 192 & 32 & 32 \\ 192 & 96 & 32 \\ 192 & 32 & 96 \\ 192 & 96 & 96 \\ 32 & 192 & 32 \\ 96 & 192 & 32 \\ 32 & 192 & 96 \\ 96 & 192 & 96 \\ 32 & 32 & 192 \\ 96 & 32 & 192 \\ 32 & 96 & 192 \\ 96 & 96 & 192 \\ 192 & 160 & 32 \\ 192 & 224 & 32 \\ 192 & 160 & 96 \\ 192 & 224 & 96 \\ 192 & 32 & 160 \\ 192 & 32 & 224 \\ 192 & 96 & 160 \\ 192 & 96 & 224 \\ 32 & 192 & 160 \\ 32 & 192 & 224 \\ 96 & 192 & 160 \\ 96 & 192 & 224 \\ 192 & 160 & 160 \\ 192 & 224 & 160 \\ 192 & 160 & 224 \\ 192 & 224 & 224 \end{matrix}$$

7. Question 7

En utilisant la fonction `quantification.m` afin de quantifier les trois images d'origine avec cette fois 32 couleurs, nous obtenons les images suivantes :

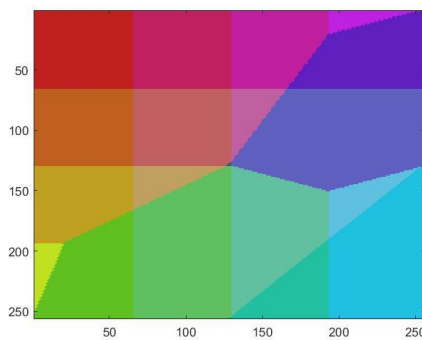


Figure 7 - Image 1 quantifiée avec $N = 32$

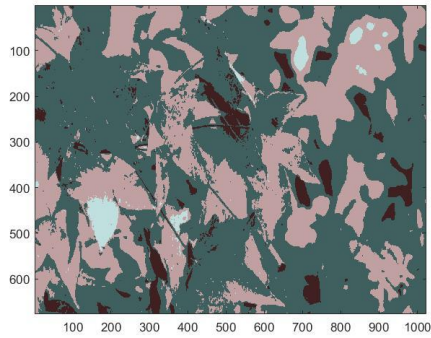


Figure 8 - Image 2 quantifiée avec $N = 32$

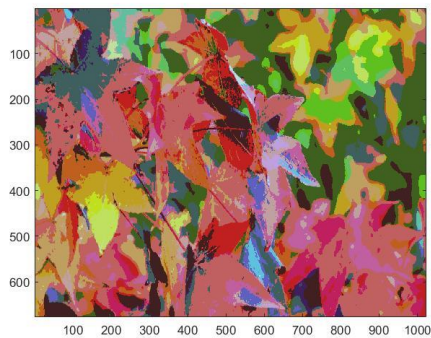


Figure 9 - Image 3 quantifiée avec $N = 32$

8. Question 8

Les erreurs quadratiques moyennes obtenues avec une LUT de 32 couleurs sont :

$\text{ErreurQuadratiqueMoyenne1} = 633.3743$

$\text{ErreurQuadratiqueMoyenne2} = 766.3228$

$\text{ErreurQuadratiqueMoyenne3} = 784.7195$

9. Question 9

Le temps de traitement pour chaque image est cette fois-ci tel que :

$\text{TempsTraitementImage1} = 0.1448\text{s}$

$\text{TempsTraitementImage2} = 0.1393\text{s}$

$\text{TempsTraitementImage3} = 0.1111\text{s}$

10. Question 10

La quantification s'est faite correctement. Les images ne possèdent plus qu'un nombre limité de couleurs présentes dans la LUT et étant le plus proche de celle de l'image d'origine.

III. Exercice 2

11. Question 1

Pour cette question nous avons d'abord crée une LUT avec 2^{15} couleurs. Nous quantifions nos images une première fois avec cette LUT avant de sélectionner les N couleurs les plus présentes dans chaque image afin de créer une nouvelle LUT pour chacune des images. Les images quantifiées avec cette première LUT sont telles que :

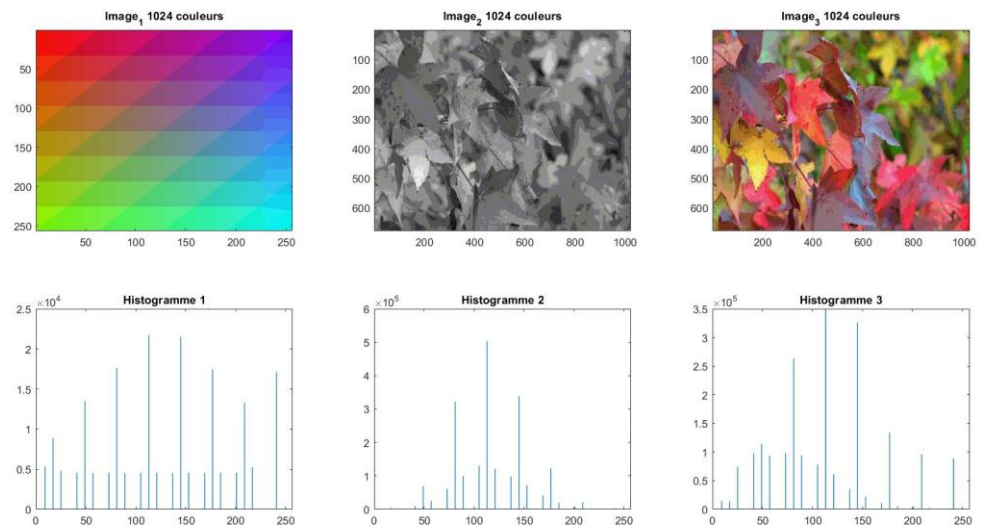


Figure 10 - Images quantifiées avec 2^{15} couleurs

Par la suite, pour $N=8$, nous avons les trois LUT suivantes pour chacune des trois images allant respectivement de 1 à 3.

LUT1=

228	4	52
140	228	4
140	4	228
244	20	4
244	4	20
132	244	4
132	4	244
252	4	4

LUT2=

84	84	84
140	140	140
132	132	132
92	92	92
124	124	124
100	100	100

116	116	116
108	108	108

LUT3=

148	84	92
124	92	108
148	108	124
148	116	124
140	84	92
140	92	108
148	100	116
140	100	116

Pour N=32, nous avons les trois LUT suivantes pour chacune des trois images allant respectivement de 1 à 3.

LUT1=

12	244	244
188	132	4
180	148	4
172	164	4
164	180	4
188	4	132
180	4	148
172	4	164
164	4	180
220	68	4
212	84	4
204	100	4
196	116	4
220	4	68
212	4	84
204	4	100
196	4	116
156	196	4
148	212	4
156	4	196
148	4	212
236	36	4
228	52	4
236	4	36
228	4	52
140	228	4
140	4	228
244	20	4
244	4	20
132	244	4
132	4	244
252	4	4

LUT2=

4	4	4
252	252	252
244	244	244
12	12	12
236	236	236
20	20	20
220	220	220
228	228	228
212	212	212
28	28	28
36	36	36
204	204	204
196	196	196
44	44	44
188	188	188
52	52	52
180	180	180
60	60	60
172	172	172
164	164	164
68	68	68
156	156	156
76	76	76
148	148	148
84	84	84
140	140	140
132	132	132
92	92	92
124	124	124
100	100	100
116	116	116
108	108	108

LUT3=

108	84	100
124	92	100
100	68	84
140	84	84
148	60	44
132	84	92
140	100	124
252	84	84
148	100	108
132	108	132
140	108	124
148	52	44
252	76	76

140	108	116
140	76	84
148	92	100
148	92	108
132	100	116
140	92	100
132	92	108
140	100	108
252	92	124
116	84	100
148	108	116
148	84	92
124	92	108
148	108	124
148	116	124
140	84	92
140	92	108
148	100	116
140	100	116

12. Question 2

Les trois images de début quantifiées avec les nouvelles LUT de 8 couleurs sont :

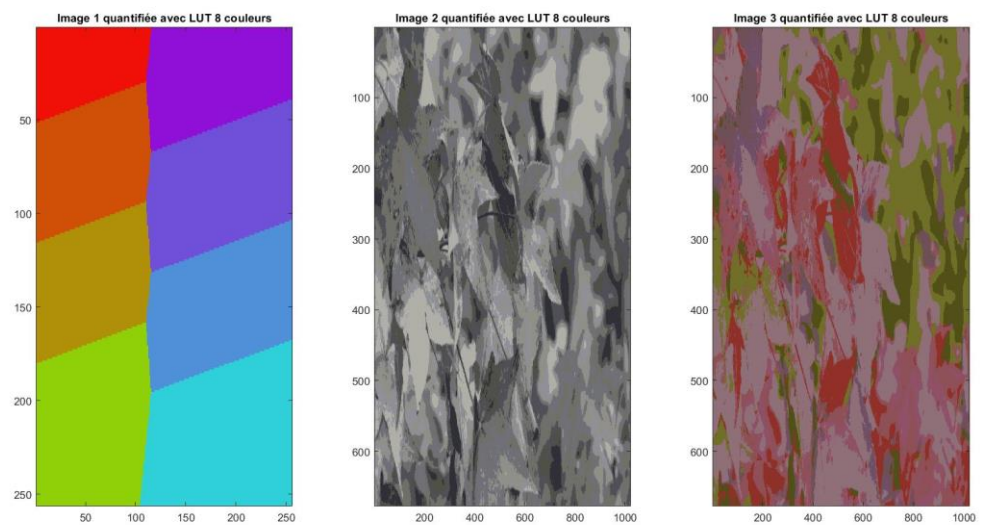


Figure 11 - Images de début quantifiées avec 8 couleurs

Les trois images de début quantifiées avec les nouvelles LUT de 32 couleurs sont :

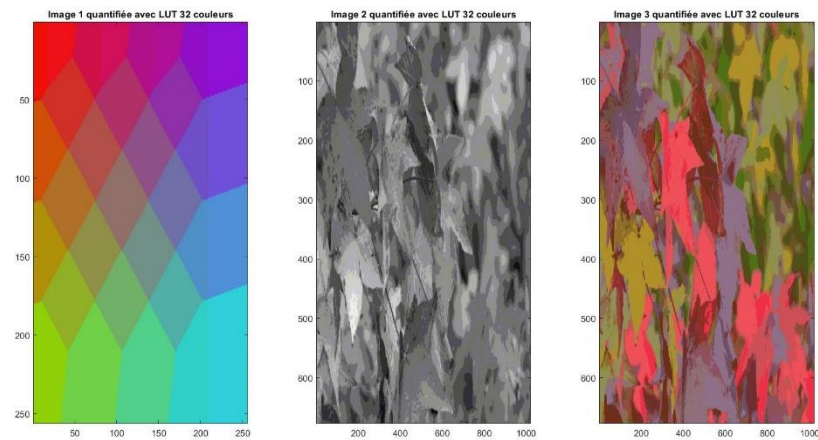


Figure 12 - Images de début quantifiées avec 32 couleurs

13. Question 3

Les erreurs quadratiques moyennes obtenues pour la quantification des images avec une LUT de 8 couleurs sont :

$$Eqm1 = 1.3463e+03$$

$$Eqm2 = 90.1467$$

$$Eqm3 = 1.2061e+03$$

Les erreurs quadratiques moyennes obtenues pour la quantification des images avec une LUT de 32 couleurs sont :

$$Eqm1 = 227.1121$$

$$Eqm2 = 63.7573$$

$$Eqm3 = 452.7086$$

14. Question 4

Les temps de traitement pour la création d'une première LUT de 2^{10} , la quantification d'une image avec cette LUT puis la création d'une nouvelle LUT de 8 couleurs et la quantification de l'image de base avec cette dernière sont

$$\text{TempsTraitementImage1} = 22.3878s$$

$$\text{TempsTraitementImage2} = 66.8108s$$

$$\text{TempsTraitementImage3} = 69.5535s$$

Les temps de traitement pour la création d'une première LUT de 2^{10} , la quantification d'une image avec cette LUT puis la création d'une nouvelle LUT de 32 couleurs et la quantification de l'image de base avec cette dernière sont

$$\text{TempsTraitementImage1} = 20.5204s$$

$$\text{TempsTraitementImage2} = 66.5926s$$

$$\text{TempsTraitementImage3} = 69.1703s$$

15. Question 5

Nous pouvons tout d'abord remarquer que le temps de traitement est beaucoup plus important. C'est avant tout dû à la création de la première LUT à 2^{10} couleurs. Nous pouvons également observer une diminution de l'erreur quadratique moyenne pour chacune des images.

IV. Exercice 3

16. Question 1

Les images quantifiées pour une LUT à 8 couleurs sont telles que :

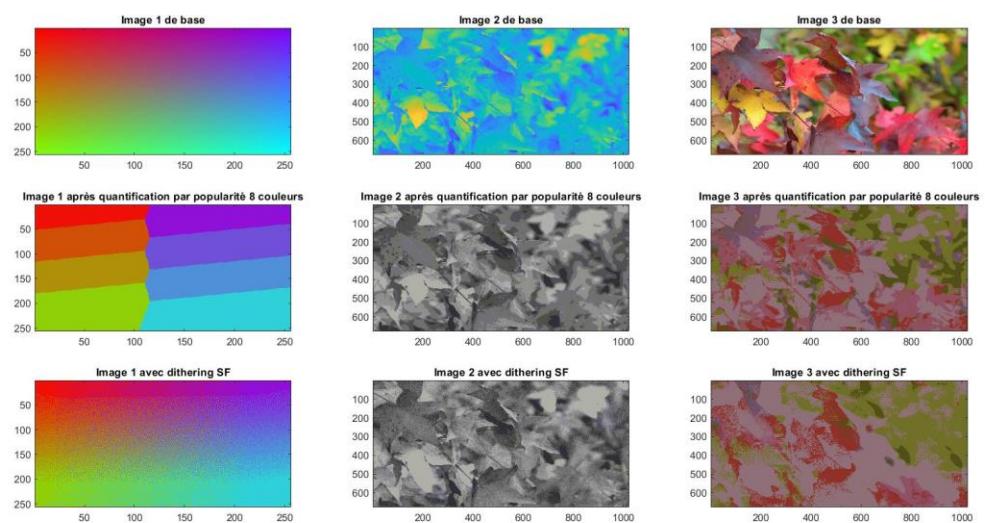


Figure 13 - Ditherting avec une LUT 8 couleurs

Les images quantifiées pour une LUT à 32 couleurs sont telles que :

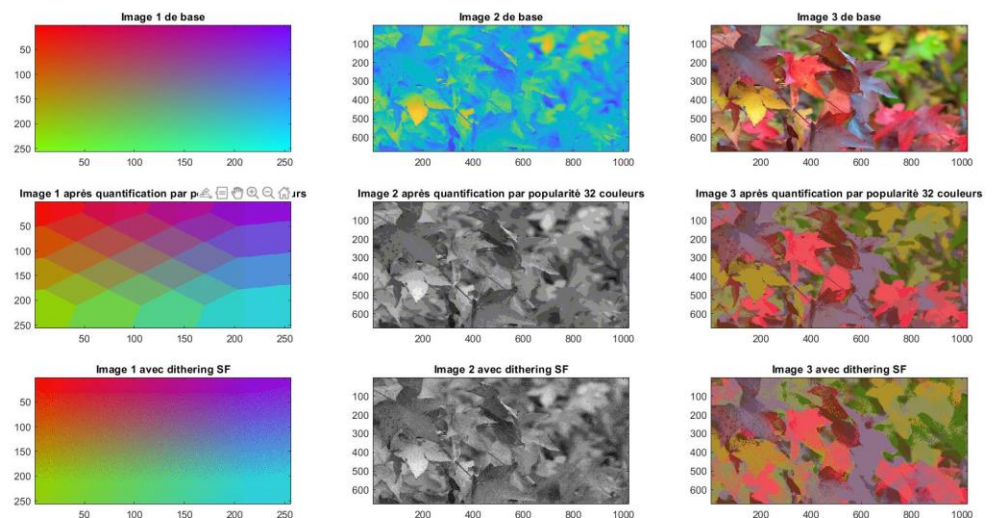


Figure 14 - Ditherting avec une LUT 32 couleurs

17. Question 2

Les erreurs quadratiques moyennes en utilisant une LUT 8 couleurs sont :

$$Eqm1 = 1.7755e+05$$

$$Eqm2 = 2.5297e+05$$

$$Eqm3 = 9.1899e+07$$

Les erreurs quadratiques moyennes en utilisant une LUT 8 couleurs sont :

$$Eqm1 = 1.7164e+05$$

$$Eqm2 = 109.4751$$

$$Eqm3 = 7.0191e+06$$

18. Question 3

Pour les temps, nous ne prenons pas en compte ce qui a été fait lors de l'exercice 2.

Les temps de traitement en utilisant une LUT de 8 couleurs sont :

$$\text{TempsTraitementImage1} = 0.0963s$$

$$\text{TempsTraitementImage2} = 0.2597$$

$$\text{TempsTraitementImage3} = 0.2112s$$

Les temps de traitement en utilisant une LUT de 8 couleurs sont :

$$\text{TempsTraitementImage1} = 0.1158s$$

$$\text{TempsTraitementImage2} = 0.3392$$

$$\text{TempsTraitementImage3} = 0.2973s$$

19. Question 4

Nous remarquons que les erreurs quadratiques moyennes sont beaucoup plus importantes que celles calculées lors des deux premiers exercices. Le temps de traitement est légèrement plus lent que lors de la classification de l'exercice 1.

TP 2 : Fonctions de transfert

I. Introduction

Lors de ce TP nous nous proposons de mettre en œuvre et de tester différentes méthodes d'amélioration d'images en niveaux de gris. Pour se faire, nous utiliserons les trois images suivantes afin de tester nos différents algorithmes :



Figure 15 - Image 1



Figure 16 - Image 2



Figure 17 - Image 3

II. Comparaison d'algorithmes

Dans cette partie nous nous proposons de comparer deux algorithmes de prétraitement. L'égalisation d'histogramme ainsi que l'expansion de la dynamique.

1. Affichage d'histogrammes

Ici, nous décidons d'afficher les différents histogrammes des trois images introduites précédemment.

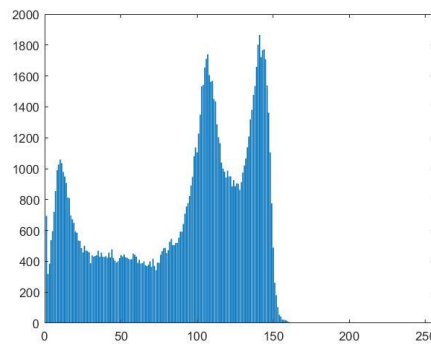


Figure 18 - Histogramme de l'image 1

L'image 1 possède un histogramme qui s'étend de l'intensité 0 à l'intensité 150. Cette image est plutôt nette mais ses couleurs sont plutôt sombres délaissant les intensités plus claires

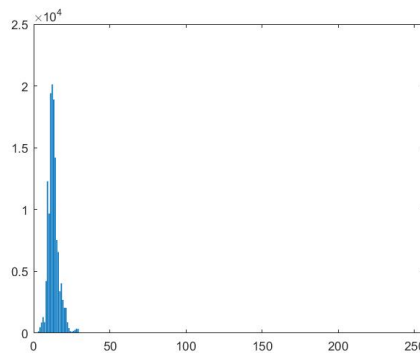


Figure 19 - Histogramme de l'image 2

L'image 2 possède un histogramme où les intensités sont concentrées entre 0 et 30. Cette image est donc extrêmement sombre où tout apparaît noir à l'œil nu.

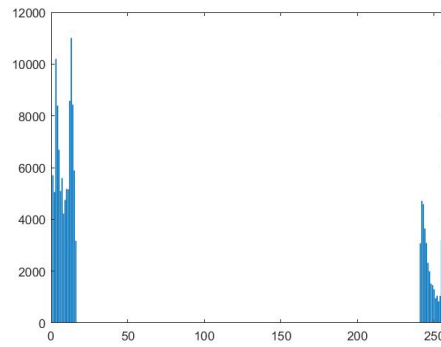


Figure 20 - Histogramme de l'image 3

L'image 3 possède des intensités très sombres (entre 0 et 15) et très claires (entre 240 et 255) sans entre deux. Il s'agit de la moins bonne image parmi les 3 car elle n'offre pas la possibilité de s'étaler sur l'ensemble des intensités.

2. Expansion de la dynamique

Les images obtenues après l'application d'un algorithme d'expansion de la dynamique sont :



Figure 21 - images après expansion de la dynamique

Les histogrammes correspondant à ces images sont :

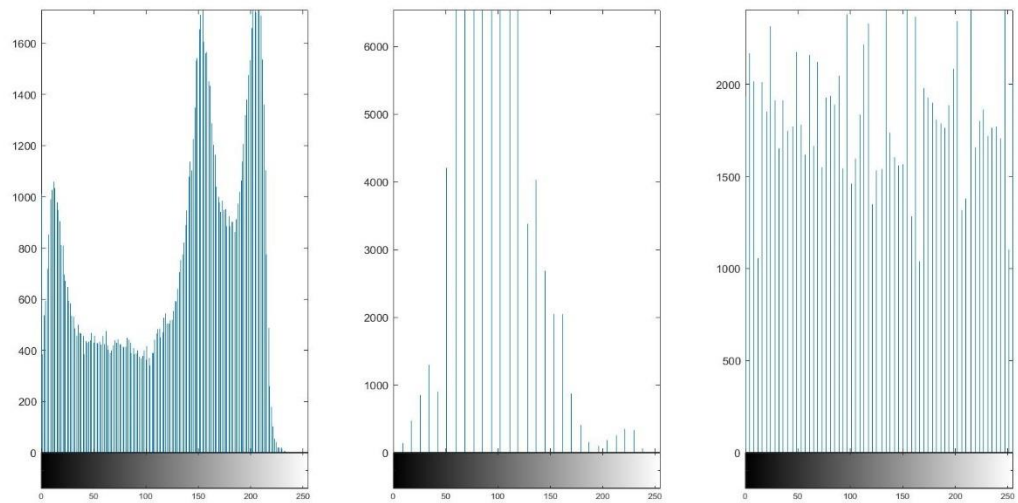


Figure 22 - Histogrammes après expansion de la dynamique

Après expansion de la dynamique, nous pouvons voir que les valeurs d'intensité sont réparties sur la totalité de l'ensemble $[0,255]$ au lieu d'être regroupés comme avant. Cela résulte sur des images plus claires, surtout pour la deuxième image. Cependant, cela n'a pas eu trop d'effet sur la dernière image car celle-ci était composée de deux ensembles beaucoup trop étroits au départ.

3. Egalisation d'histogramme

Les images obtenues après l'application d'une égalisation d'histogramme sont :

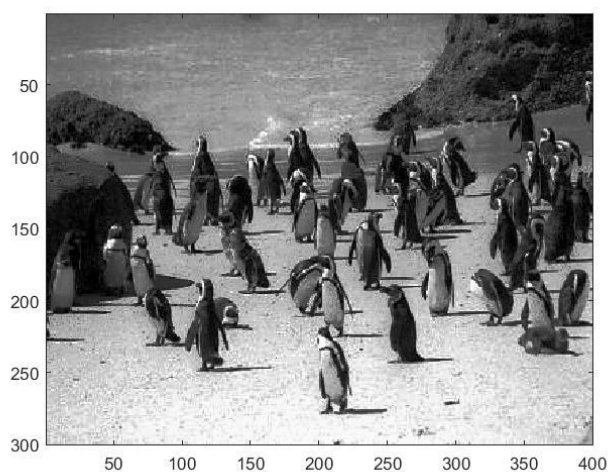


Figure 23 - Image 1 après égalisation d'histogramme

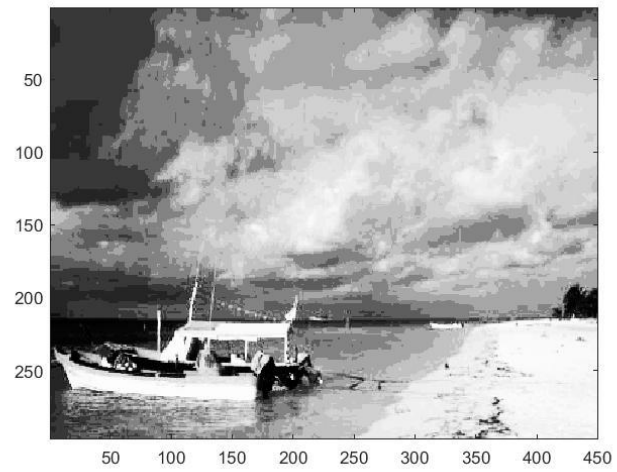


Figure 24- Image 2 après égalisation d'histogramme

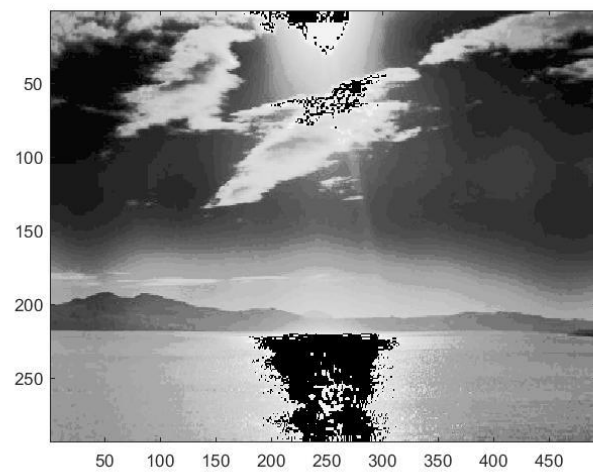


Figure 25 - Image 3 après égalisation d'histogramme

Les histogrammes correspondant à ces images sont :

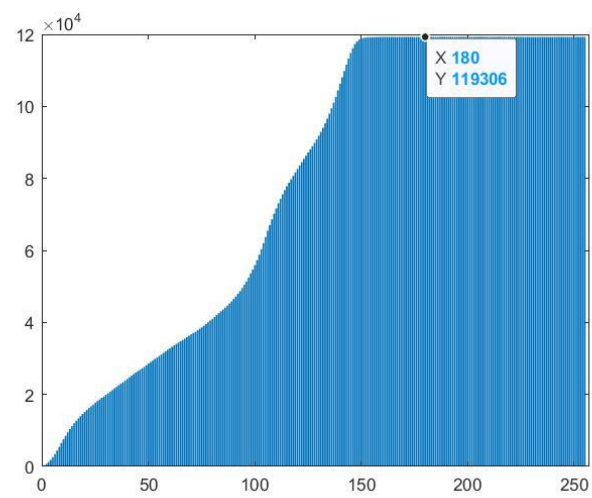


Figure 26 - Histogramme de l'image 1 après égalisation d'histogramme

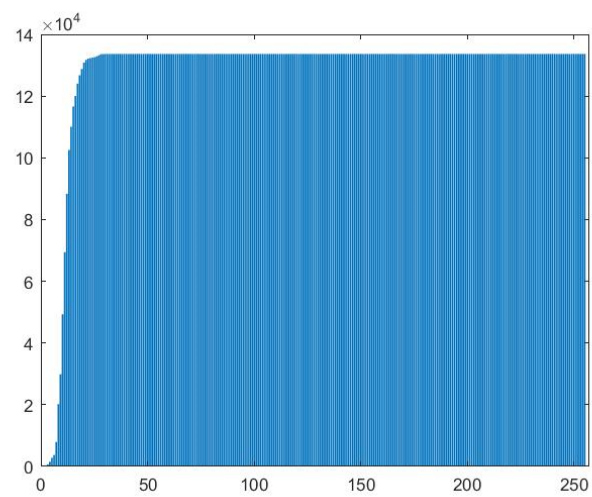


Figure 27 - Histogramme de l'image 2 après égalisation d'histogramme

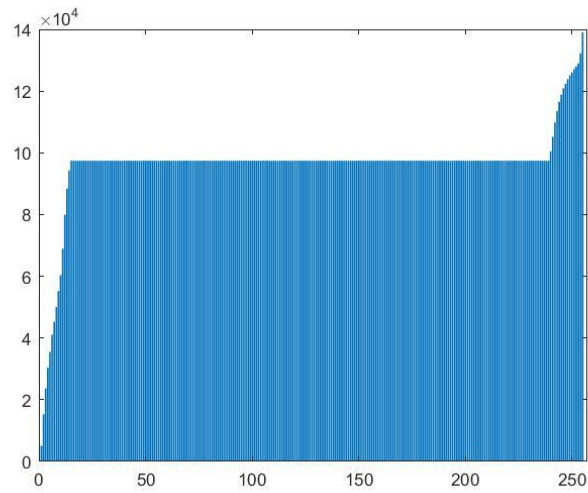


Figure 28 - Histogramme de l'image 3 après égalisation d'histogramme

III. Table de couleur

1. Sous-programme transfert

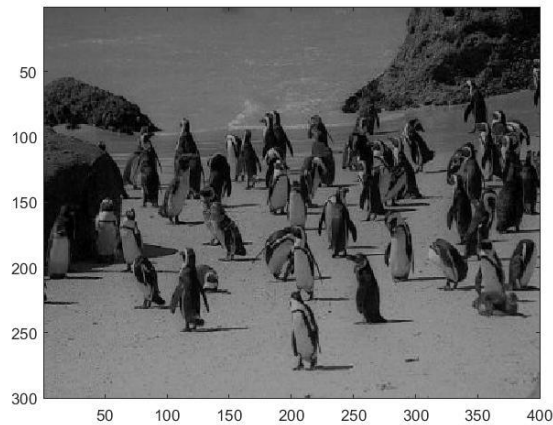


Figure 23 - Image avant transfert



Figure 4 - Image après transfert

La fonction transfert permet de créer une nouvelle couleur map en fonction de l'intervalle donné en paramètres. Ainsi, transfert crée une colormap de 255 lignes ou chaque colonne d'une ligne possède la même valeur allant de 0 à 255. Ainsi on crée des nuances de gris allant du blanc au noir dans une matrice 255x3.

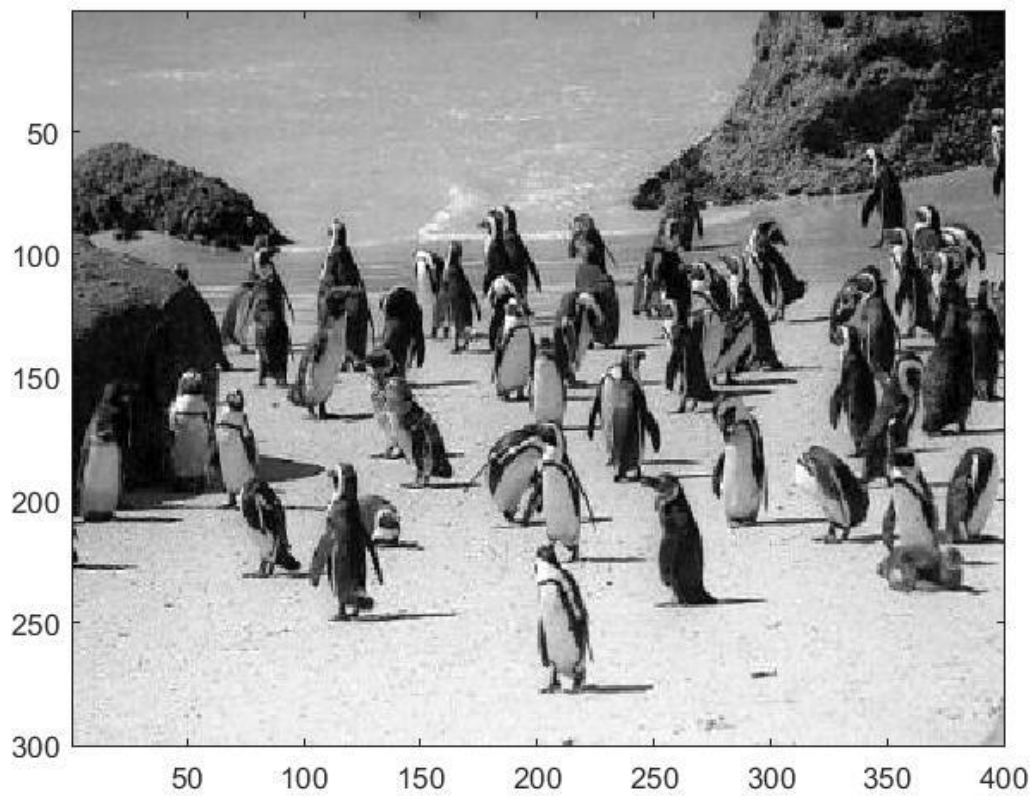
2. Traitement réalisé par la séquence indiquée

On applique une correction gamma.

3. Modification de la séquence pour faire une expansion de la dynamique

La modification apportée est :

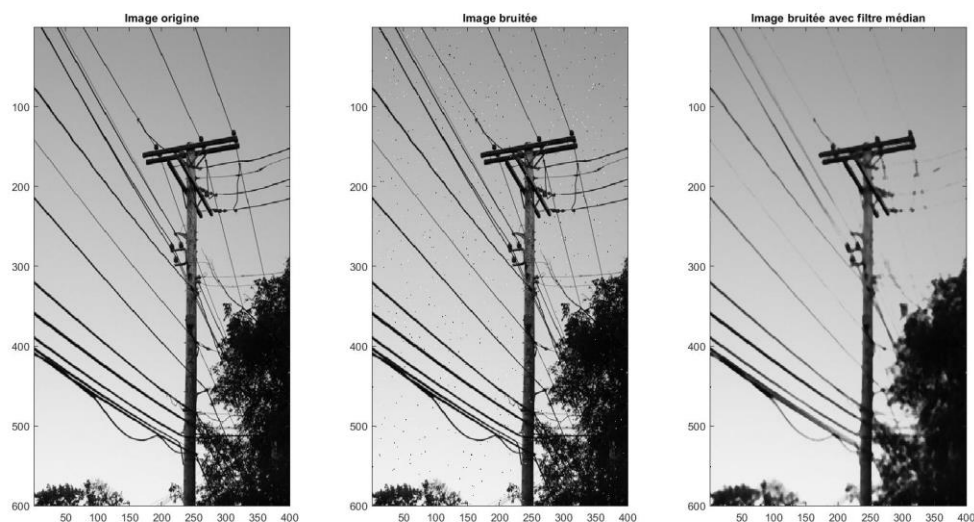
```
palette=uint8((double([0:255]') ./155) .*255);
```



TP 3 : Filtre médian et déconvolution

I. Filtre médian

Sur l'image suivante vous pouvez voir l'image fourni pour ce TP, l'image que l'on a bruitée (avec $N=500$) et enfin cette dernière avec un filtre médian appliqué.



L'erreur quadratique moyenne entre l'image bruitée et l'image de départ est : 89.7023

L'erreur quadratique moyenne entre l'image filtrée et l'image de départ est : 531.6034

Le filtre devient performant à partir de $N=3000$.

Le graphe de progression des deux erreurs quadratiques moyennes en fonction de N est :

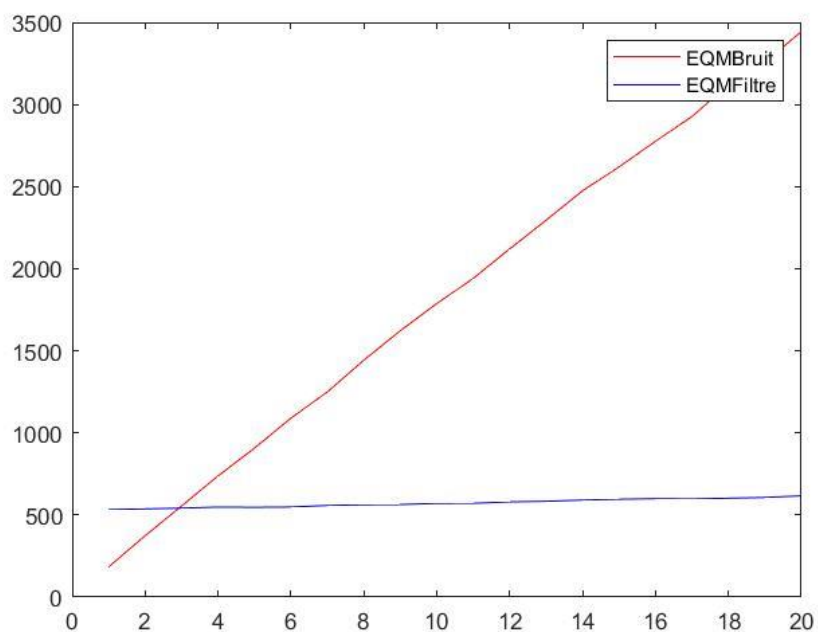
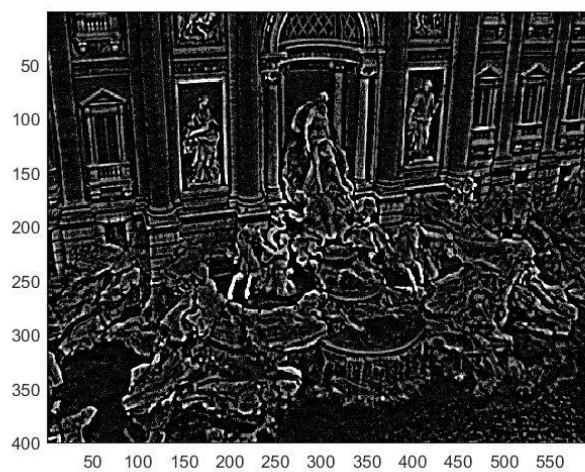
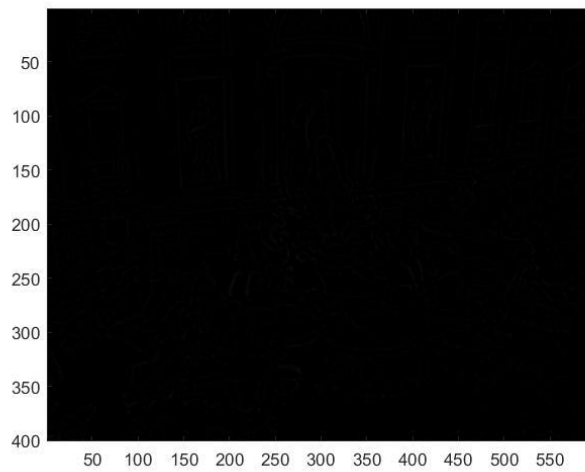
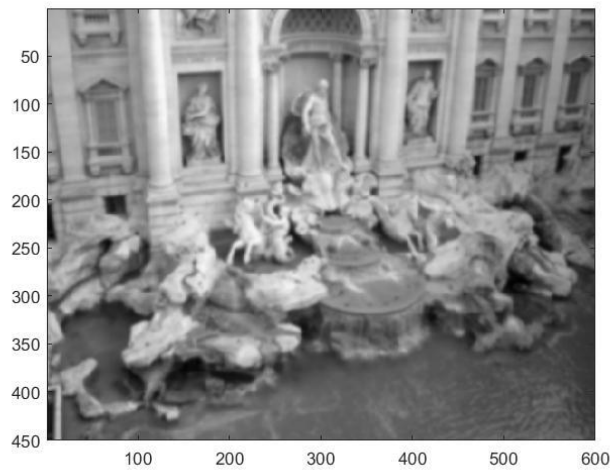


Figure 29 - Progression des deux erreurs quadratiques moyennes

II. Masque flou

Pour un coefficient de 50, les étapes intermédiaires sont telles que :





Nous testons maintenant différents coefficients.

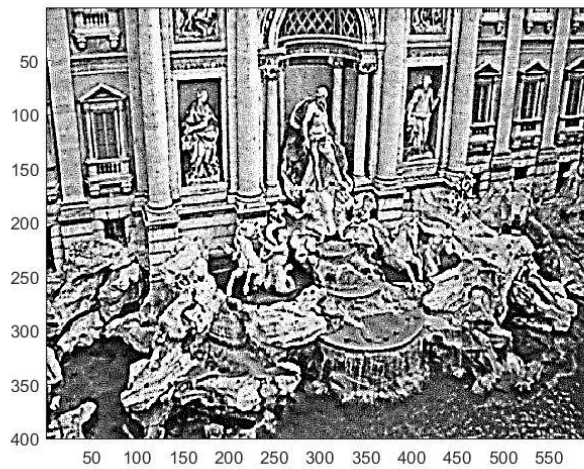
Coef = 10 :



Coef = 50 :

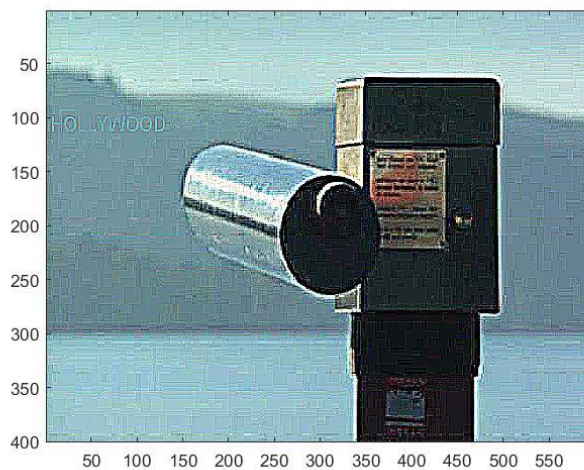


Coef = 100 :

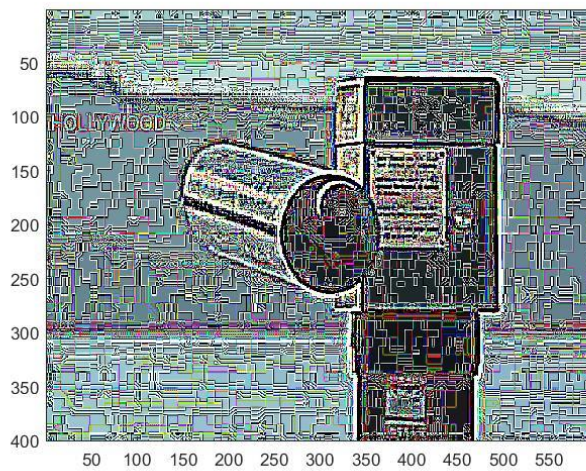


III. Masque flou sur une image couleur

Pour un coefficient de 50 nous obtenons l'image suivante.



Avec un coefficient de 500 nous obtenons l'image suivante :



L'application d'un masque flou permet ici d'observer le signe « Hollywood » qui n'était pas visible sur l'image d'origine.

TP 4 : Algorithme de la bulle

I. Gradient

L'image générée est :

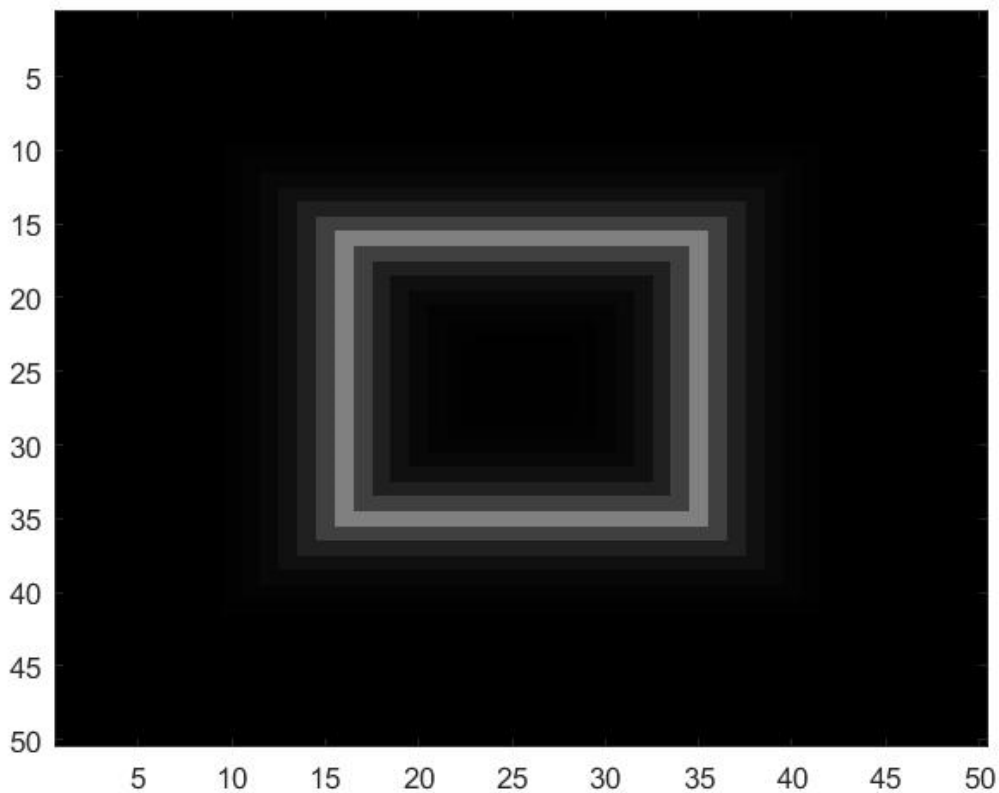


Figure 30 - Image générée par le code fourni

II. Adaptation de l'algorithme de la bulle

1. Algorithme de la tortue

La fonction tortue possède pour paramètres de sortie « contour » ainsi que « nbpoints ».

Contour est un tableau contenant les coordonnées des points représentant les contours et nbpoints est le nombre de points formant le contour. Pour visualiser le fonctionnement de la fonction nous utilisons une image toute noire avec un bloc blanc au milieu dont les coordonnées en x et y vont de 24 à 26.

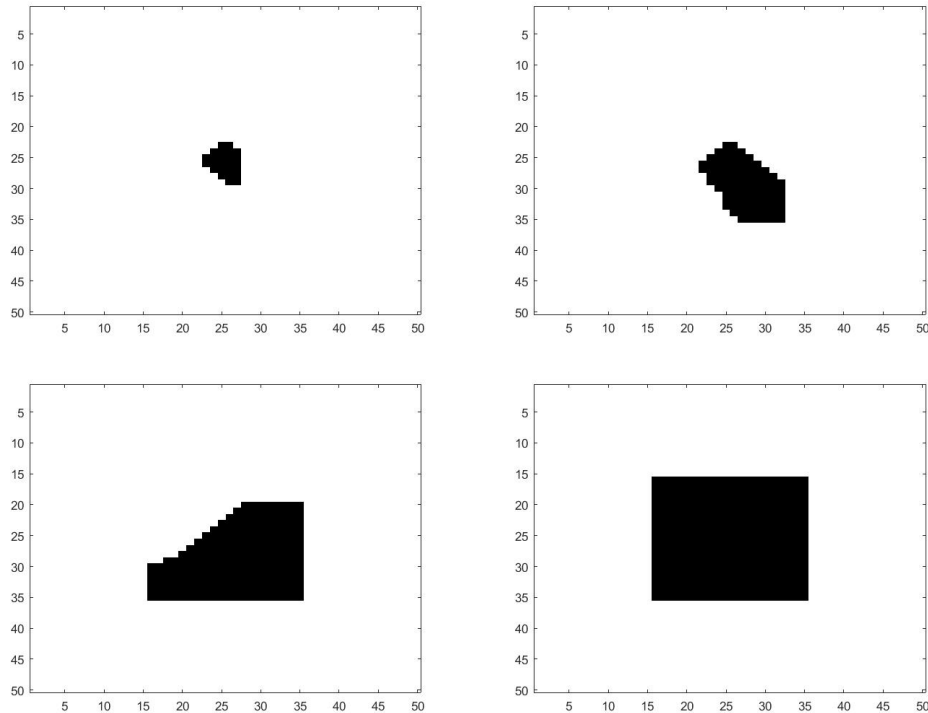
La variable contour créée est tel que :

contours			
8x2 double			
	1	2	3
1	24	24	
2	24	25	
3	24	26	
4	25	26	
5	26	26	
6	26	25	
7	26	24	
8	25	24	
9			

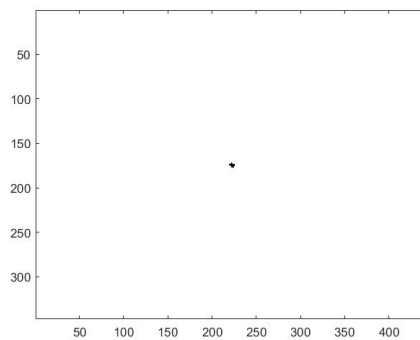
La variable nbpoints a pour valeur 8.

2. Algorithme de la bulle

Vous trouverez ci-dessous quelques étapes de l'évolution de la bulle sur l'image de la questions 1 :



Vous trouverez ci-dessous quelques étapes de l'évolution de la bulle sur l'image du robot :
En commençant vers le centre comme avant on s'arrête à la première itération :

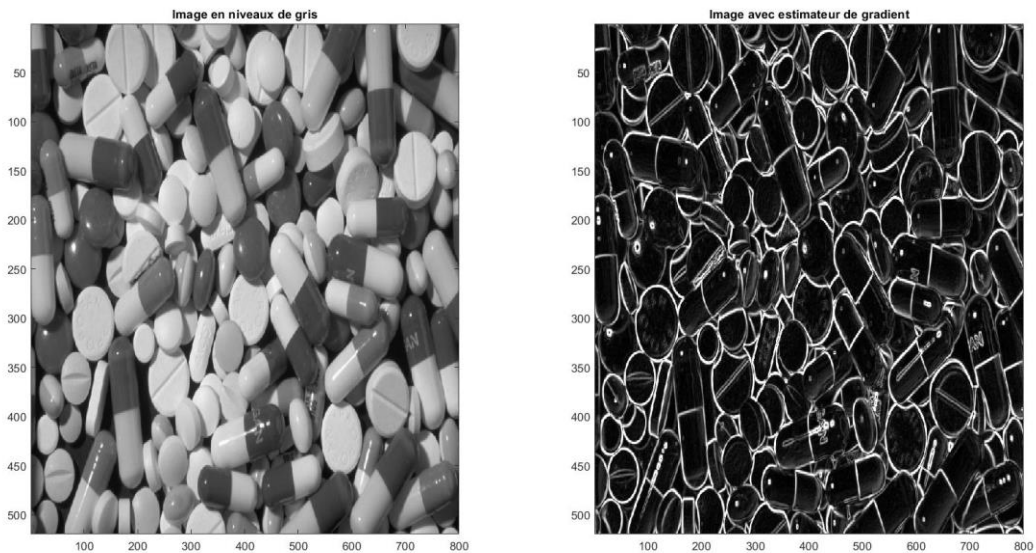


Afin d'améliorer les résultats il faudrait surement appliquer différentes bulles avec des points de départ différents.

TP 5 : Segmentation en régions

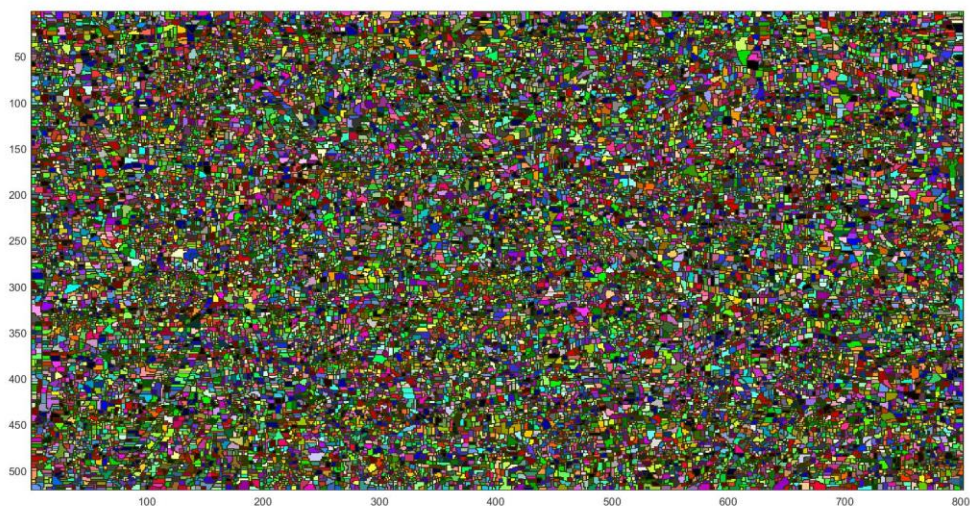
I. Estimation de gradient

L'image obtenue après application d'un estimateur de gradient à l'image de base convertie en niveaux de gris est :



II. Ligne de partage des eaux

L'image obtenue après application de l'algorithme de ligne de partage des eaux est :



Nous avons 802 régions.

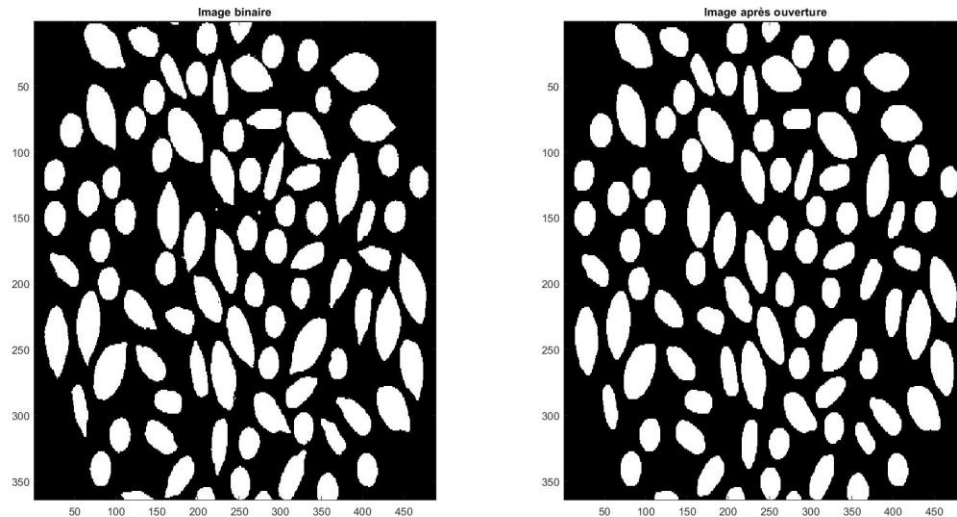
III. Fusion de régions

TODO

TP 7 : Classer la graine

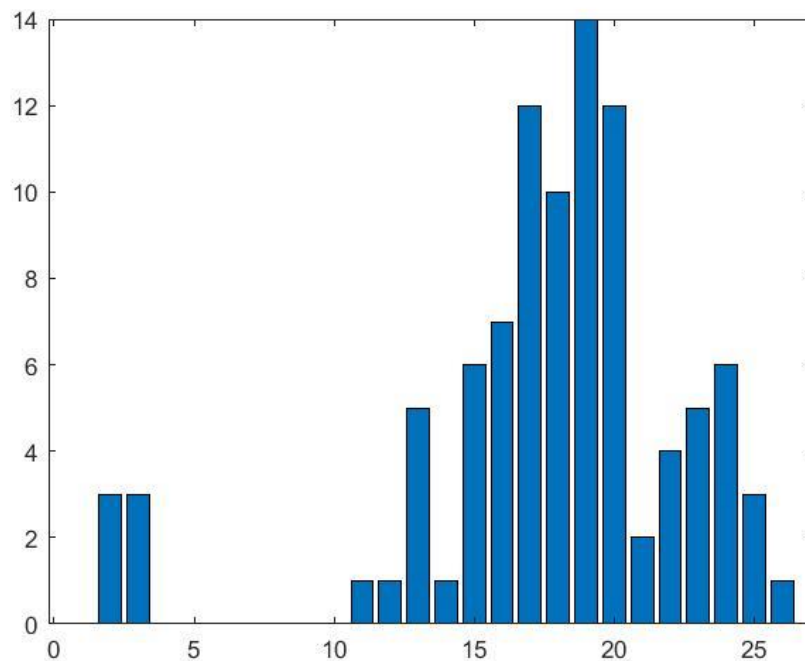
I. Débruitage

Afin d'éliminer le bruit, nous faisons une ouverture, c'est-à-dire une érosion suivie d'une dilatation. Nous avons utilisé un structurant disque de rayon 5. L'image du résultat est :



II. Identification des classes

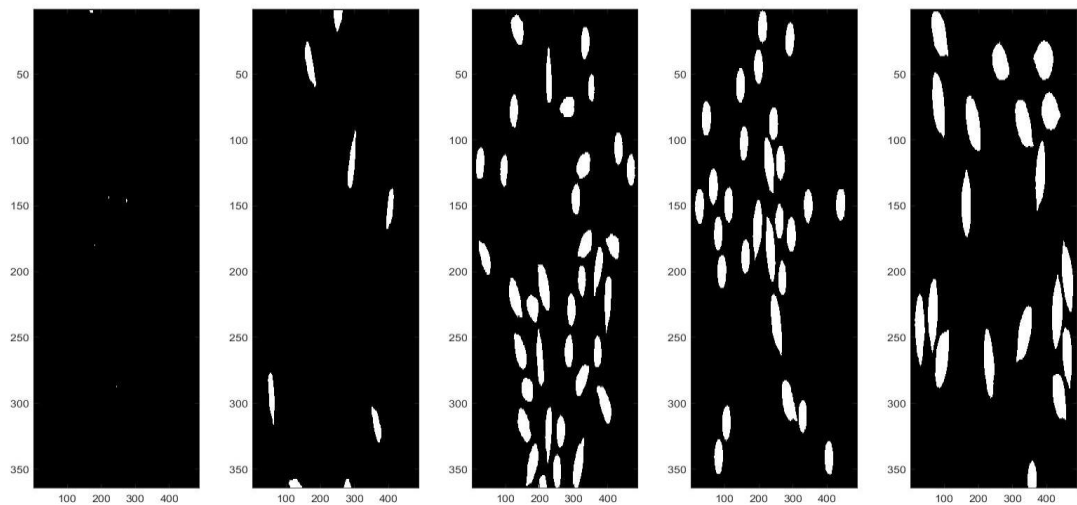
L'histogramme donné par la fonction N est :



Nous pouvons observer 5 minimums locaux. Nous pouvons considérer qu'il y a donc 5 classes. Or, dans l'image de base nous pouvons voir qu'il y a trois graines différentes. La première classe observable dans l'histogramme est due aux graines en bord d'image qui n'apparaissent pas en totalité. La seconde classe en trop est probablement la cause du choix de notre structurant ainsi que des prétraitements que nous avons effectués.

III. Reconstruction dissociée des classes en niveaux de gris

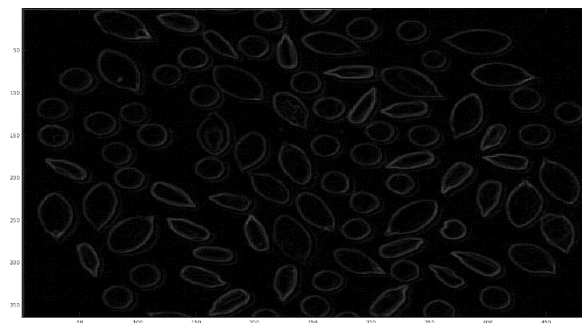
Les images reconstruites ne comportant que des représentants des principales classes de graines sont les suivantes.



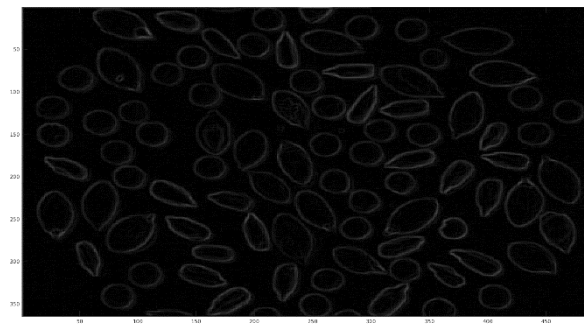
IV. Morphomathématiques en niveaux de gris – Tests sur le gradient

Les trois images de gradients sont :

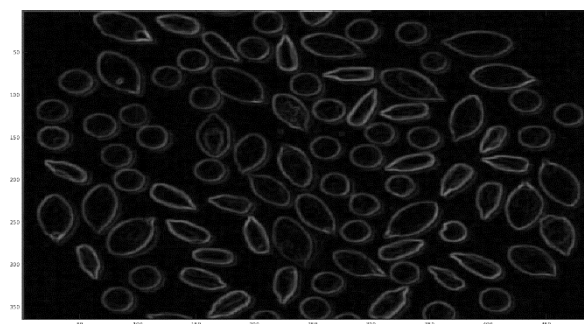
Gi :



Ge :



G :



On voit que les contours des graines sont bien définis.