

Cet article est rédigé par des élèves. Il peut comporter des oublis ou des imperfections, autant que possible signalés par nos relecteurs dans les notes d'édition.

Détection de contours [1]

2014-2015

Nom, prénom et niveaux des élèves : Blanc Charline, Honnorat Pauline, Didouh Yasmine, Goux Manon, classe de seconde

Établissement : Lycée Pasquet, Arles

Enseignant(s) : Sylvie Larras

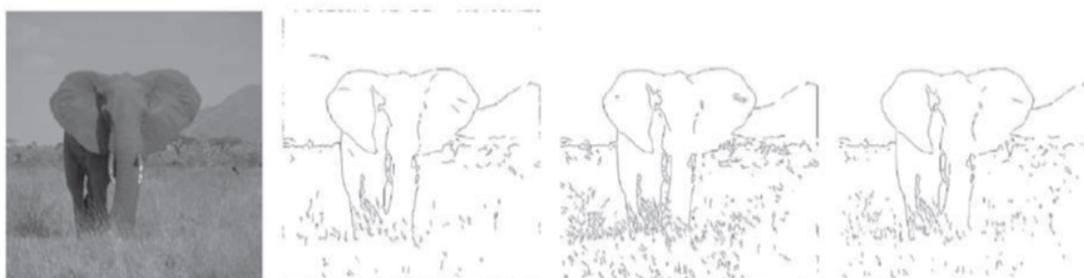
Chercheur(s) : Sophie Viseur, CEREGE, Université d'Aix-Marseille

Table des matières

1	Présentation du sujet	2
2	Annonces des conjectures et résultats obtenus	2
3	Texte de l'article	2
3.1	Qu'est-ce qu'une image numérique ?	3
3.2	Création d'images : transformations de drapeaux	4
3.2.1	Drapeau français	4
3.2.2	Drapeaux en gris	4
3.2.3	Drapeau noir et blanc	5
3.2.4	Encore d'autres drapeaux	6
3.3	Calculs de paramètres statistiques sur des images en niveaux de gris	6
3.3.1	Calculs utilisant les " premiers voisins "	6
3.3.2	Avec une fenêtre mobile ou masque	8
3.4	Application à d'autres images	11
4	Conclusion	11

1 Présentation du sujet

La détection de contours est utilisée dans beaucoup d'applications informatiques (analyse automatique d'images, de vidéos de surveillance, etc.). Elle consiste en la détection des « limites » des objets qui sont contenus dans l'image.



Exemples de détection de contours d'une image en niveau de gris par différentes approches, de gauche à droite : la méthode multi-inhibition proposée par (Papari et al. 2011), le Canny Edge Detector, et la méthode d'inhibition proposée par (Grigorescu et al., 2004).

Idée : Essayer de mettre en place une technique de détection de contours fondée sur l'étude de paramètres statistiques. La question est de savoir quels sont les paramètres statistiques qui peuvent mettre en évidence les contours d'un objet.

2 Annonces des conjectures et résultats obtenus

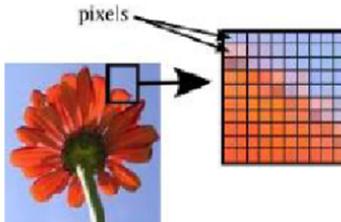
Nous avons utilisé les paramètres statistiques suivants : médiane , moyenne, étendue, variance. Nous pensons que ces paramètres pouvaient permettre de déterminer les contours dans une image.

Les tests effectués sur des images en niveaux de gris montrent que, l'obtention de contours "nets" se fait en utilisant le filtre de la médiane avec un "masque" de grande taille , suivie du filtre de la variance avec un "masque" de petite taille , puis d'un "seuillage".

3 Texte de l'article

Nous avons d'abord commencé par comprendre ce qu'est une image numérique. Puis, nous avons modifié une image à l'aide de calculs avec les pixels qui la composent. Pour cela, nous avons utilisé le logiciel Gimp. Enfin, nous nous sommes intéressés à la détection des contours, nous avons utilisé une "fenêtre mobile" ou "masque". Nous avons utilisé scilab, puis ImageJ pour visualiser les effets de nos calculs.

3.1 Qu'est-ce qu'une image numérique ?

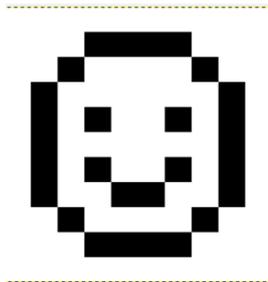


Une image numérique est un ensemble fini de points appelés "pixels" . Une image a pour vocation d'être affichée sur un écran. Chaque pixel possède une couleur.

La définition d'une image numérique correspond au nombre de pixels qui la compose en hauteur et en largeur. La résolution de l'image est définie par un nombre de pixels par unité de longueur.

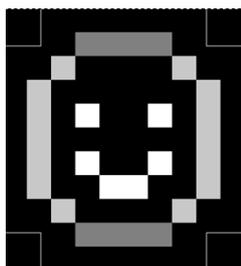
Un format d'image est une représentation informatique de l'image, associée à des informations sur la façon dont l'image est codée et fournissant éventuellement des indications sur la manière de la décoder et de la manipuler.

Le format « PBM » : ce format de fichier est utilisé pour des images noir et blanc. Un pixel noir est codé par un caractère 1, un pixel blanc est codé par un caractère 0.



```
smileynoirblanc.txt - Bloc-notes
Fichier  Edition  Format  Afficha
P1
10 11
0 0 0 0 0 0 0 0 0 0
0 0 0 1 1 1 1 0 0 0
0 0 1 0 0 0 0 1 0 0
0 1 0 0 0 0 0 0 1 0
0 1 0 1 0 0 1 0 1 0
0 1 0 0 0 0 0 0 1 0
0 1 0 1 0 0 1 0 1 0
0 1 0 0 1 1 0 0 1 0
0 0 1 0 0 0 0 1 0 0
0 0 0 1 1 1 1 0 0 0
0 0 0 0 0 0 0 0 0 0
```

Le format « PGM » est un des formats utilisé pour exprimer une image en niveau de gris. Dans ce type de codage, le blanc a pour valeur 255 et le noir 0. Plus un gris sera foncé, plus la valeur associée sera faible.



```
smileygris.txt - Bloc-notes
Fichier  Edition  Format  Affichage  ?
P2
10 11
255
0 0 0 0 0 0 0 0 0 0
0 0 0 128 128 128 128 0 0 0
0 0 200 0 0 0 0 200 0 0
0 200 0 0 0 0 0 200 0
0 200 0 255 0 0 255 0 200 0
0 200 0 0 0 0 0 200 0
0 200 0 255 0 0 255 0 200 0
0 200 0 0 255 255 0 0 200 0
0 0 200 0 0 0 200 0 0
0 0 0 128 128 128 128 0 0 0
0 0 0 0 0 0 0 0 0 0
```

Le format « PPM » : Ce format de fichier est utilisé pour des images couleur. Chaque pixel est codé par trois valeurs (rouge, vert et bleu). Un pixel bleu sera codé : 0 0 255 (Rouge : 0 , Vert : 0 , Bleu : 255 la valeur maximale)

La couleur du pixel sera le résultat de la synthèse additive de ces 3 couleurs.

Dans le codage RVB 24 bits, chaque couleur primaire sera codée sur 8 bits (3×8 bits , c'est à dire $(2^8)^3 = 2^{24}$ couleurs possibles environ 16,8 millions)

3.2 Création d'images : transformations de drapeaux

3.2.1 Drapeau français

On crée un fichier qui sera lu comme une image au format (ppm).

- Dans un bloc-notes, taper :

```
P3                                #fichier de format ppm
3      3                          #taille image (elle contient 9 pixels)
255                                         # valeur maximale des composantes
0      0      255                      #premier pixel bleu
255    255    255                      #deuxième pixel blanc
255    0      0                        #troisième pixel rouge
```

copier-coller deux fois les trois lignes précédentes

- Enregistrer le fichier sous le nom drapeau.txt
- Supprimer son extension, puis l'ouvrir dans Gimp. Zoomer pour voir l'image. Vérifier que le drapeau français a bien été créé .



3.2.2 Drapeaux en gris

Pour obtenir un drapeau en niveau de gris, à partir du drapeau français précédent, on peut :

- Gris moyen :

Pour chaque pixel (R,V,B), calculer $m = \frac{R+V+B}{3}$ et remplacer les anciennes valeurs de ce pixel par (m, m, m) .

Enregistrer ce nouveau fichier sous le nom drapeaugrismoyenne.txt, puis l'ouvrir dans Gimp.

- Gris clarté :

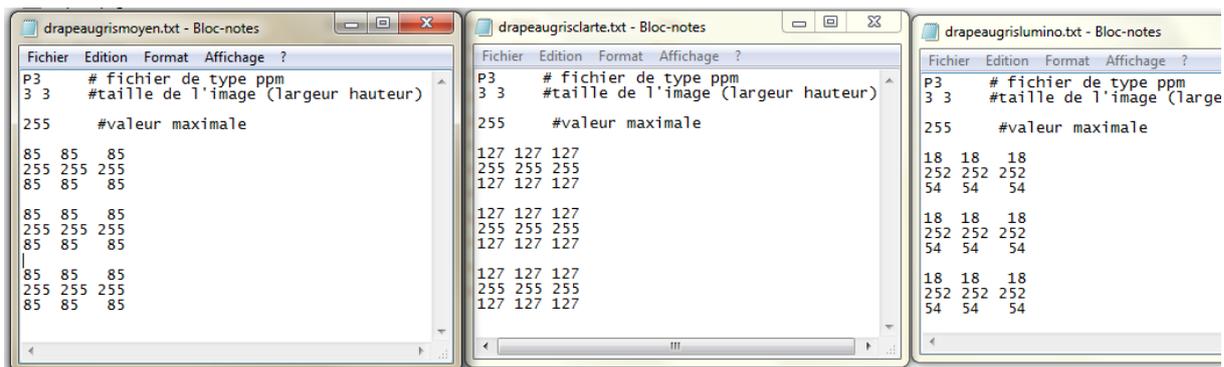
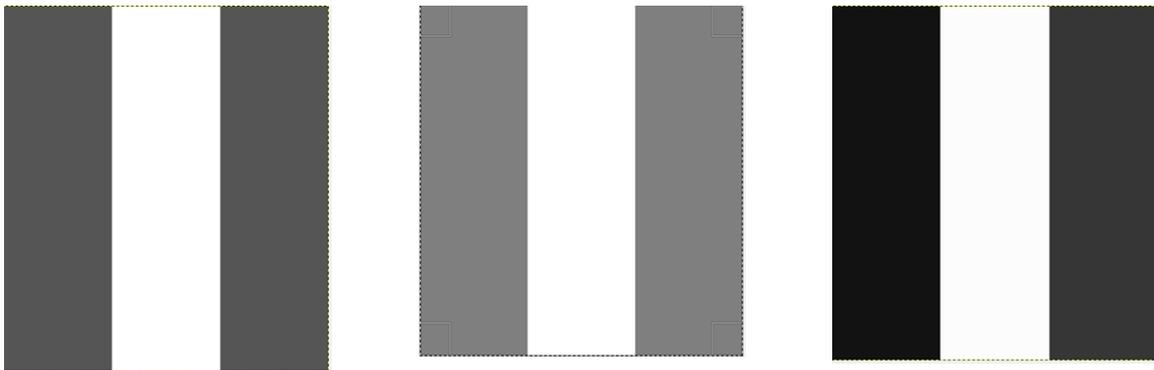
Pour chaque pixel (R,V,B), le niveau de gris est la moyenne entre le minimum et le maximum des trois composantes RVB.

Si par exemple (R,V,B) = (122,200,147), cette moyenne vaut $(122+200)/2 = 161$, et le résultat est : (R,V,B) = (161,161,161).

Enregistrer ce nouveau fichier sous le nom drapeaugrisclarte.txt, puis l'ouvrir dans Gimp.

- Gris luminosité :

Pour chaque pixel (R,V,B), le niveau de gris correspond à $G = 0,21 \cdot R + 0,71 \cdot V + 0,07 \cdot B$, et le pixel est remplacé par (G,G,G) Enregistrer ce nouveau fichier sous le nom drapeaugrisluminosite.txt.



3.2.3 Drapeau noir et blanc

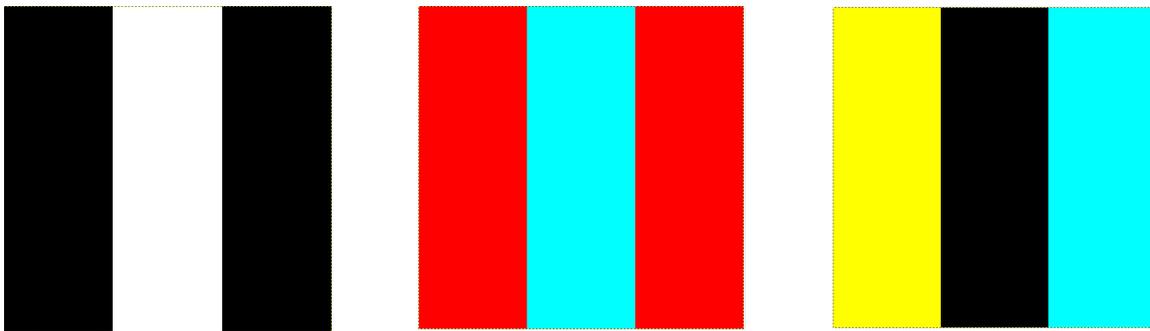
A partir de l'un des 3 fichiers .txt des drapeaux gris précédents, modifier selon la règle suivante :

Pour chaque pixel, les composantes RVB sont identiques, mais ne peuvent prendre que les valeurs 0,0,0 (noir) ou 255,255,255 (blanc). Il faut se fixer un **seuil**, par exemple 128 : en-dessous de 128, on remplace par 0 et au-dessus, par 255.

3.2.4 Encore d'autres drapeaux

On peut obtenir facilement à partir de l'image 'drapeaunoirblanc', l'image en cyan et rouge : il suffit de convertir les pixels noirs en cyan et les blancs en rouge : Cyan = Vert + Bleu, donc RVB = 0,255,255 pour le Cyan.

On peut aussi obtenir le drapeau français en **négatif** : il suffit de reprendre le drapeau français de départ et remplacer chaque pixel par (255-R ,255-V, 255-B)

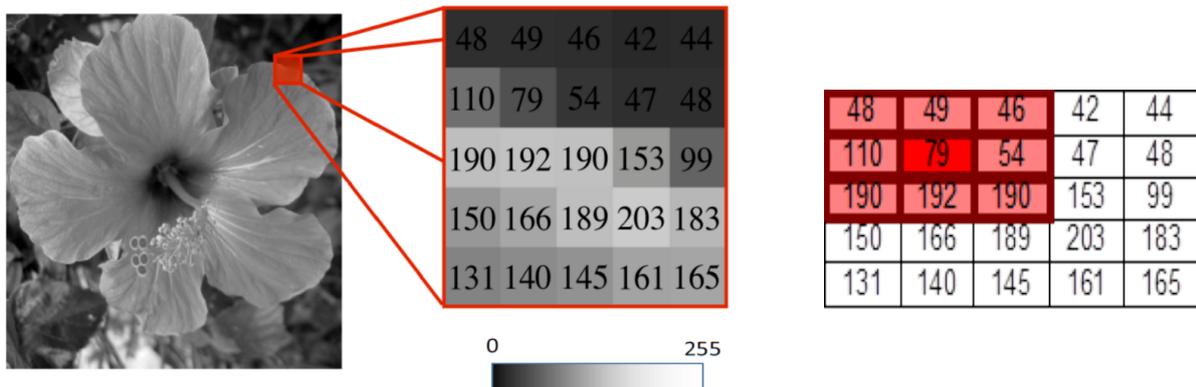


conclusion : Ce travail avec les drapeaux nous a permis d'apprendre à modifier une image par des calculs sur les pixels, nous sommes passés d'une image en couleur à une image en niveau de gris puis à l'aide de la notion de seuil à une image en noir et blanc.

3.3 Calculs de paramètres statistiques sur des images en niveaux de gris

3.3.1 Calculs utilisant les " premiers voisins"

On extrait les pixels d'une partie d'une image, on calcule différents paramètres statistiques [2] et on visualise les résultats à l'aide de Gimp.

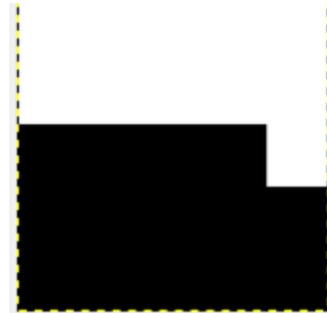


Moyenne : 79 va être remplacé par la moyenne de tous ses pixels voisins :
 $(48+49+46+110+79+54+190+192+190)/9=106$

On fait de même pour les autres pixels.

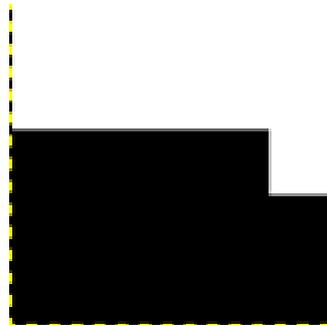
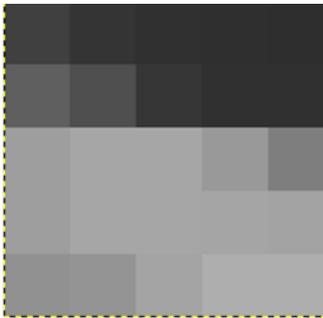
Puis on transforme d'abord le nouveau tableau en image, et on applique un seuil : Si la valeur est < 128 alors on colorie le pixel en blanc et si elle est supérieure à 128, on le colorie en noir.

72	64	53	47	45
111	106	95	80	72
148	147	141	130	122
162	166	171	165	161
147	154	167	174	178



Médiane : On construit un nouveau tableau où chaque pixel est remplacé par la médiane de ses « voisins ». 79 situé au centre de la zone rouge est remplacé par la médiane de (48,49,46,110,79,54,190,192,190) qui est 79. [3]

64	52	48	47	46
95	79	54	48	48
158	166	166	153	126
158	166	166	165	163
145	148	164	174	174



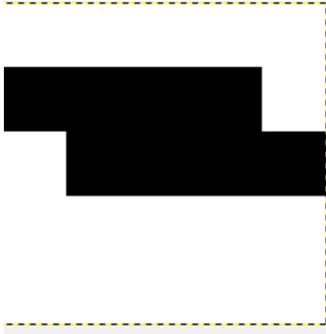
Variance : Elle permet de mesurer la dispersion des valeurs des pixels autour de la moyenne de ces pixels. 79 est remplacé par : $((48 - 106)^2 + (49 - 106)^2 + (46 - 106)^2 + \dots + (190 - 106)^2)/9 = 3908$. Ensuite, pour obtenir une image et des valeurs entre 0 et 255, on multiplie tous les résultats par 255/ 4011.

Dans l'image finale, on devine le bord du pétale en noir.

649	541	150	14	6
3607	3908	3707	2719	1696
1705	2500	3574	4011	3812
548	551	471	876	1029
169	438	606	447	277

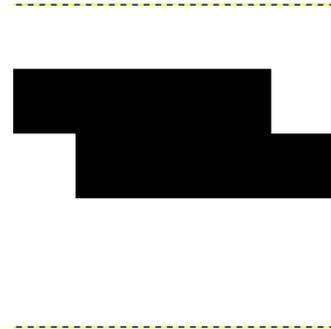
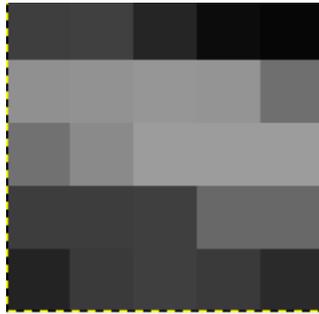
41	34	10	1	0
229	248	236	173	108
108	159	227	255	242
35	35	30	56	65
11	28	39	28	18





Etendue=max-min : 79 est remplacé par la différence entre la valeur maximale et la valeur minimale des 9 valeurs ($192-46 = 146$).

62	64	37	12	6
144	146	150	148	111
113	138	156	156	156
61	61	63	104	104
35	58	63	58	42



3.3.2 Avec une fenêtre mobile ou masque

On peut calculer les mêmes paramètres avec plus de "voisins". Pour un pixel donné non situé sur un bord, lorsqu'on prend seulement les 8 pixels immédiatement voisins, on définit une "fenêtre mobile (ou masque) de taille $n = 1$ ". Autour d'un pixel donné, si on prend les 2 premiers pixels voisins il peut y avoir jusqu'à 25 "voisins", on définit alors un masque de taille $n = 2$. On peut ainsi faire varier la taille du masque et observer les effets.

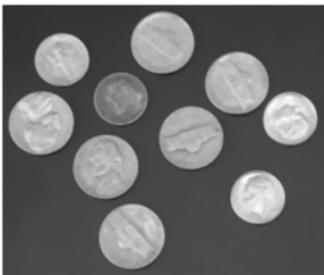
On utilise les logiciels « scilab » et ImageJ et on applique [4] les « filtres » décrits ci-contre médiane, variance etc... On peut appliquer plusieurs filtres successivement. On peut faire varier la taille du « masque » qui entoure le pixel considéré.

Image de Départ

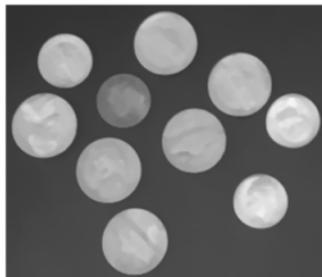


Exemples de filtres selon la taille du « masque »

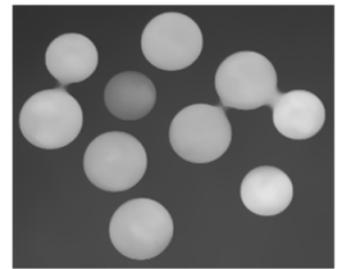
Filtre de la médiane n=1



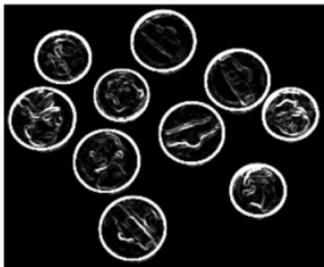
Filtre de la médiane n=2



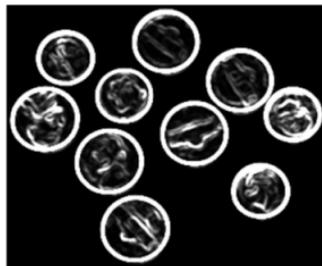
Filtre de la médiane n=10



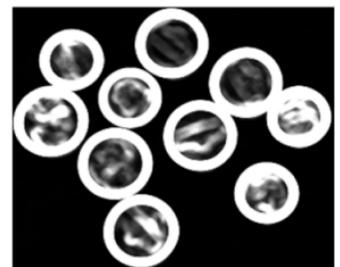
Filtre de la variance n= 1



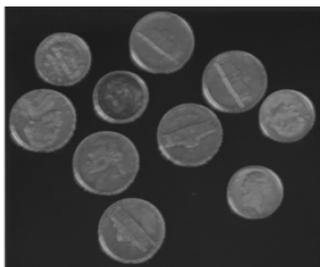
Filtre de la variance n= 2



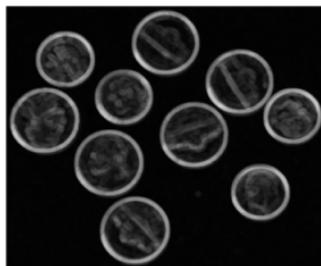
Filtre de la variance n= 10



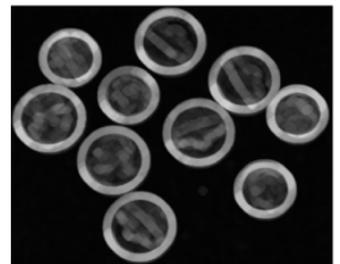
Filtre min-max n= 1



Filtre min-max n= 2



Filtre min-max n= 4

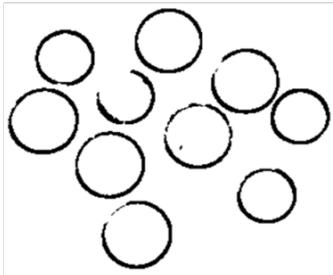


Il semble que la médiane et la moyenne "floutent" les images, enlèvent les détails et uni-

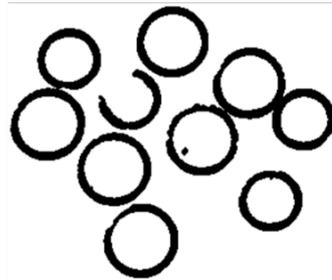
formisent les zones. La variance et l'étendue font au contraire ressortir les contrastes. [5]

Exemples de filtres successifs

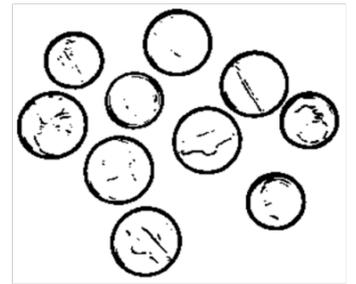
Filtre min-max $n=2$
et seuil =115



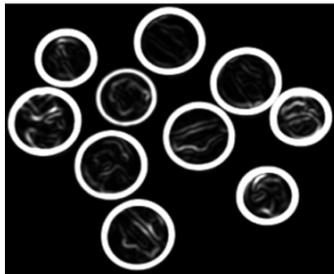
Filtres min-max $n=4$ et seuil
=128



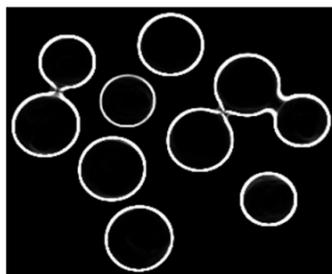
Filtres moyenne $n=1$, min-
max $n=1$ et seuil =128



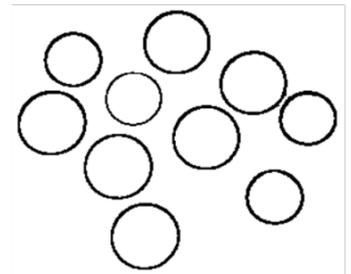
Filtres med $n=2$ puis
variance $n=2$



Filtres med $n=10$, variance
 $n=2$



Filtres med $n=7$, variance
 $n=1$ et seuil 128



Si on veut obtenir le contour des pièces uniquement, il faut combiner le filtre de la médiane avec un masque assez grand, avec celui de la variance avec un petit masque.

3.4 Application à d'autres images

Image de départ



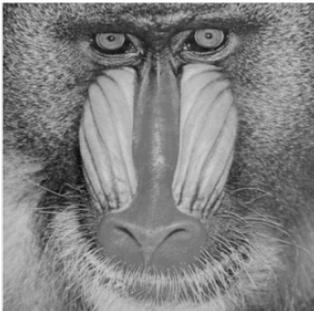
Filtres med n=2 , var n=2



Filtres med n=7, var n=1 , seuil = 65



Image de départ



Filtre s med n=5 , var n=1



Filtres med n=5 , var n=1, seuil = 75

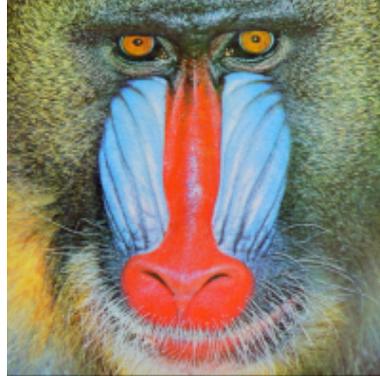


4 Conclusion

Dans le cas des pièces de monnaie :

L'utilisation de paramètres statistiques permet bien de mettre en évidence les contours des pièces. Les meilleurs résultats sont obtenus en combinant le filtre de la médiane avec un masque de taille comprise entre $n = 4$ et 7 et celui de la variance avec un masque de taille 1 ou 2 et en jouant sur le seuil.

Pour d'autres images, les résultats sont moins bons si il y a des motifs comme par exemple, les plumes du chapeau de Léna ou les moustaches ou les poils du babouin. De même , le contour du nez du babouin n'apparaît pas bien. Peut-être que cela pourrait être à cause des couleurs d'origine de l'image et du passage en niveaux de gris .



L'idée pour améliorer cela est de partir de l'image du babouin en couleur et de séparer l'image en 3 images (une qui contient les valeurs des composantes du bleu des pixels, une pour les composantes du vert et une pour les composantes du rouge) et d'appliquer les filtres précédents sur ces trois images, mais nous n'avons pas eu le temps de réaliser cela. A suivre ...

"Au départ, nous n'aurions jamais imaginé qu'une image était un tableau de nombres et qu'on pouvait la modifier en faisant des statistiques!" (Yasmine)

Notes d'édition :

- [1] L'article n'est pas réellement un travail de recherche comme il est attendu pour un article de MATH.en.JEANS. Cependant, la clarté de l'exposition justifie sa publication.
- [2] Quelles raisons ont incité les auteurs à choisir les filtres statistiques considérés dans cette section ?
- [3] Comment est calculée la médiane ?
- [4] Comment les auteurs ont-ils programmé ces transformations en sci-lab ?
- [5] Peut-on essayer d'expliquer pourquoi les filtres statistiques utilisés ont cet effet ?