Td 1 : Amélioration d'images ; approches basées sur l'histogramme

1 Introduction

Les notions d'imagerie abordées en cour sont illustrées dans ce td à l'aide de matlab, outil de calcul numérique. La table 1 montre un exemple de script matlab permettant de lire, manipuler et sauvegarder des images.

Le but de ce TD est de tester différentes approches permettant d'améliorer la qualité des images dans le but de les rendre mieux adaptées à une application.

2 Amélioration d'images par modification d'histogrammes

L'histogramme d'une image `I` est définit par :

$$H(x) = Card\{\mathbf{p} : \mathtt{I}(\mathbf{p}) = x\} \tag{1}$$

avec $x \in [0; 255]$ pour une image codée sur 256 niveaux de gris.

travail à réaliser

1. **Calcul d'histogramme** : Vous devez créer une fonction `h=histo(I,nbclasses)` (dans un fichier `histo.m`) qui retourne l'histogramme de l'image `I` passée en argument, ramené sur `nbclasses` classes. Un exemple de fonction matlab est donné dans la table 2.
2. Dans la pratique, l'histogramme est normalisé :

   $$H(x) = Card(\mathtt{I})^{-1}.Card\{\mathbf{p} : \mathtt{I}(\mathbf{p}) = x\}$$

   Modifier la fonction `h=histo(I,nbclasses)` afin d'obtenir un histogramme normalisé
3. **Etirement d'histogramme** : L'étirement d'histogramme permet de modifier l'image afin d'obtenir la dynamique maximale tolérable ($[0; 255]$). Créer une fonction `I2=histe(I)` permettant de modifier l'image `I` en étirant l'histogramme. Vous devez obtenir le résultat de la figure 1. Pour découper la figure en plusieurs sous-figures, utilisez la commande `subplot(bl,nc,i) ;`. l'aide de cette commande est consultable par : `help subplot`.
4. **Histogrammes couleur** : De nombreux travaux utilisent la notion d'histogramme pour décrire la répartition de caractéristiques dans une image. Ainsi, il est possible de construire des histogrammes de couleurs, des histogrammes d'orientation des gradients, des histogrammes de réponse à des filtres, ...

   Dans le cas d'une imae couleur, chaque pixel est décrit par un vecteur de dimension trois, codant l'information couleur. Il existe un grand nombre de codags possible pour la couleur (RGB, HSV, YUV, YCrCb, LAB, ...). Parmis ces codages certains

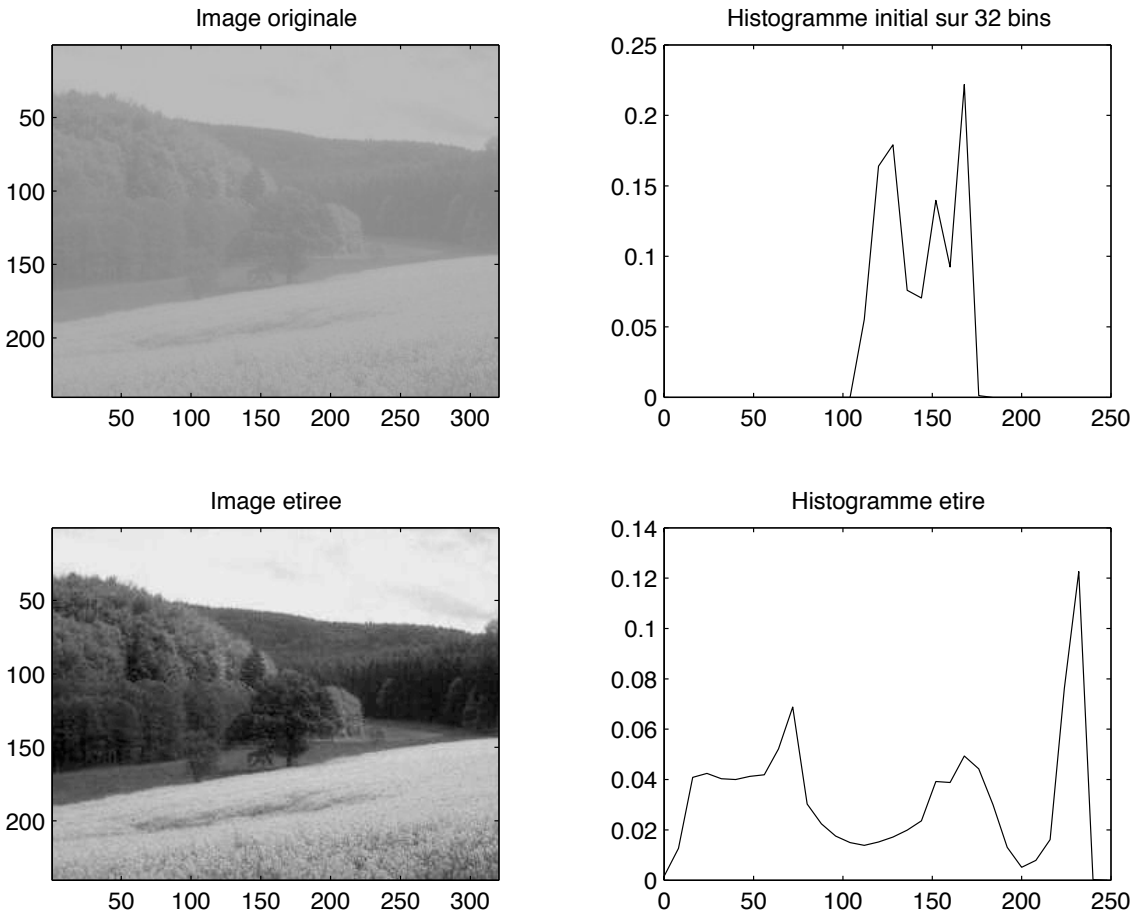

Figure 1 – Etirement d'histogramme

séparent la composante de chrominance de la composante de luminance. C'est le cas, par exemple du codage HSV (H : *hue* ou teinte, S : *Saturation*, et V : *Value* ou luminance).

Vous devez modifier le code actuel pour afficher, sous la forme d'une pseudo-image, les composantes de teinte, saturation et luminance d'une image (la fonction matlab `RGB2HSV`) vous permet de transformer une image RGB et image HSV), ainsi que l'histogramme marginalisé de la teinte. Les tests seront effectués sur l'image `billard large.jpg`.

5. **Modification automatique de teinte** : L'image `billard large`. montre une table de billard. Comme cette dernière occupe la partie la plus importante de l'image, et que sa teinte est homogènene, il en résulte un pic d'histogramme de teinte important autour de cette valeur. Vous devez rechercher ce pic, et effectuer une extraction de zone autour de cette valeur.Le masque ainsi crée pourra alors être utilisé pour modifier la teinte des pixels correspondants, en appliquant un offset. Le résultat obtenu sera alors une modification de la teinte des pixels du tapis du billard (cf figure 2).

```
%%%%%%%%%%%%%%%%%%%%%%%%
% script d'exemple matlab
%%%%%%%%%%%%%%%%%%%%%%%%
%
% lecture d'une image
[I,map]=imread('meadownb.jpg');
%%%%%%%%%%%%%%%%%%%%%%%%
% Affichage d'une image
image(I);
% pause il faut appuyer sur une
% touche pour reprendre
pause ;
colormap(gray(256));
%%%%%%%%%%%%%%%%%%%%%%%%
% Convertion de l'image en
% format double afin de la modifier
Id=double(I);
%%%%%%%%%%%%%%%%%%%%%%%%
% Extraction d'une partie de l'image
% des lignes 100 à 300 et
% des colonnes 200 à 400
I3=Id(100:300,200:400);
%%%%%%%%%%%%%%%%%%%%%%%%
% Sauvegarde d'une image
imwrite(uint8(I3),gray(256),'toto.jpg','JPEG');
```

Table 1 – Exemple de script matlab.

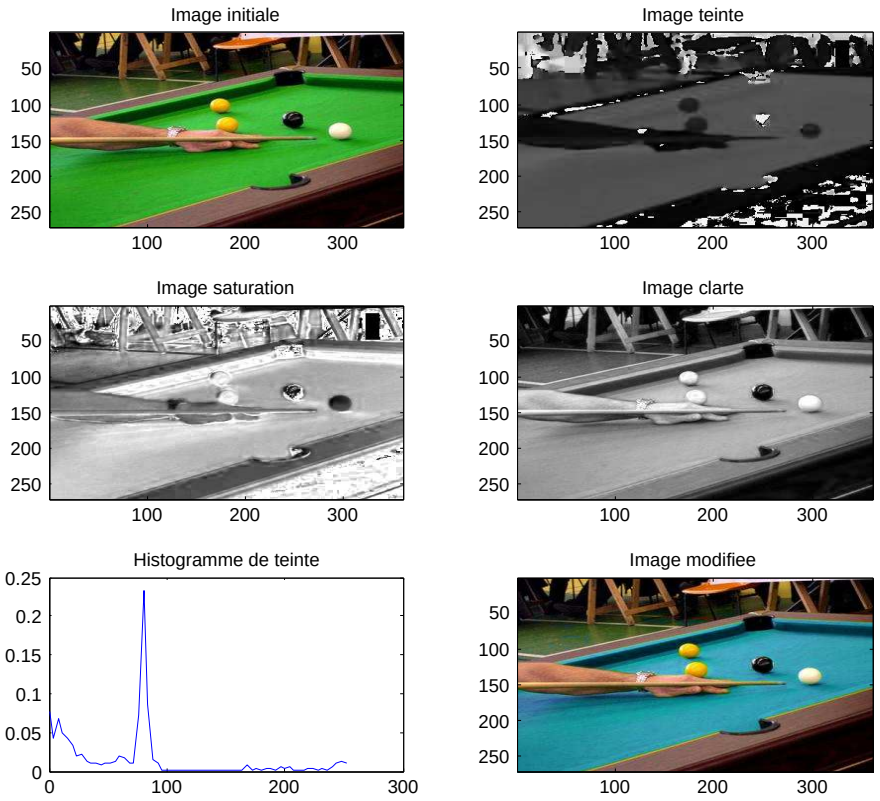

FIGURE 2 – Modification de teinte automatique

```
%%%%%%%%%%%
% Exemple de fonction matlab fichier fo.m
% Les parametres d'entree sont a,b,c
% les parametres de sortie sont t1 et t2
%%%%%%%%%%%
%%%%%%%%%%%
% Prototype
function [t1,t2]=fo(a,b,c)
%
t1=a+b;
t2=b+c;
```

Table 2 – Exemple de fonction matlab.