

TP1 – Espaces de représentation des couleurs

Exercice 1 : corrélations et contrastes des canaux RVB

Le script `exercice_1.m` lit l'image couleur `autumn.tif` codée en RVB (rouge, vert, bleu), la stocke dans une matrice tridimensionnelle `I` et l'affiche : `I(i,j,1)`, `I(i,j,2)` et `I(i,j,3)` contiennent les niveaux de couleur (entiers compris entre 0 et 255) du pixel situé sur la ligne `i` et la colonne `j`, dans les canaux R, V et B. Complétez ce script de façon à découper la matrice `I` en trois matrices bidimensionnelles `R`, `V` et `B` correspondant aux trois canaux. Affichez ces trois matrices sous la forme d'images en niveaux de gris.

Mis à part pour quelques régions colorées (observez la « disparition » de l'arbre situé au centre de l'image, dans le canal B), il existe une forte corrélation entre les trois canaux. Calculez les trois coefficients de corrélation linéaire correspondants, à l'aide des opérateurs `mean`, `.^2` et `.*` (évités les fonctions `var` et `covar`, qui appliquent certains prétraitements aux données).

Le contraste total de l'information de couleur est donné par $c = \sigma_R^2 + \sigma_V^2 + \sigma_B^2$, où σ_R^2 , σ_V^2 et σ_B^2 désignent les variances des niveaux de couleur dans les canaux R, V et B. Calculez les *proportions de contraste* σ_R^2/c , σ_V^2/c et σ_B^2/c contenues dans les différents canaux.

Transmission d'une image couleur par un seul canal

Le choix d'un espace de représentation des couleurs s'est posé lorsque les chaînes de télévision sont passées à la couleur, dans les années soixante. En effet, il était inutile de transmettre trois canaux R, V et B aux utilisateurs (encore nombreux) possédant des téléviseurs noir et blanc, qui ne pouvaient afficher qu'un seul canal. Le principal critère était de maximiser la proportion de contraste contenue dans ce seul canal.

L'idée la plus simple semblerait de transmettre un des trois canaux R, V ou B, choisi arbitrairement. Lancez le script `exercice_1.m` sur l'image `gantrycrane.png`, à dominante bleue, afin de comprendre pourquoi cette idée ne serait pas acceptable.

La conversion d'une image couleur en une image en niveaux de gris consiste en une *réduction de dimension*. Or, l'*analyse en composantes principales* (ACP) est une technique très générale de réduction de dimension.

Exercice 2 : analyse en composantes principales

Complétez le script `exercice_2.m` qui consiste, dans un premier temps, à construire une matrice de données `X` de taille $n \times 3$, où n est le nombre de pixels de l'image et où chaque ligne contient les niveaux de couleur d'un pixel dans les canaux R, V et B. À l'aide de la fonction `plot3`, affichez le nuage de points 3D correspondant. Construisez la matrice `Xc` des *données centrées*, dans un repère ayant pour origine le centre de gravité `G`.

Calculez la matrice de variance/covariance `Sigma` des données centrées, de taille 3×3 . Cette matrice symétrique réelle admet une base orthonormée de vecteurs propres. Calculez ses valeurs et vecteurs propres grâce à un appel de la forme `[W,D] = eig(...)`. Pour une matrice de valeurs propres telle que `Sigma`, les valeurs propres sont triées par ordre croissant, bien que cela ne soit pas indiqué explicitement dans la documentation en ligne (cf. `help eig`). Quant aux colonnes de la matrice `W`, elles constituent les vecteurs propres de `Sigma`, donc `W` est la matrice de passage vers le nouveau repère (cette matrice est orthogonale). Il est donc nécessaire d'inverser l'ordre des colonnes de `W` pour que les valeurs propres associées soient triées par ordre décroissant.

Calculez, et stockez dans une matrice de nom `Z`, les composantes des pixels dans le nouveau repère, appelées *composantes principales*. Affichez les trois images en niveaux de gris correspondant à ces composantes. Vous constatez que les valeurs affichées peuvent être négatives, contrairement aux niveaux de couleur de l'image d'origine. Calculez les coefficients de corrélation linéaire entre les trois paires de composantes principales.

Selon vous, d'où vient que le contraste total soit conservé dans ce nouveau repère (orthonormé), c'est-à-dire que $\sigma_1^2 + \sigma_2^2 + \sigma_3^2 = c$? Comparez la proportion de contraste σ_1^2/c contenue dans la première composante principale à la proportion de contraste σ_R^2/c contenue dans le canal rouge. Concluez.

Exercice 3 : combinaison linéaire des trois canaux RVB

Si le critère retenu est celui du contraste, l'ACP est incontestablement le meilleur moyen pour convertir une image couleur en une image en niveaux de gris. Cependant, la matrice de passage dépend de l'image considérée. Or, il n'était pas envisageable, dans les années soixante, d'effectuer une ACP pour chaque image d'une séquence télédiffusée. C'est pourquoi une matrice de passage commune à toutes les images couleur a été choisie.

Que ce soit pour `autumn.tif` ou pour `pears.png`, la première colonne de la matrice W (après inversion de l'ordre de ses colonnes) contient trois valeurs très proches de $1/\sqrt{3}$. Cela semble indiquer que la maximisation du contraste s'obtient en donnant le même poids aux trois canaux. Ce résultat n'est pas vrai lorsque la proportion de contraste diffère beaucoup d'un canal à l'autre (faites le test sur l'image `gantrycrane.png`), mais il est vrai *en moyenne*. Donner le même poids aux trois canaux revient à transformer une image couleur en une image en niveaux de gris de la manière la plus intuitive qui soit, à savoir :

$$I_{\text{avg}} = \frac{1}{3}(R + V + B) \quad (1)$$

Pourtant, la fonction `rgb2gray` de Matlab, qui est spécifiquement conçue pour transformer une image RVB en une image en niveaux de gris, effectue une autre combinaison linéaire (cf. `help rgb2gray`) :

$$Y = 0,2989 R + 0,5870 V + 0,1140 B \quad (2)$$

qui disymétrise les canaux R , V et B , contrairement à (1). Cela vient de ce que le système visuel humain n'est pas également sensible aux différentes fréquences lumineuses. En particulier, comme sa courbe de sensibilité est maximale dans le vert, le poids du canal vert dans (2) est supérieur aux deux autres.

Relancez le script `exercice_2.m` sur l'image `autumn.tif`. Affichez ensuite l'image en niveaux de gris obtenue en appliquant la fonction `rgb2gray` à cette même image couleur : parmi ces images en niveaux de gris, laquelle vous semble-t-elle posséder le meilleur contraste ?

Espace de représentation des couleurs YCbCr

L'expression (2) n'est autre que la première des trois formules de passage de l'espace RVB à un autre espace de représentation des couleurs appelé YCbCr : Y est la *luminance*, C_b la *chrominance bleue* et C_r la *chrominance rouge*. Les deux autres formules de passage s'écrivent comme suit (`rgb2ycbcr` permet d'effectuer ce changement d'espace en Matlab) :

$$\begin{aligned} C_b &= -0,1687 R - 0,3313 V + 0,5 B + 128 \\ C_r &= 0,5 R - 0,4187 V - 0,0813 B + 128 \end{aligned} \quad (3)$$

L'ajout de 128 à C_b et C_r permet d'obtenir des valeurs comprises entre 0 et 255.

Il est notable que l'espace de représentation des couleurs YCbCr n'est pas orthonormé : en utilisant les formules (2) et (3), on montre facilement que la matrice de passage n'est pas orthogonale. Mais c'est surtout parce qu'il est utilisé par le format JPEG que l'espace YCbCr est connu. En effet, comme le système visuel humain est plus sensible à la luminance qu'aux chrominances, on peut réduire la taille d'une image en dégradant ses chrominances sans trop altérer sa qualité.