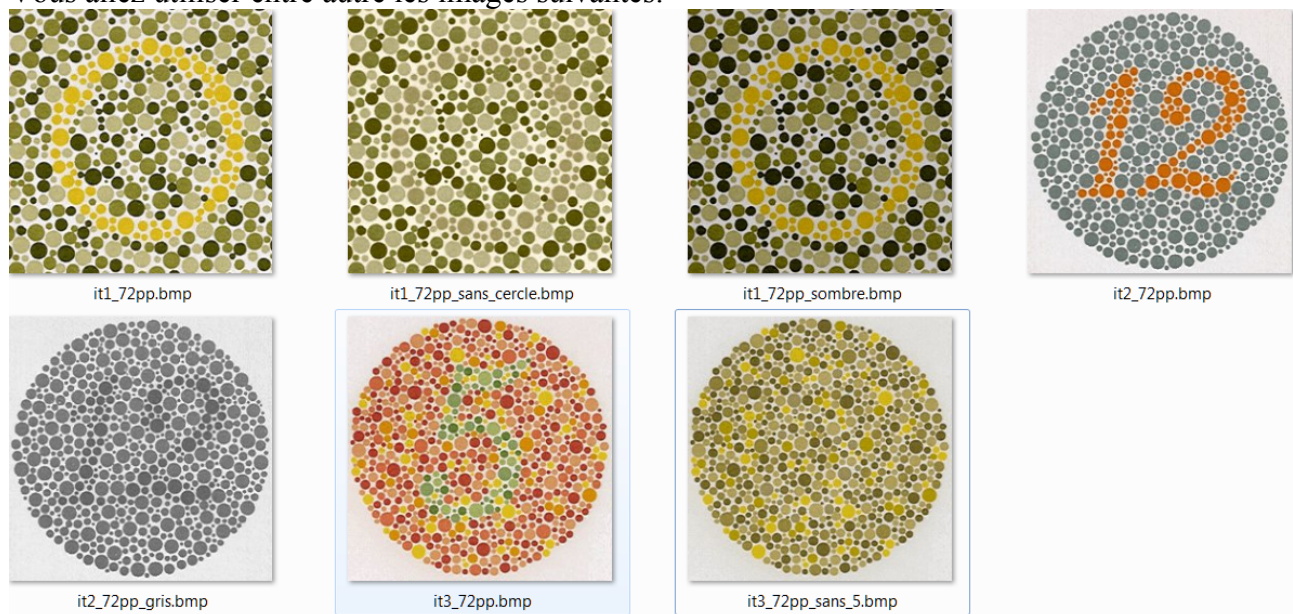


## M2 IVI – Module VISA - TP 1 Segmentation d'images couleur – version 2019

L'objectif du TP est de convertir l'image couleur dans l'espace HSV ou CIE L\*a\*b\* et de retrouver les chiffres par simple seuillage de la composante que vous aurez sélectionnée par des macros ImageJ que vous allez concevoir.

La couleur d'un pixel est stockée dans un mot 32bits, 8 bits étant alloués pour chaque composante. Les 8 bits de poids faible sont alloués à la composante bleue (B), les 8 bits suivants sont alloués à la composante verte et les 8 bits encore suivants sont alloués à la composante rouge. Les 8 bits de poids fort ne sont pas utilisés.

Vous allez utiliser entre autre les images suivantes:



Les espaces couleur considérés par l'outil 'Color Inspector3D' et 'Color Space Converter' d'ImageJ sont RGB, HSL, Lab.

### A RENDRE:

Les macros commentées des exercices.

### Exercice 1 Images Cas1:

Q1. Trouver avec le plugins 'Plugins-Color- Color Inspector 3D' la composante couleur qui permettra de retrouver le nombre 26 dans l'image 'cas\_1\_dalton26.bmp' par un simple seuillage . Développer alors la macro qui code l'image dans la composante couleur sélectionnée (plugin Color Space Converter). Puis il faudra chercher les seuils min et max les mieux adaptés grâce au menu (Image-Adjust-Threshold). La macro devra alors appliquer les seuils par les fonctions setThreshold et run("Convert to Mask").

La macro doit permettre de segmenter automatiquement l'image 'cas\_1\_dalton42.bmp'.

### Exercice 2 Images Cas2:

Suivre la même approche pour segmenter les images 'cas\_2\_dalton7.bmp', 'cas\_2\_dalton16.bmp' et 'cas\_2\_dalton73.bmp'.

### Exercice 3 Images Cas4:

Suivre la même approche pour segmenter les images 'cas\_4\_dalton8.bmp'. Cependant, il faudra dans ce cas convertir l'image codée dans l'espace RGB vers l'espace Y'UV (télévision) selon les

équations suivantes :

Equivalently, substituting values for the constants and expressing them as [matrices](#) gives these formulas for BT.601:

$$\begin{bmatrix} Y' \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.14713 & -0.28886 & 0.436 \\ 0.615 & -0.51499 & -0.10001 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix},$$

Pour ce faire, il faudra utiliser la macro exemple image J couleur disponible sur Moodle. Après conversion dans Y'UV, vous identifierez la composante la plus discriminante (après avoir fait la commande Image-Color-Split Channel). Dans ce cas, on tolèrera que les chiffres soient segmentés avec le fond de l'image.

Faire la macro qui permet de segmenter automatiquement par binarisation simple des images du cas 4.