

Programmation pour Ingénieur

*Color Lookup table
&
Image indexée*

*ME 3e semestre
Rev. 2025.1*

Christophe Salzmann

Laboratoire
d'Automatique



Color Lookup table

Une image non compressée est composée d'une matrice de pixels. Chaque pixel a 3 composants de couleurs rouge, vert et bleu (RGB). Chaque couleur est représentée par un nombre encodé sur 8 bit (valeur courante en 2017), des formats professionnels utilisent plus de bits par couleur ou d'autres représentation des couleurs. Chaque pixel utilise donc 24bit, 8 bit de "transparence" ou *alpha channel* peuvent être associés au 24bits de base pour avoir au final 32bit par pixel.

Ex. une image de 640 par 480 pixels, utilise $640 \times 480 \times 3$ bytes
= 921600 Bytes ~0.9 MB

Une option pour réduire l'espace mémoire d'une image encodée en 24bits est de choisir par exemple 2^8 (256) couleurs parmi les 2^{24} (16777216) couleurs possibles.

Les 256 couleurs sélectionnées forment un index appelé *color lookup table* (CLUT).

Cette sélection peut se faire en choisissant les 256 couleurs les plus courantes dans l'image source ou en utilisant un index standardisé, ex. *web standardized clut*.

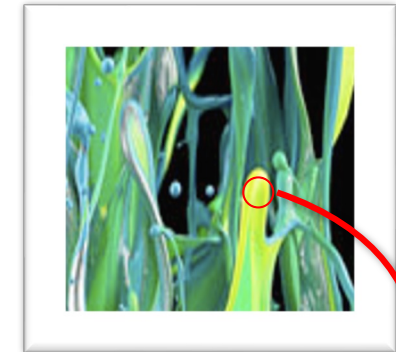
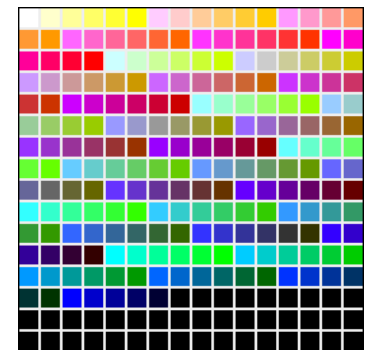


Image source
24bits/pixel

R=85, G=255, B=66,
0055FF42_{Hex}



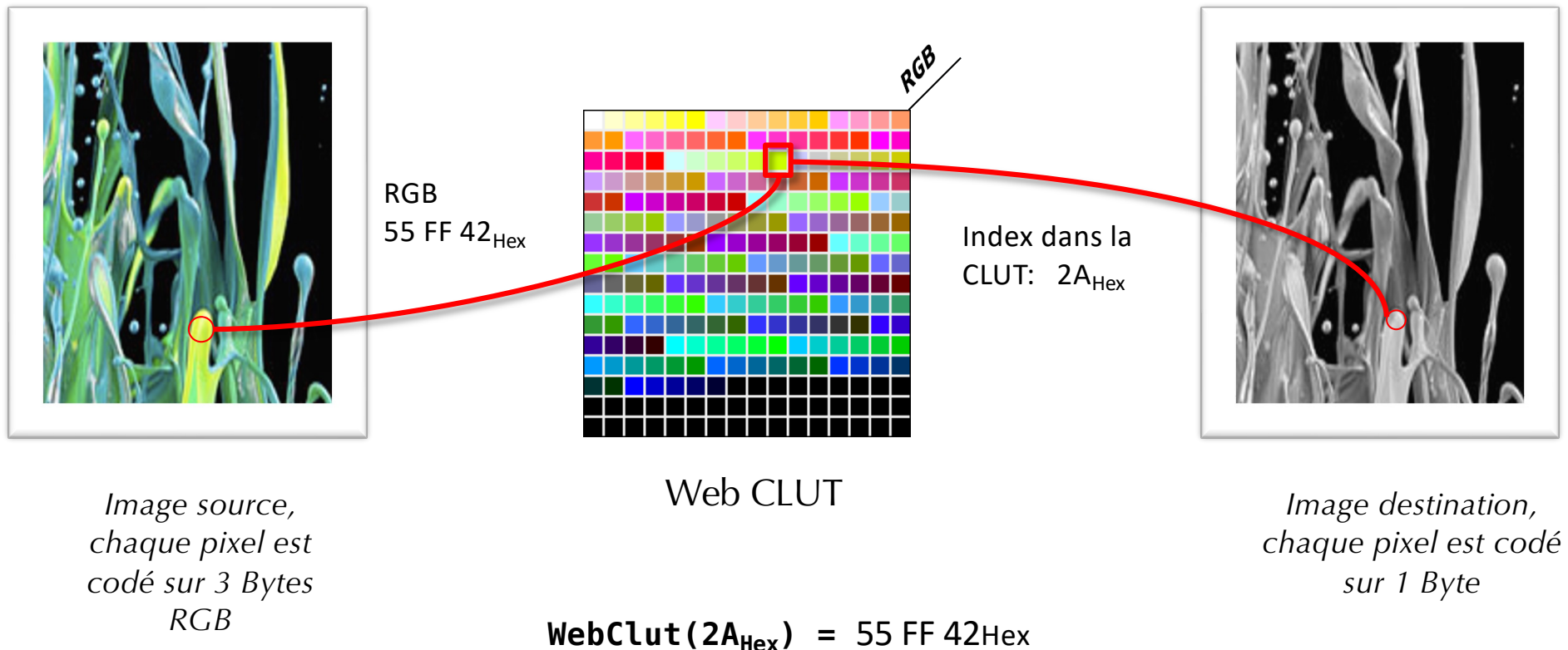
web standardized clut

Color Lookup table

La seconde étape consiste à chercher dans la CLUT/index la couleur la plus proche de la couleur source (24bit) et de sauver uniquement l'index de la CLUT dans l'image destination, ceci pour chaque pixel.

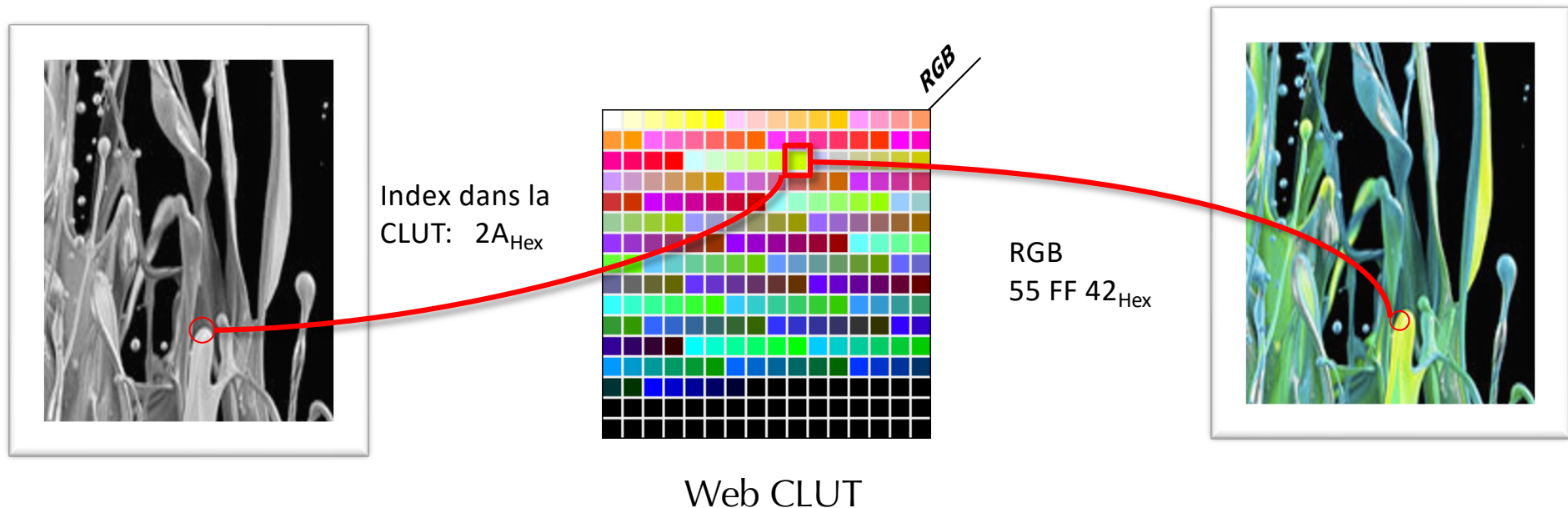
Si l'on néglige la taille du CLUT, la taille mémoire de l'image indexée est ainsi divisée par 3 par rapport à l'image source 24bits.

Ex. image de 640 x 480 px, utilise $640 \times 480 \times 1 + 256 \times 3$ bytes = 307968 Bytes ~0.3MB



Color Lookup table

Pour afficher l'image il suffit de faire l'opération inverse en cherchant dans la CLUT les couleurs qui correspondent aux indexes.

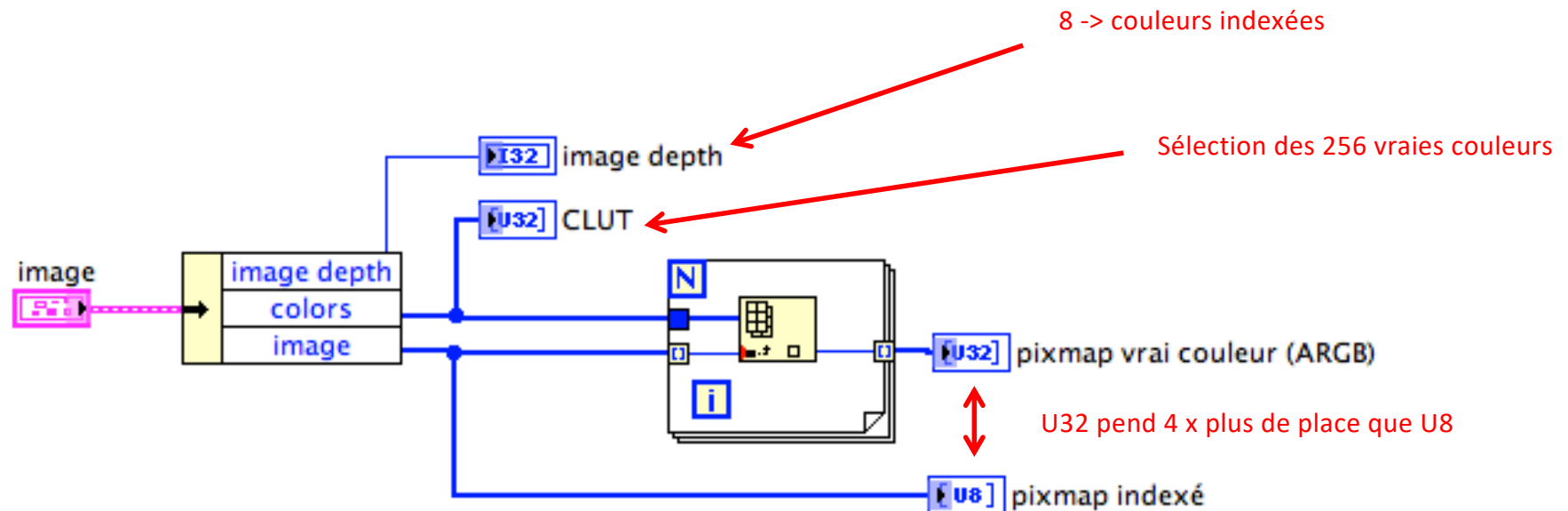


Il y aura des pertes si le nombre de couleurs distinctes dans l'image source est plus grande que le nombre possible de couleurs dans la CLUT ou si les couleurs imposées par la CLUT ne correspondent pas aux couleurs de l'image source.

Color Lookup table

La couleur d'un pixel d'une image LabVIEW peut être codée à l'aide d'une CLUT ou représenté de manière directe. La profondeur de l'image (**image depth**) informe sur le type d'encodage choisit. Si **image depth** vaut **8** cela indique que les couleurs sont indexées. Une valeur de **24** ou **32** indique que la couleur du pixel est représentée de manière directe. Dans ce dernier cas **colors (CLUT)** est vide.

Dans le cas des couleurs indexées, **colors/CLUT** contient un vecteur de 256 valeurs correspondant aux 256 vraies couleurs choisies parmi les 2^{24} couleurs possible.



Color Lookup table

Le VI **color to RGB** extrait chaque composant (RGB) de la couleur fournie. Ces valeurs sont encodées sur 8bits non signés [U8].

Notez que LabVIEW est capable d'afficher uniquement des couleurs 24bits dans les *color box*.

Si vous essayez de connecter un des composants RGB [U8] dans la *color box* [U32], la couleur résultante ne sera pas celle espérée (entre noir et bleu foncé)

