

Compte Rendu 6 – Projet Image & Compression

Arthur Chateauneuf, Luna Bossu



Fonctionnement du logiciel avec une interface graphique

Nous avons décidé de faire une application graphique (GUI) qui sert de "wrapper" à l'application console (CLI), c'est à dire que cette dernière invoque l'application console avec des arguments spécifiques (spécifié avec des input et boutons grâce à l'interface). Pour ce faire et garder l'application multi-plateforme, nous avons décidé d'utiliser de la mémoire partagée pour permettre la communication entre l'application GUI et l'application CLI. Les informations communiquées sont le statut de l'application CLI (ce qu'elle est en train de faire) et le progrès (nombre de sous image traitées) et la valeur maximale du progrès (la valeur quand toute l'image sera traitée).

De plus, après recherches et comparaisons entre différents framework graphiques, nous avons décidé d'utiliser le framework [FLTK](#). Ce framework est un framework léger, cross-platform, qui est simple à utiliser et permet de rapidement créer des interfaces graphiques.

Éléments de l'application GUI

L'application est composée de plusieurs éléments, tout d'abord un bouton pour sélectionner un fichier d'entrée (le dialogue de sélection de fichier est géré par FLTK), et des boutons pour sélectionner l'espace couleur et la technique utilisée. Ainsi qu'une zone pour sélectionner une résolution et un bouton pour tout valider et lancer la génération.

Ensuite, on réserve une large zone à l'affichage de l'image, cette zone est originalement vide puis, lorsqu'on sélectionne une image ou qu'on fini d'en générer une, on l'affiche.

Enfin, on a une barre de progression qui se remplit en fonction du progrès de l'application CLI, cela indique à l'utilisateur que quelque-chose se passe bien, même si cela n'est pas forcément évident. De plus, nous avons mis en place une contingence dans le cas où le programme CLI ne répond plus (par exemple dans le cas d'une erreur de segmentation) qui, si l'application ne met plus à jour la valeur dans la mémoire partagée de manière depuis quelques secondes, nous indiquons à l'utilisateur que l'application a "timeout".

Exemple d'apparence de l'application :

