

Université Gustave Eiffel
Licence 3 Informatique
Programmation avancée en C
Groupe 1 et Groupe 3
Année universitaire 2020-2021

Compression d'images avec des quadrees

Etudiants : MOHAMMAD Haseeb, JAAFRI Amyr

Table des matières

Table des matières	2
Préambule	3
Introduction	4
Développement du projet	5
Lancement de l'application	9
Documentation.....	10
Conclusion	12

Préambule

Le lecteur trouvera dans cette documentation une description générale du projet *Compression d'images avec quadrees*. Il s'agira ici de fournir un **manuel d'utilisateur** mais aussi un **manuel de développement** afin de cerner l'implémentation qui a été la nôtre. La description exhaustive de chaque fonction implémentée ne sera donc pas détaillée ici. Ainsi, nous conseillons vivement au lecteur de se reporter aux commentaires des fonctions présents dans les fichiers d'en-têtes et dans les fichiers sources. En outre, une brève documentation des modules sera disponible.

Introduction

Le but du projet *Compression d'images avec des quadrees* est de fournir une application muni d'une interface graphique permettant la compression et la sauvegarde d'une image. Le projet est à réaliser en binôme. Le projet à été réalisé par Mohammad Haseeb et Jaafri Amyr. Ce projet présente plusieurs problématique, à savoir :

- Modularisation ;
- Parcours de l'arbre (*cf. récursivité*) ;
- Minimisation de l'arbre (*cf. suppression de nœuds*)

Développement du projet

Définition

L'objectif de ce projet est de développer une application pouvant encoder graphique pouvant encoder, décoder et compresser avec ou sans perte des images en utilisant des *quadtrees*.

Contrainte de réalisation

- Codage d'images par arbre quartique
- Approcher une image avec un *quadtree* grandissant
- Minimisation des *quadtrees* (sans et avec perte)
- Format d'encodage
- Description du programme
- Modularité
- Pour aller plus loin (bonus)
- Conditions de développement
- Remarques importantes
- Conditions de rendu

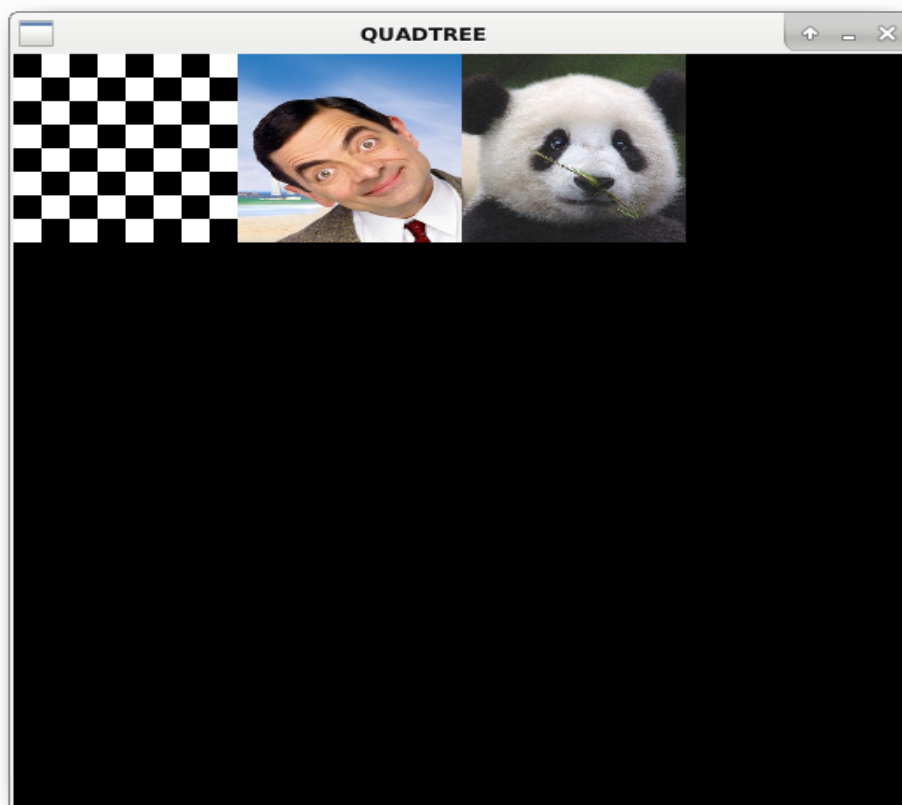
Outils de développement

Pour développer cette application nous avons utilisé le langage de programmation *C* qui est généraliste. Inventé au début des années 1970, *C* est devenu un des langages les plus utilisés.

Interface graphique

L'interface graphique doit être intuitive afin de rendre son utilisation agréable. De plus, cela pourra épargner à l'utilisateur la lecture d'une documentation redondante.

Au lancement du programme, l'interface suivante apparaît :



Il s'agit ici pour l'utilisateur de cliquer sur une des images à compresser afin de procéder à la création de l'arbre et y effectuer les tests .

Ensuite, l'interface suivant apparaît :



L'utilisateur aura donc en fond l'image sélectionné et aura le choix entre plusieurs options représenter par **6 icônes**, tel que (resp. de droite à gauche) :

- (En haut, à gauche) Save as bin in color : sauvegarde sous forme de *quadtree* en binaire et en couleur dans un fichier **.qtc** dans le dossier *img/* ;
- (En haut, à gauche) Save as bin in b & w : sauvegarde sous forme de *quadtree* en binaire (noir et blanc) dans un fichier **.qtn** dans le dossier *img/* ;
- (En haut, à droite) Save as graph in color : sauvegarde du *quadtree* sous forme de graphe en couleur dans un fichier **.gmc** dans le dossier *img/* ;

- (En haut, à droite) Save as graph in b & w : sauvegarde du quadtree sous forme de graphe en noir et blanc dans un fichier **.gmn** dans le dossier **img/** ;
- (En bas, à gauche) Minimisation : Minimise l'arbre avec perte.
- (En bas, à droite) Open img : Ouvre une image ou un fichier encoder. Dans ce cas la fenêtre suivante apparaît :

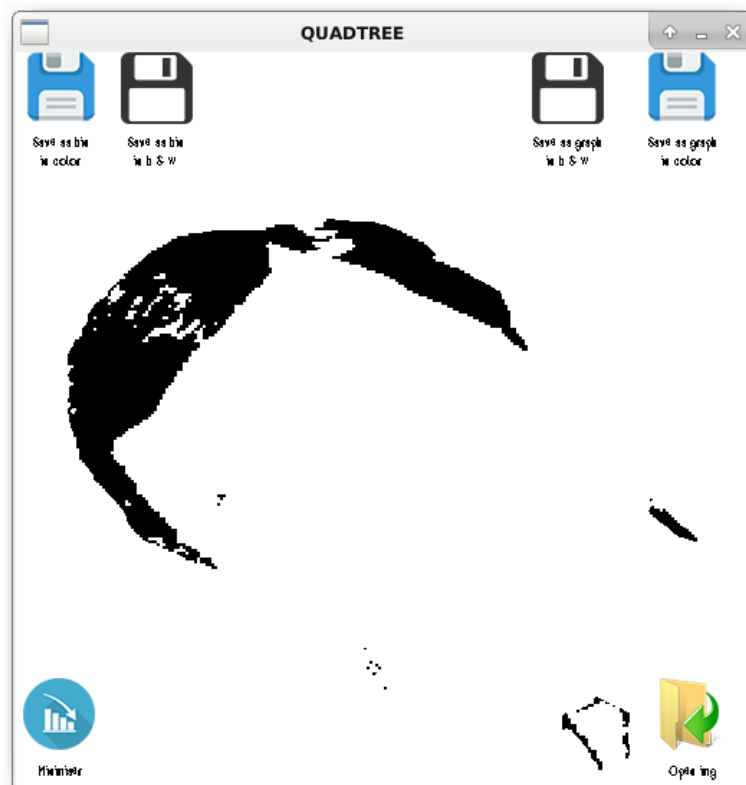


Il s'agira pour l'utilisateur d'indiquer le chemin dans l'arborescence du fichier. Ce dernier peut aussi au préalable déposer une image dans le répertoire **img/** et ainsi la voir apparaître à la première interface (cf. section suivante).

Il est possible d'ouvrir des fichiers d'autres extensions que **.jpg** . Ainsi si on encode l'image précédente dans un quadtree en couleur (i.e. **.qtc**), on obtiendra :



Si on encode l'image *mr_bean.jpg* dans un *quadtree* en noir et blanc (i.e. *.qtn*), on obtiendra :



Lancement de l'application

Commandes

- Pour installer le projet, extraire l'archive, entrer dans le dossier extrait et lancer la commande : **make**.
- Les commandes **make clean**, **make fclean**, et **make re** sont disponible (resp. supprimer les fichiers objets ; supprimer les fichiers objets et l'exécutable ; recompiler le projet).

Options

Il est possible d'ajouter les images que vous voulez visualiser dans le dossier *img/* puis lancer la commande *./quadtrees*.

Par la suite pour ouvrir une autre image ou fichier, cliquer sur le bouton **Open img** et ensuite taper *img/nom fichier* suivi de l'extension (exemple : *img/mr_bean.jpg*).

Documentation

Avant-propos

Cette documentation n'a pas pour visé de remplacer la documentation précisé dans le code. Ainsi, il serait de bonne voie pour le lecteur de se reporter aux commentaires précisés dans les fichiers sources pour de plus amples précisions. Cette section constituera un survol de l'architecture du projet.

Modules

➤ **ColorList**

Auteur : MOHAMMAD Haseeb

Description : Ce module contiendra les fonctions liées à la liste de couleur, à sa mise-à-jour, ainsi qu'à la minimisation.

➤ **Quadtree**

Auteur : MOHAMMAD Haseeb, JAAFRI Amyr

Description : Ce module contiendra les fonctions liées à la distance, à la conversion des couleurs, à la construction de l'arbre de base, l'encodage et le décodage (quadtree/graphe), ainsi que quelques fonctions liées à l'interface graphique.

➤ **Get_next_line**

Auteur : JAAFRI Amyr

Description : Ce module nous permettra de représenter un buffer et de lire ce buffer.

➤ **Libinclude**

Description : Ce module contient toutes les bibliothèques nécessaires à la réalisation du projet.

➤ **Err**

Auteur : MOHAMMAD Haseeb

Description : Ce module nous permettra de gérer les erreurs liées à la lecture/écriture sur un fichier.

➤ **Test**

Auteur : MOHAMMAD Haseeb, JAAFRI Amyr

Description : Ce module nous permettra de tester les fonctions du projet dans un soucis d'éventuel débogage.

En outre, le lecteur soucieux de l'évolution du projet trouvera dans le fichier `log_dev.txt` l'historique des dépôts des fichiers liés au projet.

Conclusion

Ce projet a représenté plusieurs intérêts dont :

- Manipulation des arbres (*i.e.* parcours récursif, ajout récursif, etc ...);
- Manipulation de fichiers.

Les principales difficultés auxquelles nous avons fais face sont :

- La définition d'une bonne structure ;
- Minimisation de l'arbre.