

Rapport du projet IDM VideoGen2

Lien du Git : <http://github.com/cdelourme/VideoGen2>

Le répertoire unique du projet est DelourmeDemeautis.

Les vidéos utilisées sont en DelourmeDemeautis/video.

Les vidéos générés pour la partie web sont en DelourmeDemeautis/src/main/webapp/public/video.

Les images générés pour la partie web sont en DelourmeDemeautis/src/main/webapp/public/thumbnail.

Les videoGen sont en DelourmeDemeautis /samples.

La playlist, le csv sont en DelourmeDemeautis/.

Etude empirique

Pour concaténer les séquences vidéos, nous avons utilisé l'outil FFmpeg. Nos vidéos sont des séquences de films.

Afin de multiplier les variantes, nous avons privilégié des vidéos de basse qualité provenant de Youtube.

En src/main/java/main/model, se trouve une classe TP3 qui a une fonction de Test, generationCSVFile(), générant un fichier csv-file.csv. Ce fichier comporte toutes les variantes possibles pour le fichier samples/example1 ainsi que les tailles des fichiers GIF.

Les fichiers GIF générés en mode normal étant trop volumineux, nous les avons limités à 15 fps et 640x480. Nous avons également testé avec un filtre avancé pour avoir un meilleur rendu, mais la génération devenait trop coûteuse en temps. Il est possible de reproduire les tests en modifiant les paramètres dans la fonction de VideoGenTools.convertMediaToGIF(fichierSource, fichierGifGénéré, 15, 640x480).

Nous obtenons les informations suivantes :

	v08	v09	v0	v04	v05	tailles cumulées	mp4 généré	Gif généré	Ecart % cumul/mp4	Ecart % cumul/Gif
1	FALSE	TRUE	TRUE	FALSE	TRUE	10206206	10204568	56406639	99,98	552,67
2	FALSE	TRUE	TRUE	TRUE	FALSE	9326968	9325370	48838554	99,98	523,63
3	TRUE	FALSE	TRUE	FALSE	TRUE	8238998	8237220	84367215	99,98	1024,00
4	TRUE	FALSE	TRUE	TRUE	FALSE	7359760	7358022	77363396	99,98	1051,17

Il paraît évident que le fait de concaténer les vidéos ne fait pas réduire significativement la taille du fichier généré par rapport à la somme des tailles des séquences. En revanche, la conversion en GIF génère un fichier volumineux. Avec les paramètres utilisés, on multiplie jusqu'à 10 fois la taille.

Le site Web

La partie serveur est en Spring Boot et la cliente en AngularJS. Le tout accessible depuis le port 9200.

Les requêtes utilisées sont :

GET /generate : renvoie au client le nom de la variante vidéo générée.

GET /custom : renvoie au client un videogen « réduit » avec les images représentant les vidéos. Dans le but de composer une vidéo personnalisée.

POST /generatedVOD : récupère depuis le client, une liste de vidéos à concaténer

GET /generatedVOD : renvoie au client le nom de la vidéo personnalisée demandée.

Il reste à implémenter la sauvegarde du fichier GIF. Il faudrait demander à l'utilisateur les paramètres fps et scale puis procéder comme pour generateVOD.

Exercices du TP

Les différents exercices sont réalisés dans les classes TP2, TP3 et TP4 avec des méthodes de test en src/main/java/main/model.

Difficultés rencontrées

L'utilisation de ffmpeg est très compliquée. Elle nécessite de bien connaître le monde des codecs et de la vidéo en générale. Le nombre impressionnant d'options de cet outil permet des réglages fins mais nous effraie également. Finalement, nous aurons passé autant d'heures à utiliser ffmpeg qu'à coder ce projet.