



Projet TNI

Compression vidéo

Dreyer Mathieu

Année scolaire 2019 - 2020

Table des matières

Introduction.....	4
I) Le codec	5
I.1) Comment fonctionne un codec ?	5
II) Choix et étude des codecs.....	6
II.1) Snow	6
II.1.1) Fonctionnement	6
II.2) H264	7
II.3) HEVC	8
II.3.1) Fonctionnement	8
III) Comparaison.....	9
III.1) Etude des codecs	10
IV) Conclusion	14

Introduction

Dans ce projet, une vidéo intitulée Basket est fournie. Le but de ce projet est d'expliquer l'effet de différents codecs sur une vidéo.

Pour se faire, j'ai donc choisi quelques codecs qui seront décrits dans la suite de ce rapport, pour les comparer à taux de compression similaire. Ceci permettra de mieux voir l'effet des codecs sur la vidéo.

Dans un premier temps, nous verrons ce qu'est un codec, comment fonctionnent les codecs en général.

Ensuite, nous étudierons les codecs choisis, en décrivant leur fonctionnement, et autre fonction.

Enfin, une comparaison de tous les codecs choisis sera effectuée, avec des taux de compression similaire pour éviter de fausser les résultats.

I) Le codec

Un codec est un logiciel qui décompresse ou compresse les signaux numériques. Ce mot vient de la fusion entre **c**odeur et **dé**codeur. Il est présenté comme un instrument capable de compresser ou décompresser un flux, vidéo ou audio tout en respectant un certain standard (vidéo ou audio).

I.1) Comment fonctionne un codec ?

Un codec suit des règles bien précises, généralement appelé des normes. Les codecs les plus connus sont H.264, DivX, MP3, et MPEG4.

Ils permettent de stocker facilement sur ordinateur une vidéo, mais aussi une transmission rapide à un lecteur (une PSP par exemple) ou en ligne.

Sans le codec adapté pour une vidéo, celle-ci se retrouve illisible. Ce problème peut bien évidemment être corrigé en installant le codec sur le poste en question.

II) Choix et étude des codecs

Afin de choisir les codecs, il faut les étudier et voir lequel correspond le mieux à nos attentes.

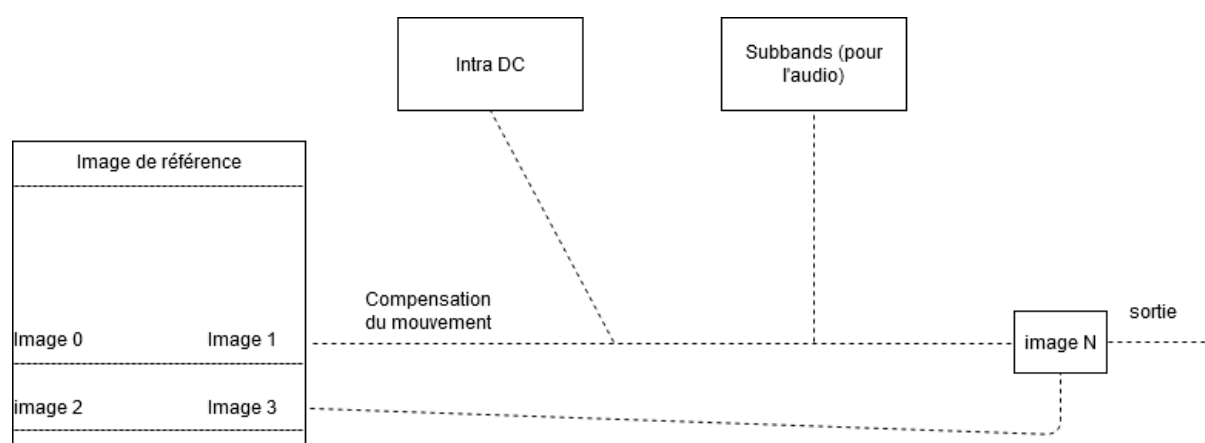
J'ai voulu partir sur des codecs qui m'étais inconnu afin de pouvoir les étudier.

Codecs choisis : Snow, hevc et h264

II.1) Snow

Le codec snow est un codec encore expérimental. Tout autre décodeur doit produire le même rendement pour un flux de snow conforme aux spécifications.

II.1.1) Fonctionnement



Une image inter-trame (Intra DC) spécifie une image qui appartient à un flux qui a été codé à l'aide d'un algorithme de prédiction inter-trame. Cela dépend de l'image précédemment encodée afin de prédire la position du macrobloc d'origine.

Le codeur de gamme mis en œuvre est une version adaptée basée sur un algorithme permettant de supprimer la redondance d'un message numérisé.

Les symboles codés par le codeur de plage de Snow sont des bits (0|1). Le site

Les probabilités associées ne sont pas fixes mais changent en fonction de la combinaison de symboles vu jusqu'à présent.

Un tableau contenant la table de transition existe :

```

state_transition_table = {
    0,  0,  0,  0,  0,  0,  0,  0, 20, 21, 22, 23, 24, 25, 26, 27,
    28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 37, 38, 39, 40, 41, 42,
    43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 56, 57,
    58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73,
    74, 75, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88,
    89, 90, 91, 92, 93, 94, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103,
    104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 114, 115, 116, 117, 118,
    119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 133,
    134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149,
    150, 151, 152, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164,
    165, 166, 167, 168, 169, 170, 171, 171, 172, 173, 174, 175, 176, 177, 178, 179,
    180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 190, 191, 192, 194, 194,
    195, 196, 197, 198, 199, 200, 201, 202, 202, 204, 205, 206, 207, 208, 209, 209,
    210, 211, 212, 213, 215, 215, 216, 217, 218, 219, 220, 220, 222, 223, 224, 225,
    226, 227, 227, 229, 229, 230, 231, 232, 234, 234, 235, 236, 237, 238, 239, 240,
    241, 242, 243, 244, 245, 246, 247, 248, 248,  0,  0,  0,  0,  0,  0,  0,  0};

```

Exemple Si on voit le bit 0 : 256 – state_transition_table[256 – état précédent)

II.2) H264

Le H264 est une norme de codage vidéo développée par l'UIT-T Q.6 / SG16 Groupe d'experts en codage vidéo (VCEG).

Son fonctionnement se base sur le facteur de débit constant (CRF).

Le contrôle du débit détermine le nombre de bits qui seront utilisés pour chaque trame. Cela déterminera la taille du fichier et comment la qualité est distribuée

II.2.1) Fonctionnement

Le H264 contient de nombreuses nouvelles technologies qui lui permettent de compresser la vidéo plus efficacement que les normes précédentes. Il utilise l'inter-prédiction, l'estimation pour les mouvements (comme le snow). Il utilise également un système d'intra-prédiction (cette fonctionnalité sera détaillée dans le HEVC)

Il offre également une précision pour les mouvements avec des blocs de tailles différentes (16 × 16, 16 × 8, 8 × 16, 8 × 8, 8 × 4, 4 × 8, 4 × 4). Ceci augmente la précision de la zone de mouvement.

II.3) HEVC

Le HEVC ou H265 est créé par le même groupe que celui qui a créé le H264. Celui-ci a un objectif simple, faire pareil que le H264 mais avec une réduction du taux de compression de moitié.

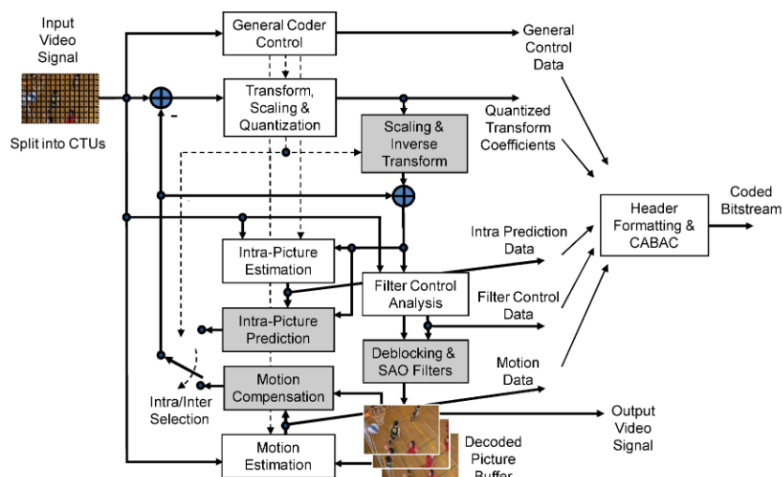
II.3.1) Fonctionnement

Le fonctionnement du H.265 repose globalement sur les principes du H.264 mais améliorés. Tout démarre par le codage d'une image Intra. L'encodage de cette image ne référencera aucune autre image.

L'image à traiter sera découpée en blocs de pixels. Dans le codec précédent, nous découpons uniquement l'image à traiter en un macrobloc de 16 pixels et un macrobloc chroma de 8 pixels. Le codec H.265 utilise le principe des unités d'arbre de codage (CTU). Ces blocs d'arbre de codage (CTB) sont généralement de grande taille (16, 32 ou 64) et sont sélectionnés par l'encodeur.

L'image interne sera codée à l'aide d'un ensemble de prédictions spatiales à l'intérieur de l'image traitée. L'image inter-images ne sera pas codée, mais sera uniquement décrite en utilisant le bloc de prédiction (calculé à partir de l'image I) auquel le vecteur de mouvement est appliqué. Ensuite, le codeur génère un signal de prédiction inter-images, qui sera principalement transmis sous forme de données auxiliaires en utilisant la compensation de mouvement.

Ensuite, le codeur génère un signal de prédiction inter-images, qui sera principalement transmis sous forme de données auxiliaires en utilisant la compensation de mouvement. Enfin, la DCT (Transformation en Cosinus Discret) est appliquée au signal de prédiction (inter-trame ou intra-trame), puis le résultat est envoyé avec les informations de prédiction. Pour améliorer la qualité et éliminer les artefacts et autres erreurs de prédiction, l'encodeur décode l'image qu'il vient d'encoder, puis les compare et utilise ces différences pour finaliser le double flux final.



III) Comparaison

Pour pouvoir faire les traitements nécessaires, les comparaisons nous allons nous servir d'outils différents tels que : FFMPEG, Excel, Matlab.

Ces 3 codecs sont des codecs vidéo avec et sans perte. Pour l'étude j'ai utilisé donc les mêmes paramètres aux niveau de la perte.

La vidéo de base s'appelle Basket.mov.

Il va donc falloir l'encoder dans chacun des codecs pour pouvoir comparer le tout.

Exemple de commande:

```
ffmpeg -i Basket.mov -c:v snow -preset slow -crf 22 -c:a copy snow.mkv
```

```
ffmpeg -i Basket.mov -c:v h264 -preset ultrafast -crf 0 h264_ultrafast.mkv
```

```
ffmpeg -i Basket.mov -c:v hevc -preset ultraslow -crf 0 hevc_ultrafast.mkv
```

Je ne mentionnerais pas toute les commandes car certaines ont été faites sans que je les sauvegarde, et je ne souviens plus de la syntaxe précise.

En général les paramètres importants sont :

- vcodec : pour le codec vidéo
- b : pour le bitrate
- r : pour le nombre de fps (ou d'ips en français)
- s : si on veut redimensionner la vidéo
- preset : l'option de rendu, en fonction des codecs
 - ultra fast
 - super fast
 - Very fast
 - faster
 - Fast
 - Medium - préréglage par défaut
 - Slow
 - slower
 - Very slow
 - Placebo (mais inutile)

III.1) Etude des codecs

Pour étudier les codecs et leurs effets, plusieurs données vont être utilisées :

- Le rendu final, qui sera jugé à l'œil (cependant c'est subjectif),
- Le PSNR, étudié en cours qui va se révéler très utile ici,
- Le SSIM, qui va mesurer la similarité entre des images numériques.
- Le VMAF, qui est un algorithme d'évaluation de la qualité perceptuelle de la vidéo développé par Netflix.

Le PSNR a été calculé de 2 manières :

- Via matlab, sur une image précise en commun à chaque vidéo
- Via la commande FFMPEG : `ffmpeg -i Basket.mov -i vidéo de comparaison -lavfi psnr -f null -`

Pour le SSIM et le VMAF, j'ai dû installer un module externe trouvé sur github.

(<https://github.com/Netflix/vmaf>)

Une fois qu'il est installé, il calcule différentes données pour 2 vidéos (une de référence et une de comparaison)

Cependant le VMAF n'est pas disponible sur tout types de codec et de compression, car si la compression est trop forte donc la valeur calculée sera erronée.

Pour les études suivantes, je me suis basé sur le PSNR de la vidéo plutôt que l'image, car le PSNR de la vidéo est une moyenne du PSNR de chaque image, ce qui donne une valeur plus juste.

Le VMAF est combinée à l'aide d'une régression basée sur SVM pour fournir un score de sortie unique dans la plage de 0 à 100 par image vidéo, où la qualité de 100 est la même que la qualité de la vidéo de référence. Ensuite, une moyenne arithmétique est utilisée pour grouper par temps ces scores sur toute la séquence vidéo pour fournir un score d'opinion moyen de différence globale.

Le SSIM lui est calculé de cette façon :

$$SSIM(x, y) = l(x, y) \cdot c(x, y) \cdot s(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_x\sigma_y + c_2)(cov_{xy} + c_3)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)(\sigma_x\sigma_y + c_3)}$$

Il permet de mesurer la qualité visuelle de l'image compressée par rapport à l'image d'origine.

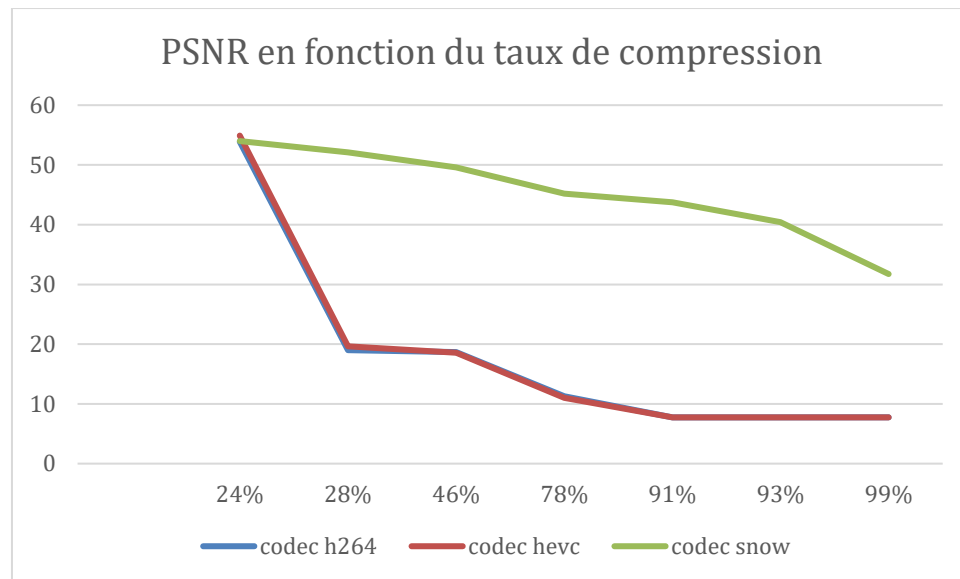
L'avantage du SSIM par rapport au PSNR est qu'il mesure la similitude structurelle entre deux images, tandis que le PSNR mesure la différence entre des pixels.

L'hypothèse de base est que l'œil humain est plus sensible aux changements dans la structure de l'image.

Codec	Taux de compression	Rendu final	Valeurs calculées
H264	Fort	Images très flou, perte de qualité visuelle	PSNR : 7.74763 VMAF : Pas calculable SSIM : Pas calculable
	Moyen	Moins de fluidité de la vidéo	PSNR : 18.649713 VMAF : 19.061918 SSIM : 0.0818187
	Faible	Aucune différence perceptible à l'œil nu	PSNR : 53.875575 VMAF : 19.46914 SSIM : 0.0868261
HEVC	Fort	Images flou, perte de qualité visuelle	PSNR : 7.73862 VMAF : Pas calculable SSIM : Pas calculable
	Moyen	Aucune différence perceptible à l'œil nu	PSNR : 18.587586 VMAF : 19.072309 SSIM : 0.0818187
	Faible	Aucune différence perceptible à l'œil nu	PSNR : 54.908652 VMAF : 19.21814 SSIM : 0.0867506
SNOW	Fort	Image floue lors de mouvement	PSNR : 31.73177 VMAF : 7.86367 SSIM : 0.0813893
	Moyen	Vidéo très nette, quelque saut d'image	PSNR : 49.578586 VMAF : 15.713589 SSIM : 0.0883128
	Faible	Aucune différence perceptible à l'œil nu	PSNR : 53.989848 VMAF : 18.402891 SSIM : 0.0870224

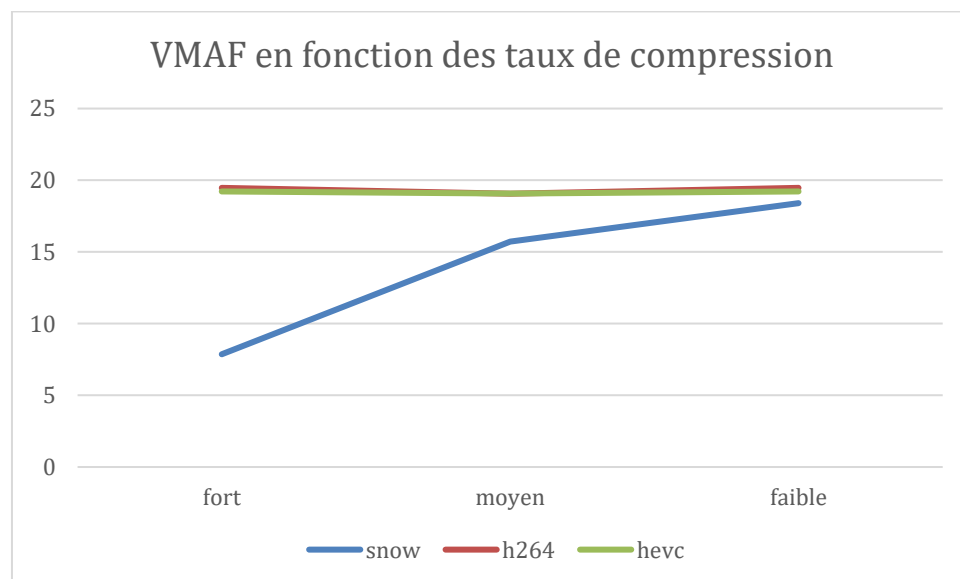
Pour mieux comprendre ces valeurs et les mettre en comparaison j'ai réalisé plusieurs graphiques sur Excel :

Le premier graphique compare le PSNR des 3 codec en fonction du taux de compression.



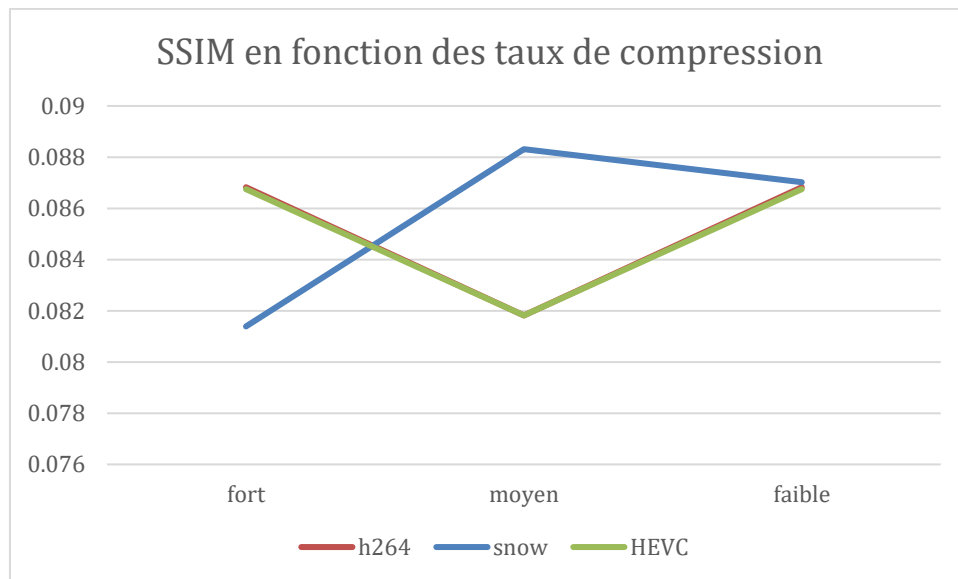
On remarque que le PSNR du codec SNOW est assez stable, tandis que le PSNR des codec HEVC et H264 est faible. Comme ces 2 codecs utilisent un fonctionnement similaire, on remarque que les valeurs sont très proches.

Maintenant en comparant les taux VMAF, on remarque que ces taux sont constants chez les codec H264 et HEVC, ce qui n'est pas le cas pour le codec SNOW.



Ceci nous montre qu'à un taux de compression fort, le score du SNOW est assez faible. Il est similaire aux autres codecs pour un taux de compression faible.

Maintenant comparons les SSIM des différentes vidéos.



Comme pour les résultats précédents, on remarque que le H264 et le codec HEVC se suivent dans les données, ce qui n'est pas étrange. Cependant le codec SNOW lui à un faible score pour une compression forte, mais pour le reste, il reste supérieur aux 2 autres codecs.

IV) Conclusion

Pour conclure sur ces 3 codecs, nous avons pu remarquer que le codec H264 et HEVC ont des données similaires, car ils fonctionnent sur le même algorithme, la seule différence est le taux de compression. En effet, les vidéos compresser de la même façon avec les mêmes paramètres sont toutes plus légères avec le codec HEVC.

Pour visualiser les vidéos SNOW j'ai dû installer KDenLive, tandis que tout les autres codecs sont visualisables sur VLC.

Le VMAF permet également de trouver avec le score, si la vidéo s'affichera bien sur téléphone, ou sur petit écran. (C'est pour cela que Netflix a dû développer cette mesure)

Pour un taux de compression fort, le SNOW offre un meilleur rendu visuel. Cependant, différentes mesures telles que le PSNR, VMAF et SSIM montrent qu'au niveau du traitement, le codec SNOW est celui qui a le pire rendu.

Pour un taux de compression moyen, le snow est le codec qui rend le mieux la vidéo visuellement et au niveau des mesures également.

Pour un taux de compression faibles, les 3 codecs se valent, mais le snow reste un peu inférieur avec le VMAF, mais possède un meilleur PSNR et un meilleur SSIM.

Pour résumer, il existe une multitude de codec. Ils apportent des avantages et des inconvénients.

On ne peut pas dire qu'il existe un bon codec ou un mauvais, cela dépend de l'utilisation final.

Sur ces codecs, si on souhaite stocker une vidéo sur ordinateur, sans la transférer sur une autre machine, le SNOW se révèle plus utile car il offre des taux de compression intéressant tout en ayant un rendu similaire à l'original.

Si nous souhaitons transférer des vidéos sur mobile, ou PSP par exemple, le codec HEVC se verra plus utile car il offre des vidéos plus légères que le H264, et le score en VMAF est meilleur pour ce codec que le SNOW.