

ffmpeg, la boîte à outils multimédia

 linuxtricks.fr/wiki/ffmpeg-la-boite-a-outils-multimedia

Table des matières

1. [Introduction](#)
2. [Commandes](#)



Introduction

ffmpeg est un logiciel très puissant de lecture et encodage de vidéo, s'utilisant en ligne de commande. Il existe des interfaces graphiques, mais ici, je parlerai de la ligne de commande.

Commandes

Quelques commandes tordues qui m'ont "sauvé la vie" se trouvent ci-dessous.

Que puis-je lire ? et encoder ?

Hé oui, cela ne sert à rien d'utiliser ffmpeg si le format qu'on souhaite traiter n'est pas pris en charge 😊

La commande suivante répondra à nos interrogations :

Code BASH :

```
ffmpeg -formats
```

Pivoter une vidéo de 90°

Vous avez pris une vidéo avec votre smartphone préféré en mode portrait, et quand vous lisez celle-ci sur votre PC, il faut tourner la tête.

Voici la commande magique (exemple réalisé avec un fichier mp4)

Code BASH :

```
ffmpeg -i ori.mp4 -vf "transpose=1" ret.mp4
```

Valeurs possibles de transpose:

0 = 90CounterClockwise and Vertical Flip (default)

1 = 90Clockwise

2 = 90CounterClockwise

3 = 90Clockwise and Vertical Flip

Pour cela, rien de plus simple :

Code BASH :

```
ffmpeg -i video.mp4 audio.mp3
```

Par défaut, la qualité est de 128kb/s Pour changer cette valeur utiliser **-ab** :

Code BASH :

```
ffmpeg -i video.mp4 -ab 256k son.mp3
```

Capturer l'écran de son bureau

Il est aussi possible d'enregistrer son bureau :

Code BASH :

```
ffmpeg -f x11grab -r 30 -s 1920x1080 -i :0.0 -vcodec libx264 out.mkv
```

-s 1920x1080 : correspond à la résolution en partant du coin supérieur gauche

Et avec le microphone en HD :

Code BASH :

```
ffmpeg -f alsa -ac 2 -i pulse -f x11grab -r 30 -s 1920x1080 -i :0.0 -acodec pcm_s16le -vcodec libx264 bureau.mkv
```

Encore plus optimisé pour le jeu :

Code BASH :

```
ffmpeg -threads 0 -async 30 -f alsa -i pulse -f x11grab -s 1920x1080 -r 30 -i :0 -vcodec libx264 -preset superfast -crf 16 -acodec libmp3lame -f mp4 video.mp4
```

Attention, il faut penser à régler convenablement l'entrée audio avec **pavucontrol** (microphone ou mixeur)

Convertir une vidéo mp4 en avi en conservant la même qualité

Pour conserver la même qualité, utiliser le paramètre **-qscale** :

Code BASH :

```
ffmpeg -i video.mp4 -qscale 0 video.avi
```

Si vous êtes comme moi, à aimer les photos, parfois, on est moins réactif qu'en effectuant une vidéo.

Pour extraire les images d'une vidéo (à environ 25 images par secondes), voici l'a ligne de commande à utiliser :

Code BASH :

```
ffmpeg -i video.mp4 -r 25 output_%04d.png
```

Couper une vidéo

Pour couper une scène de la vidéo, on peut utiliser **-ss** pour le début de la séquence et **-t** pour sa durée.

Par exemple, pour couper 30 secondes de vidéo à partir de 39 minutes 45 secondes (en conservant la qualité originale), on utilisera :

Code BASH :

```
ffmpeg -i sequenceacouper.avi -ss 00:39:45.00 -t 00:00:30.00 -c:v copy -c:a copy nouvellesequence.avi
```

Rogner une vidéo

Avec OpenShot, je ne peux exporter en HD qu'en format 16/9. Or, ma vidéo de base est en 4/3. Ca me créé donc 2 bandes noires de chaque côté. Voici l'astuce pour rogner la vidéo avec ffmpeg

La vidéo a été sortie en 1280x720. Je conserve les 720 de hauteur. La vidéo est centrée (Les 2 bandes noires sont de la même largeur).

$(720 \times 4) / 3 = 960$ au total.

Donc, j'ai $1280 - 960 = 320$ px à rogner (160 de chaque côté).

La commande que je vais utiliser est

Code BASH :

```
ffmpeg -i in.mp4 -filter:v "crop=out_w:out_h:x:y" out.mp4
```

où

- **out_w** est la largeur de la vidéo finale
 - **out_h** est la hauteur de la vidéo finale
 - **x** et **y** spécifient les coordonnées du coin supérieur gauche du rectangle précédemment défini (les coordonnées 0:0 sont en haut à gauche de la vidéo)
- Dans mon exemple, j'ai donc utilisé :

Code BASH :

```
ffmpeg -i in.mp4 -filter:v "crop=960:720:160:0" out.mp4
```

Convertir une vidéo en forçant un bitrate

Ici j'ai une vidéo en MP4, je veux la convertir en OGG mais en ne dépassant pas un bitrate vidéo de 512k et audio de 96k :

Code BASH :

```
ffmpeg -i CLDM.mp4 -b:v 512k -b:a 96k CLDM-2.ogg
```

Réduire le nombres d'images par secondes

Quand je fais des vidéos parfois il n'est pas nécessaire d'avoir un grand nombre d'images par secondes.

Voici une astuce pour réduire les FPS à la conversion :

Code BASH :

```
ffmpeg -i CLDM.mp4 -r 9 CLDM.ogg
```

-r 9 correspond à peu près à 25 FPS, pourquoi, je ne sais pas. La console l'indique de toute façon :

Code RUBY :

```
frame= 3645 fps= 25 q=0.0 Lsize= 12018kB time=00:06:04.55 bitrate= 270.1kbits/s dup=0 drop=7273
```

Concaténer plusieurs vidéos

Il m'arrive de faire des vidéos (avec le même appareil) ou avec SimpleScreenRecorder (donc au même format) et de vouloir les concaténer.

Voici la méthode utilisée. Créer un fichier texte avec tous les noms dans l'ordre des vidéos à concaténer :

Code BASH :

```
vi fic.txt
```

Code TEXT :

```
file 'out-2016-06-16_17.49.16.ogg'  
file 'out-2016-06-16_18.04.20.ogg'  
file 'out-2016-06-16_18.12.01.ogg'
```

Ensuite, exécuter la commande suivante pour tout concaténer :

Code BASH :

```
ffmpeg -f concat -i fic.txt -c copy out.ogg
```

Et le tour est joué !

Supprimer le son d'une vidéo

Code BASH :

```
ffmpeg -i video.mp4 -c copy -an video-sansson.mp4
```

Ajouter une piste audio à une vidéo

Voici une astuce pour ajouter une piste audio. On a une vidéo sans son par exemple et on a la piste son à côté. Pour poser l'audio sur la vidéo sans réencoder, procéder ainsi :

Code BASH :

```
ffmpeg -i video.mp4 -i audio.mp3 -c copy -map 0:0 -map 1:0 video_finale.mp4
```

Augmenter ou diminuer le volume de la bande son d'une vidéo sans réencoder

Si on se rend compte après montage que le son n'est pas très fort dans une vidéo, on peut monter celui-ci sans réencoder la vidéo.

Voici comment utiliser ffmpeg pour monter le son de 10 décibels :

Code BASH :

```
ffmpeg -i video.mp4 -vcodec copy -af "volume=10dB" video-augmentee.mp4
```

On peut aussi baisser le son :

Code BASH :

```
ffmpeg -i video.mp4 -vcodec copy -af "volume=-5dB" video-diminuee.mp4
```

Faire une vidéo avec un fichier JPG et un son MP3

Si on veut créer une vidéo avec une piste audio (MP3) et y inclure une image statique (JPG par exemple), procéder comme ceci :

Code BASH :

```
ffmpeg -loop 1 -i image.jpg -i son.mp3 -shortest -c:v libx264 -c:a copy video.mp4
```

Convertir des fichiers en lots

Exemple ici de conversion de fichiers mp3 en ogg :

Code BASH :

```
for f in *.mp3
do
    ffmpeg -i "$f" "${f/.mp3/.ogg}"
done
```