

FFmpeg

Cours d'approfondissement

Reto Kromer, Eléonore Bernard et Kamilla Ødegård
AV Preservation by reto.ch et Ødegård & Bernard Restaurierung
Berne, le 14 mars 2025

1

Déroulement du cours

- Objectifs du cours
- Présentation des problèmes/questions apportées
- Rappel rapide des notions de base de FFmpeg
- Input sur le flux de travail, le contrôle qualité et la préservation
- Travail de groupe
- Présenter les résultats et les conclusions
- Utiliser FFmpeg dans d'autres applications
- Discussion finale

2

Objectifs du cours

- Appliquer FFmpeg pour différentes conversions de sons et d'images
- Utiliser les ressources et les outils relatifs à FFmpeg
- Trouver les commandes appropriées
- Appliquer FFmpeg et d'autres programmes pour l'extraction et l'interprétation de métadonnées
- Options de contrôle de qualité
- Utilisation des commandes FFmpeg pour le traitement par lots

3



Link zum FFmpeg-Aufbaukurs (D)
<https://reto.ch/training/2025/2025-03-13/>



Lien vers le cours avancé de FFmpeg (F)
<https://reto.ch/training/2025/2025-03-14/>

4

Exemples d'application

Berne, le 14 mars 2025

5

Exemples

- « Ripper » des fichiers à partir de DVD et les convertir en copie de visualisation
- Conversion d'images DPX en flux d'images Matroska/FFV1
- Conversion d'images DPX individuelles à l'aide de RAWcooked
- Conversion de fichiers non-compressé/master en copie de visualisation H.264
- Convertir ProRes en copie de visualisation H.264
- Ajouter un filigrane, un timecode, un logo ou un générique

6

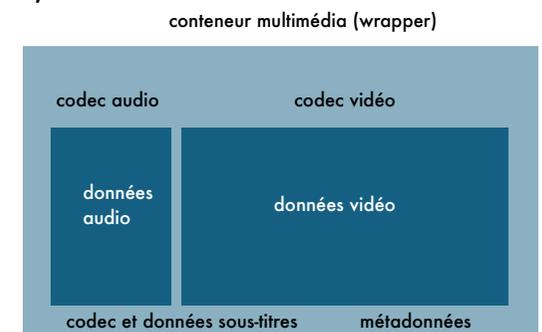
Rappel

Berne, le 14 mars 2025

7

Structure des fichiers

- conteneur multimédia (wrapper)
- codec audio
- données audio
- codec vidéo
- données vidéo



8

La famille FFmpeg

programmes

- ffmpeg
- ffplay
- ffprobe

bibliothèques

- libavutil
- libavcodec
- libavformat
- libavdevice
- libavfilter
- libswscale
- libswresample
- libpostproc

9

démultiplexage : libavformat

décodage : libavcodec

modification : libavfilter

codage : libavcodec

multiplexage : libavformat

10

Structure des commandes FFmpeg

```
$0      $1      ${n}  
command argument_1 ... argument_n
```

Syntaxe des arguments :

```
-parameter  
-parameter value  
-p  
-p value
```

11

Syntaxe FFmpeg

```
ffmpeg  
[global_options]  
[input_options_n] -i input_file_n  
[output_options_n] output_file_n
```

```
ffprobe [input_options] input_file
```

```
ffplay [input_options] input_file
```

12

Ressources, chercher de l'aide

FFmpeg Cookbook for Archivists

→ avpres.net/FFmpeg/

ffmprovizr

→ amiaopensource.github.io/ffmprovizr/

ffmpeg -h

ffmpeg -codecs

13

Input sur le flux de travail

Berne, le 14 mars 2025

14

Flux de travail possible

- Définir les formats de fichiers acceptés
- Contrôler la qualité des fichiers
- Créer des dérivés (par ex. copie de visualisation)
- Créer des paquets d'archives
- Sauvegarde dans une solution d'archivage numérique (par ex. bandes magnétiques LTO, serveur d'archive, etc.)

15

Input sur le contrôle de la qualité

Berne, le 14 mars 2025

16

Cas pratique

Couper le fichier

- MKV/FFV1 converti en MP4/H.264
- MP4 coupé avec la commande suivante :
→ `ffmpeg -i inputfile.mp4 -ss hh:mm:ss -to hh:mm:ss -c:v copy -c:a copy outputfile.mp4`
- Contrôle de qualité par visionnement dans VLC/QuickTime/Gridplayer
→ La vidéo est lue/interprétée différemment
- La même commande fonctionne bien pour un fichier MKV/FFV1

17

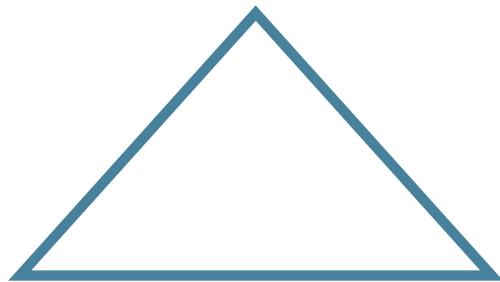
Cas pratique

Couper le fichier

- Contrôle de qualité dans VLC/QuickTime/Gridplayer
→ La vidéo est lue/interprétée différemment
- Explication : les différents programmes interprètent la compression différemment
La synchronisation audio/vidéo n'est donc pas assurée.
- Solution : Préciser la commande
→ `ffmpeg -accurate_seek -i inputfile.mp4 -ss hh:mm:ss -to hh:mm:ss -c:v libx264 -preset veryslow -crf 18 -pix_fmt yuv420p -c:a aac outputfile.mp4`

18

qualité de l'image



temps d'encodage

taille du fichier

19

Contrôle de qualité

Proposition de flux de travail

- Noms de dossiers/fichiers et structure
- Sommes de contrôle, intégrité des données
- Vérifier les métadonnées techniques
 - Format de fichier : conteneur, codec, sous-titres, timecode, etc.
- Analyser les signaux
- Visualiser et/ou écouter le fichier
 - Contenu de l'image et du son

20

Nom de fichier et structure

- Vérifier l'archivage des données et leur désignation selon la convention de dénomination interne.
 - Cela dépend des archives et des processus internes !
- Les fichiers peuvent-ils être clairement attribués ?
- L'archivage des données et la documentation permettent-ils de reconnaître d'où proviennent les fichiers et quel est leur statut (master d'archivage, copie de visualisation, etc.) ?

21

Sommes de contrôle

Cryptographique

- MD5
- SHA-1
- SHA-256
- SHA-512

Non cryptographique

- CRC-32
- xxHash 32
- xxHash 64
- xxHash 128

22

Sommes de contrôle

Outils de création de somme de contrôle MD5

- MD5 (CLI, Mac)
- Checksum+ (GUI, Mac)

- MD5Checker (Windows)
<http://getmd5checker.com/>

23

Sommes de contrôle

Commandes

- MD5
 - **Mac** : `md5 input_file`
 - **Windows** : `certutil -hashfile input_file MD5`
 - `md5sum`
- SHA
 - `shasum / sha1sum / sha256sum / sha512sum`
- Framemd5
 - `ffmpeg -i input_file -f framemd5 output_file_framemd5.txt`

24

Commande pour comparer des fichiers

- Sur Linux/Mac/Windows Terminal ou WSL
→ `diff -s file01.txt file02.txt > file_diff.txt`
- Sur Windows
→ `fc file01.txt file02.txt > file_diff.txt`

Peut par exemple être utilisé pour comparer des fichiers `MediaInfo.txt` ou `Framemd5.txt`

25

Programmes d'analyse

Analyse des métadonnées

- FFprobe (CLI)
- MediaInfo (GUI, CLI)

Analyse des signaux

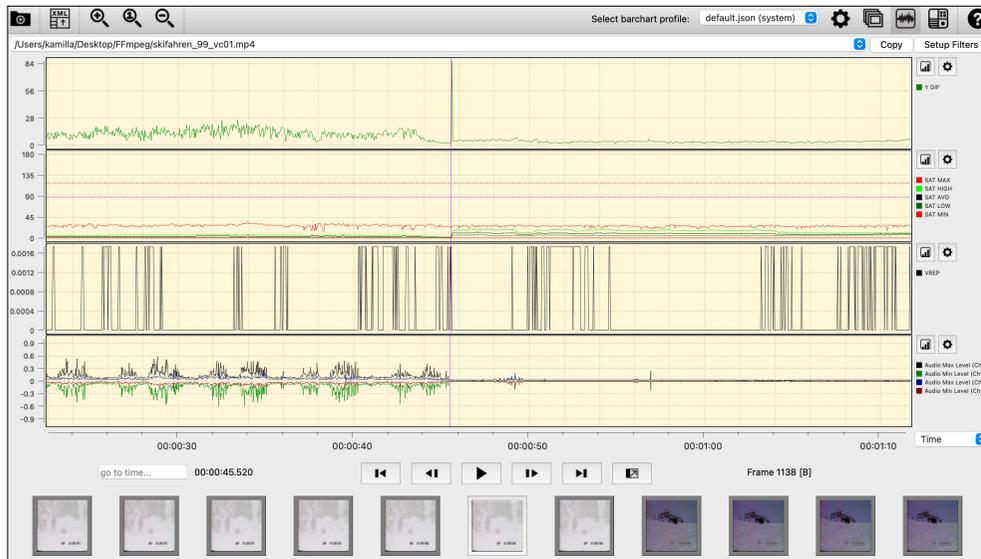
- QCTools (GUI)
- qcli (CLI)
- SignalServer (Web App)

```
Allgemein
Vollständiger Name      : skifahren_1999.mov
Format                  : MPEG-4
Format-Profil           : QuickTime
Codec-ID                : qt 0000.02 (qt )
Dateigröße              : 2,37 GiB
Dauer                   : 1 min 31s
Gesamte Bitrate         : 223 Mb/s
Bildwiederholungsrate   : 25,000 FPS
Kodierendes Programm    : Lavf61.7.100

Video
ID                      : 1
Format                  : YUV
Codec-ID                : v210
Codec-ID/Hinweis       : AJA Video Systems Xena
Dauer                   : 1 min 31s
Bitraten-Modus          : konstant
Bitrate                 : 221 Mb/s
Breite                  : 720 Pixel
Clean aperture width    : 783 Pixel
Höhe                   : 576 Pixel
Clean aperture height   : 576 Pixel
Bildseitenverhältnis   : 4:3
Clean aperture display aspect ratio : 4:3
Modus der Bildwiederholungsrate : konstant
Bildwiederholungsrate   : 25,000 FPS
Standard                : PAL
Color space             : YUV
Chroma subsampling      : 4:2:2
Bit depth               : 10 bits
Scantyp                 : Interlaced
Scan type, store method : Interleaved fields
Scanreihenfolge         : oberes Feld zuerst
Compression mode        : Lossless
Bits/(Pixel*Frame)      : 21,333
Stream-Größe            : 2,34 GiB (99%)
Sprache                 : Englisch
Color primaries         : BT.601 PAL
Transfer characteristics : BT.709
Matrix coefficients     : BT.601

Audio
ID                      : 2
Format                  : PCM
```

26



27

Contrôle de qualité avec FFprobe et MediaInfo

Commandes

- FFprobe (CLI)
→ `ffprobe -show_format -show_streams -print_format json input_file > input_file.txt`
- MediaInfo (CLI)
→ `mediainfo file.ext`
→ `mediainfo - file.ext`
→ `mediainfo --Details=1 file.ext`
→ `mediainfo --Output=JSON file.ext`

28

Programmes de visualisation de fichiers AV

Lecteurs de média AV basé sur FFmpeg

- FFplay (CLI)
→ ffmpeg.org
- MPV (GUI, CLI)
→ mpv.io
- VLC (GUI)
→ www.videolan.org/vlc
- Gridplayer (GUI)
→ <https://github.com/vzhd1701/gridplayer>

29

Commandes pour l'analyse de fichiers AV

Commandes

- Écran divisé (<https://avpres.net/FFmpeg/split.html>)
→ `ffmpeg -i input_file_1 -i input_file_2 -filter_complex "[0] crop=iw/2:ih:0:0 [left]; [1] crop=iw/2:ih:iw/2:0 [right]; [left][right] hstack [out]" -map "[out]" output_file`
- Différentiel de deux fichiers (<https://avpres.net/FFmpeg/delta.html>)
→ `ffmpeg -i input_file_1 -i input_file_2 -filter_complex "[1] format=yuva444p, lut=c3=128, negate [1_with_alpha]; [0] [1_with_alpha] overlay [out]" -map "[out]" output_file`

30

Input sur la conservation et les formats de fichiers

Berne, le 14 mars 2025

31

Conservation

Éléments analogiques

- Numériser avec la meilleure qualité possible
- Continuer à préserver l'élément analogique
- La profondeur de bits est plus importante que la définition

Éléments numériques

- Conserver autant que possible le format natif
- Ne pas convertir dans un format « supérieur »
- Le ProRes natif est adapté à l'archivage

32

SMPTE REGISTERED
DISCLOSURE DOCUMENT

SMPTE RDD 36:2015

Apple ProRes Bitstream Syntax
and Decoding Process



Page 1 of 39 pages

The attached document is a Registered Disclosure Document prepared by the sponsor identified below. It has been examined by the appropriate SMPTE Technology Committee and is believed to contain adequate information to satisfy the objectives defined in the Scope, and to be technically consistent.

This document is NOT a Standard, Recommended Practice or Engineering Guideline, and does NOT imply a finding or representation of the Society.

Every attempt has been made to ensure that the information contained in this document is accurate. Errors in this document should be reported to the proponent identified below, with a copy to eng@smpte.org.

All other inquiries in respect of this document, including inquiries as to intellectual property requirements that may be attached to use of the disclosed technology, should be addressed to the proponent identified below.

Proponent contact information:

ProRes Program Office
Apple Inc.
1 Infinite Loop, MS: 77-2YAK
Cupertino, CA 95014
USA

Email: ProRes@apple.com

33

Formats d'archives d'images individuelles

- Dossier, TIFF, 2K/4K, RGB, 4:4:4:4, 16 bit
- MXF (OP 1a), DPX, 2K/4K, R'G'B', 4:4:4:4, 12/10 bit
- MKV, FFV1, 2K/4K, R'G'B', 4:4:4:4, 12/10 bit

34

Formats d'archives de flux d'images

- MKV, non-compressé, SD/HD, Y'C_BC_R 4:2:2, 10 bit
- MKV, FFV1, SD/HD, Y'C_BC_R 4:2:2, 10 bit

35

Formats d'archives de télévision numérique

Productions SD

- MXF (OP 1a), MPEG IMX, PAL, 50 Mbit/s

Productions HD

- MXF (OP 1a), XDCAM HD 422, PAL, 50 Mbit/s

D'autres formats de bandes numériques doivent également être lus dans le codec natif (par ex. Digital Betacam, DV, etc.)

36

Travail de groupe

Berne, le 14 mars 2025

37

Exemples

- « Ripper » des fichiers à partir de DVD et les convertir en copies de visualisation
- Conversion d'images DPX en flux d'images Matroska/FFV1
- Conversion d'images DPX individuelles à l'aide de RAWcooked
- Conversion de fichiers non-compressé/master en copie de visualisation H.264

Autres commandes à tester

- Convertir ProRes en copie de visualisation H.264
- Ajouter un filigrane, un timecode, un logo ou un générique

38

Travail de groupe

- Choisir et définir la question et l'objectif de l'exercice
- Rechercher, rassembler et tester les commandes
- Contrôler la qualité des résultats

Thèmes

- Sujet que vous avez apporté vous-même
- DVD
- Convertir DPX en MKV
- Convertir de la vidéo SD non-compressée, 422, entralacée en copie de visualisation
- Conversion de ProRes en copie de visualisation

39

Ressources

FFmpeg Cookbook for Archivists
→ avpres.net/FFmpeg/

ffmpegprovisr
→ amiaopensource.github.io/ffmpegprovisr/

Afficher la liste de paramètres existants

```
ffmpeg -h
ffmpeg -codecs
ffmpeg -decoders
ffmpeg -h decoder=flac
ffmpeg -encoders
ffmpeg -h encoder=ffv1
ffmpeg -filters
ffmpeg -formats
ffmpeg -layouts
ffmpeg -sample_fmts
ffmpeg -pix_fmts
ffmpeg -bsfs
```

40

Input sur les DVD / supports de données optiques

- Que sont les supports de données optiques ?
 - CD (CD-DA, CD-ROM, CD-R, CD-RW)
 - DVD (DVD-Video, DVD-Audio, DVD-ROM, DVD+R, DVD-R, DVD+RW, DVD-RW..)
 - Laserdisc, WORM, Blu-ray, M-Disc, MiniDisc etc.
- L'identification est importante pour pouvoir choisir le bon flux de travail pour la sauvegarde

41

Input sur les DVD / supports de données optiques

- Analyse d'un support de données optique
 - S'agit-il d'un DVD vidéo ? Quelle est la structure des données ?
 - Analyse de métadonnées à l'aide de programmes
- Ripper et/ou imager le contenu d'un DVD ?
 - Image de disque (archiver les contenus AV, y compris la structure du disque)
 - Extraire les fichiers AV et, si nécessaire, les réencoder dans un fichier unique

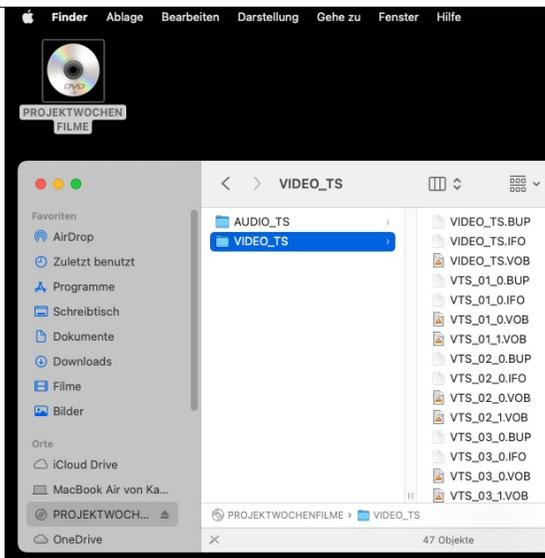
42

DVD vidéo - Structure

Données

- BUP - fichier de backup
- IFO - informations
- VOB - fichier AV
- Les contenus AV sont divisés en plusieurs fichiers VOB

Illustration : Structure des données d'un DVD brûlé



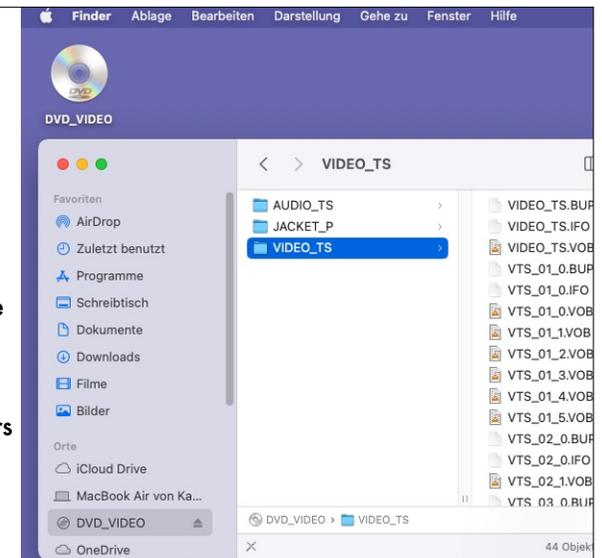
43

DVD vidéo - Structure

DVD commerciaux

- Peuvent être protégés
- Gestion des droits d'auteurs ?
- Contiennent le même type de fichiers
- Les contenus AV sont divisés en plusieurs fichiers VOB

Illustration : Structure des données d'un DVD commercial



44

Exercice : Transcoder des fichiers DVD

- Extraire des fichiers AV et les transcoder en un fichier unique
 - Quels sont les fichiers à prendre en compte pour le transcodage ?
 - Comment créer un fichier vidéo à partir de plusieurs fichiers ?
 - Comment convertir correctement les fichiers en MP4/H.264 ?

45

Commandes : DVD

- Convertir le contenu AV d'un DVD en fichier MP4

```
→ ffmpeg -i "concat:VTS_01_1.VOB|VTS_01_2.VOB|VTS_01_3.VOB" -b:v 1500k -r 30 -vcodec h264 -strict -2 -acodec aac -ar 44100 -f mp4 output_file.mp4
```

```
→ ffmpeg -i "concat:input_file_1|input_file_2|input_file_3" -c:v libx264 -c:a aac output_file.mp4
```

```
→ prendre en compte tous les canaux audio et vidéo : ffmpeg -i "concat:input_file_1|input_file_2" -map 0:v -map 0:a -c:v libx264 -c:a aac output_file.mp4
```

46

Commandes : DVD

- Convertir le contenu AV d'un DVD en fichier MP4

→ désentrelacer :

```
ffmpeg -i "concat:VTS_02_1.VOB|VTS_03_1.VOB|VTS_04_1.VOB" -filter:v "idet, bwdif" -c:v libx264 -c:a aac output_H264.mp4
```

→ désentrelacer et mise à l'échelle en HD :

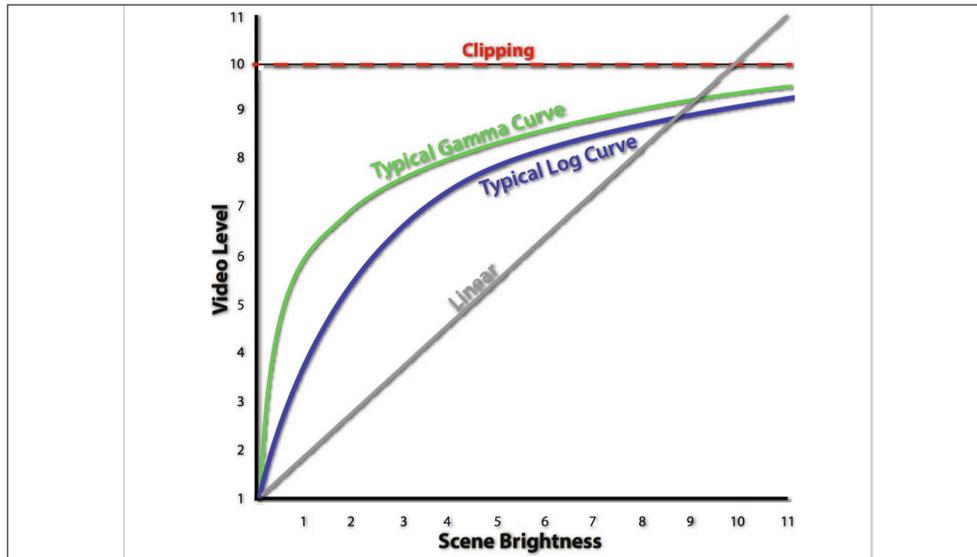
```
ffmpeg -i "concat:VTS_01_1.VOB|VTS_01_2.VOB|VTS_01_3.VOB" -filter:v "idet, bwdif, scale=1440:1080:flags=lanczos, pad=1920:1080:240:0" -c:v libx264 -c:a aac output_H264.mp4
```

47

Input sur les fichiers DPX

- Les fichiers DPX (Digital Picture Exchange) sont générés par des caméras ou des scanners haut de gamme lors de la prise de vue ou de la numérisation
- Que contient un DPX ?
 - Encodage log-négatif
 - Codage log RGB ou codage semi-log
 - Codage gamma ou codage par fonction de puissance
 - Codage linéaire de scène

48



49

```

Licensed DPX flavors:
Yes DPX/Raw/RGB/8bit/U/LE
Yes DPX/Raw/RGB/10bit/U/LE/FilledA
Yes DPX/Raw/RGB/10bit/U/BE/FilledA
Yes DPX/Raw/RGB/12bit/U/LE/FilledA
Yes DPX/Raw/RGB/12bit/U/BE/Packed
Yes DPX/Raw/RGB/12bit/U/BE/FilledA
Yes DPX/Raw/RGB/16bit/U/LE
Yes DPX/Raw/RGB/16bit/U/BE
Yes DPX/Raw/RGBA/8bit/U/LE
Yes DPX/Raw/RGBA/10bit/U/LE/FilledA
Yes DPX/Raw/RGBA/10bit/U/BE/FilledA
Yes DPX/Raw/RGBA/12bit/U/LE/FilledA
Yes DPX/Raw/RGBA/12bit/U/BE/Packed
Yes DPX/Raw/RGBA/12bit/U/BE/FilledA
Yes DPX/Raw/RGBA/16bit/U/LE
Yes DPX/Raw/RGBA/16bit/U/BE
Yes DPX/Raw/Y/8bit/U/LE
Yes DPX/Raw/Y/10bit/U/BE/FilledA
Yes DPX/Raw/Y/10bit/U/BE/FilledB
Yes DPX/Raw/Y/12bit/U/BE/Packed
Yes DPX/Raw/Y/16bit/U/LE
Yes DPX/Raw/Y/16bit/U/BE

Licensed TIFF flavors:

```

50

Exercice : Fichiers DPX

- Convertir des images individuelles DPX en fichier vidéo (MKV)
- Fichiers d'exercice (images individuelles) :
 - DPX_10-bit/Test_000nnnnn.dpx oder DPX_12-bit/Test_000nnnnn.dpx
 - Source : Scène de film 16 mm, générée dans DaVinci Resolve
- Fichiers d'exercice (flux d'images) :
 - Test_DNxHR_444_12-bit.mxf

51

Commandes : Fichiers DPX

- Commandes FFmpeg :

→ Pour images individuelles : `ffmpeg -f image2 -framerate 24 -i input_file_%08d.dpx -c:v ffv1 -level 3 -threads 8 -coder 1 -context 1 -g 1 -slices 24 -slicecrc 1 -start_number 00086400 -c:a copy output_file.mkv`

→ Pour flux d'images : `ffmpeg -i input_file.mxf -c:v ffv1 -level 3 -threads 8 -coder 1 -context 1 -g 1 -slices 24 -slicecrc 1 -c:a copy output_file.mkv`

52

DPX - RAWcooked

- RAWcooked permet d'encoder des DPX en Matroska avec FFV1/FLAC
- Possibilité d'intégrer des métadonnées dans un fichier sidecar avec RAWcooked

• RAWcooked : <https://mediaarea.net/RAWcooked>

53

Input sur ProRes

- Les fichiers ProRes sont générés par de nombreuses caméras et présentent alors la meilleure qualité possible de ce contenu
- De la même façon, si les fichiers traités numériquement (FinalCut, Adobe Premiere, DaVinci Resolve) sont exportés en ProRes, alors il s'agit de la meilleure qualité possible de ce contenu
- Les codecs ProRes peuvent se trouver dans les conteneurs MOV, MXF ou MKV Container.
- Que faut-il prendre en compte ?

54

Input sur ProRes

La famille ProRes-422

- ProRes 422 HQ (high quality)
- ProRes 422 (standard)
- ProRes 422 LT (light)
- ProRes Proxy

La famille ProRes-4444

- ProRes 4444 XQ
- ProRes 4444

La famille ProRes-RAW

- ProRes RAW HQ
- ProRes RAW

55

Commandes : Transcoder un fichier ProRes en une copie de visualisation

- Exemple : fichier ProRes apporté par vous-même
- Analyser les métadonnées du fichier avant le transcodage

• Commande FFmpeg :

```
→ ffmpeg -i inputfile.mov -pix_fmt yuv420p -c:v libx264 -preset veryslow -crf 30 -movflags +faststart+write_colr outputfile_ProRes_H264.mp4
```

• Source : https://avpres.net/FFmpeg/im_H264.html

56

Exercice : Transcoder un fichier non-compressé en copie de visualisation

- Exemple 1 : skifahren_99.mov
 - Source : cassette Video8 numérisée
 - Format : MOV/V210; non-compressé, YUV, 4:2:2, entrelacé, PAL, 10 bit
- Que faut-il prendre en compte ?
 - Entrelacé vers progressif / désentrelacement souhaité ?
 - Conserver le 4:3 ou transcoder en 16:9 ?

57

Commandes : Transcoder un fichier non-compressé en copie de visualisation

Exemple 1 : non-compressé, V210, 4:2:2, entrelacé, PAL, 10 bit

- Commandes FFmpeg :

→ **ffmpeg** -i input_file.MOV -c:v libx264 -preset veryslow -crf 18 -pix_fmt yuv420p -c:a aac output_file.MP4

→ désentrelacement : **ffmpeg** -i input_file -c:v libx264 -pix_fmt yuv420p -filter:v "idet, bwdif" output_file

→ 4:3 vers 16:9 avec pillarbox : **ffmpeg** -i input_file -c:v libx264 -filter:v "yadif, scale=1440:1080:flags=lanczos, pad=1920:1080:(ow-iw)/2:(oh-ih)/2, format=yuv420p" output_file

58

Autres possibilités avec FFmpeg

- Changer le mode de balayage - désentrelacer (entrelacé vers progressif)
- Couper le fichier (voir cas d'étude)
- Modifier l'espace colorimétrique (quand cela serait-il utile ?)
- Insérer un filigrane / générique de début ou fin / logo / timecode

59

Changer le mode de balayage

→ désentrelacement **ffmpeg** -i input_file -c:v libx264 -pix_fmt yuv420p -filter:v "idet, bwdif" output_file

→ désentrelacement **ffmpeg** -i input_file -c:v libx264 -vf "yadif,format=yuv420p" output_file

Quelles différences peut-on constater ?

60

Input - Changer l'espace colorimétrique

- Modifier la matrice des couleurs
- Quand faut-il modifier la matrice de couleurs ?
- Comment reconnaître la matrice de couleurs actuelle dans les métadonnées ?

Modifier les espaces colorimétriques

Quand faut-il modifier les espaces colorimétriques ?

61

Commandes : Changer l'espace colorimétrique

- Modifier la matrice des couleurs :

→ `ffmpeg -i input_file -c:v libx264 -vf colormatrix=src:dst output_file`

→ Exemple Rec. 601 vers Rec. 709 : `ffmpeg -i input_file -c:v libx264 -vf colormatrix=bt601:bt709 output_file`

Espace colorimétrique pour PAL :

→ `ffmpeg -i input_file -c:v libx264 -color_primaries bt470bg -color_trc bt709 -colorspace bt470bg output_file`

Espace colorimétrique pour NTSC :

→ `ffmpeg -i input_file -c:v libx264 -color_primaries smpte170m -color_trc bt709 -colorspace smpte170m output_file`

62

Modifier les propriétés d'un fichier

- Ajouter un filigrane, un timecode, un générique ou un logo
- Au préalable, il faut clarifier quelles polices sont installées sur l'ordinateur et connaître le chemin d'accès de la police souhaitée afin de l'utiliser

→ Commande pour vérifier les polices installées

Sur macOS :

```
> ls /Library/Fonts
```

Sur Windows :

```
> dir \Windows\Fonts
```

63

Filigrane

- Insérer un filigrane

• https://amiaopensource.github.io/ffmprovizr/#text_watermark

→ `ffmpeg -i input_file -vf drawtext="fontfile=font_path:fontsize=font_size:text=watermark_text:fontcolor=font_color:alpha=0.4:x=(w-text_w)/2:y=(h-text_h)/2" output_file`

→ Exemple : `ffmpeg -i input_file -filter:v "drawtext=text='watermark':fontfile='/Library/Fonts/Arial.ttf':fontsize=35;fontcolor=white:alpha=0.25:x=(w-text_w)/2:y=(h-text_h)/2" output_file`

64

Logo

Insérer un logo

- Le logo doit être disponible en tant que fichier PNG
- Le logo doit être beaucoup plus petit que l'image vidéo

→ **ffmpeg** -i input_file.mp4 -i logo.png -filter_complex "overlay=10:main_h-overlay_h-10" with_logo.mp4

65

Timecode

- Insérer un timecode

→ **ffmpeg** -i input_file -filter:v drawtext="timecode=starting_timecode:rate=timecode_rate:fontfile=font_path:fontsize=font_size:fontcolor=font_colour:box=1:boxcolor=box_colour:x=(w-text_w)/2:y=h/1.2" output_file

→ **Exemple** : **ffmpeg** -i input_file.mp4 -filter:v drawtext="timecode='01\:00\:00\:00':rate=25:fontfile='/Library/Fonts/Arial.ttf':fontsize=35:fontcolor=white:x=(w-text_w)/2:y=h/1.2" with_timecode.mp4

66

Générique de début ou fin

- Insérer un générique de début ou fin
 - Doit exister en tant que fichier vidéo et avoir les mêmes caractéristiques que le fichier vidéo à traiter

67

Modifier les propriétés d'un fichier

- Ajouter des sous-titres au format SRT

→ **ffmpeg** -i output_archive.mkv -filter:v "subtitles=subtitles.srt" -c:v libx264 -preset veryslow -crf 18 -pix_fmt yuv420p -c:a aac -ar 48k -movflags +faststart+write_colr output_streaming.mp4

68

Input sur le format SRT

- Fichier de sous-titres SubRip
- Le fichier texte contient le texte des sous-titres dans l'ordre correct avec le timecode de début et de fin
- Il existe des outils pour générer des sous-titres. On peut aussi les créer avec un éditeur de texte

Le format SRT :

- Nombre entier consécutif
- Début du timecode --> fin du timecode
- Texte de l'intertitre
- Ligne vide

Exemple :

- 5
- 00:01:28,250 --> 00:01:30,500
- Ceci est un sous-titre
- sur deux lignes

69

Observations

- Qu'avez-vous testé ?
- Qu'avez-vous observé, appris ?
- Dans quelle mesure les fichiers changent-ils (métadonnées et visuel) ?
- Quelles sont les commandes qui conviennent le mieux à quoi ?

70

Input sur le travail par lots

Berne, le 14 mars 2025

71

Traitement par lots

Possibilités de traitement par lots avec des commandes FFmpeg

- FFCommand Engine
- HandBrake
- Audacity
- ShutterEncoder
- MediaConch (intégrer ffprobe)
- Scripts (Bash, Python etc.)

72

Traitement par lots

- Exemple d'application d'une commande à plusieurs fichiers
 - La commande doit être exécutée pour tous les fichiers se trouvant dans un dossier
- Exemple de combinaison de commandes
 - Exécuter plusieurs commandes dans un ordre précis sur le même fichier

73

Workflow pour le traitement par lots

Application d'une commande à plusieurs fichiers

- Préparer un dossier avec plusieurs fichiers de même type
- Quelle commande doit être exécutée ?

Application de plusieurs commandes dans un ordre précis sur un même fichier

- Réfléchir au workflow
- Combiner les commandes

74

FFcommand Engine

- Installer FFCommand Engine
- Lien : https://github.com/ColorlabMD/FFCommand_Engine
- Plusieurs commandes FFmpeg peuvent être importées, enregistrées dans des fichiers et exécutées les unes après les autres

75

Programmes

- Installer HandBrake
 - Lien : <https://handbrake.fr/>
 - Intégrer les commandes FFmpeg Commands dans HandBrake
- Installer Audacity
 - Lien : <https://www.audacityteam.org/download/>
 - Intégrer les commandes FFmpeg souhaitées dans Audacity
- ShutterEncoder
 - Lien : <https://www.shutterencoder.com/>

76

Observations

- Quels programmes avez-vous testé ?
- Quelles commandes avez-vous combinées ?

77

Merci pour votre attention !

- Feedback
- Que manquait-il ?

78

AV Preservation by reto.ch Ødegård & Bernard Restaurierung

Reto Kromer :

reto.ch / info@reto.ch

Eléonore Bernard et Kamilla Ødegård :

<https://atelier40a.ch/odegard-bernard-restaurierung-klg/>
eleonore.bernard@atelier40a.ch /
kamilla.oedegard@atelier40a.ch

79