

# FFmpeg Einsteiger Kurs

Reto Kromer, Eléonore Bernard und Kamilla Ødegård  
AV Preservation by reto.ch und Ødegård & Bernard Restaurierung  
Bern, 3. Februar 2025

1

## Einführung in den Kursablauf

- Ziele des Kurses
- Einführung in FFmpeg
- Input zu Dateien
- Input zur Kommandozeile
- Vor- und Nachteile von FFmpeg
  
- Eigenschaften von Dateien
- Praktische Anwendung in Gruppen
- Input zu Ressourcen
- Schlussbesprechung

2

## Ziele des Kurses

- Vor- und Nachteile von FFmpeg kennenlernen
- Verstehen, weshalb das Verständnis von Dateien für die Anwendung von FFmpeg wichtig ist
- Anwendungsmöglichkeiten von FFmpeg in der Erhaltung audiovisueller Dateien kennen lernen
- FFmpeg-Commands kennen lernen, verstehen und anwenden

3

## Einführung in FFmpeg

Reto Kromer, Eléonore Bernard und Kamilla Ødegård  
AV Preservation by reto.ch und Ødegård & Bernard Restaurierung  
Bern, 3. Januar 2024

4

## Was ist FFmpeg?

„A complete, cross-platform solution to record, convert and stream audio and video.“ - [www.ffmpeg.org](http://www.ffmpeg.org)

- Programm welches über die Kommandozeile bedient wird
- FFmpeg ist ein „Framework“ welches nahezu alle AV Dateien dekodieren, enkodieren, transcodieren, muxen, demuxen, streamen und abspielen kann.
- FFmpeg kann auf allen gängigen Betriebssystemen angewendet werden
- Mittlerweile gibt es viele Programme, die auf FFmpeg basieren (z. B. VLC, QCTools, vrecord)

5

## Anwendungen von FFmpeg

- Dateiumwandlung
  - Format, Codec verändern
- Dateien verändern
  - Eigenschaften der Dateien verändern
  - Einfügen von Vorspann oder Abspann
- AV-Dateien analysieren und Metadaten extrahieren
- Dateien abspielen

6

## FFmpeg-Befehlsstruktur

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

### FFmpeg-Syntax der Argumente:

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

7

## FFmpeg-Syntax

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

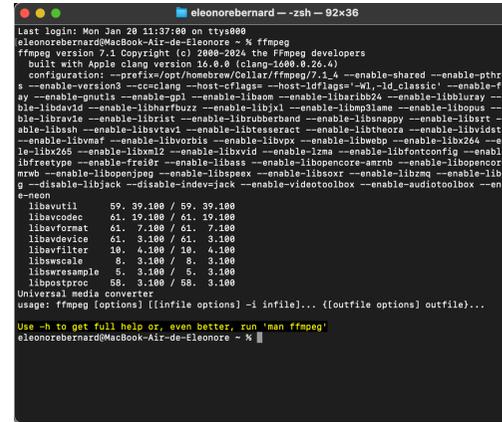
```
ffprobe [input_options] input_file
```

```
ffplay [input_options] input_file
```

8

# Benutzerschnittstellen

- Kommandozeile (command-line interface, CLI)
  - ffmpeg (CLI)
  - [ffmpeg.org](https://ffmpeg.org)
- Grafische Benutzeroberfläche (graphical user interface, GUI)
  - FFCommand Engine (GUI)
  - [github.com/ColorlabMD/FFCommand\\_Engine](https://github.com/ColorlabMD/FFCommand_Engine)



Screenshot ffmpeg



Screenshot FFCommand Engine

# Ressourcen, Hilfestellungen

- FFmpeg Cookbook for Archivists
  - [avpres.net/FFmpeg/](https://avpres.net/FFmpeg/)
- ffmprovisr
  - [amiaopensource.github.io/ffmprovisr/](https://amiaopensource.github.io/ffmprovisr/)
- ffmpeg -h
- ffmpeg -codecs

# Die FFmpeg Familie

- Programme
  - ffmpeg
  - ffplay
  - ffprobe
- Bibliotheken
  - libavutil
  - libavcodec
  - libavformat
  - libavdevice
  - libavfilter
  - libswscale
  - libswresample
  - libpostproc

demultiplexen: libavformat

decodieren: libavcodec

ändern: libavfilter

codieren: libavcodec

multiplexen: libavformat

13

## Programme, die auf FFmpeg basieren

### Mediaplayer

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

### Analysetools

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

### Digitalisierungssoftware

- vrecord

14

## Kommandozeile

Einführung in FFmpeg

Bern, 3. Februar 2025

15

## Was ist die Kommandozeile?

- Benutzerschnittstelle
  - Kommandozeile (CLI: command-line interface)
  - Graphische Benutzeroberfläche (GUI: graphical user interface)
- Wie findet man die Kommandozeile?
  - Linux: „Terminal“
  - Mac OS X: „Terminal“
  - Microsoft Windows: „Eingabeaufforderung“ („cmd“)
- **Befehle** können sich je nach Betriebssystem unterscheiden
- **Befehle** müssen korrekt eingegeben werden

16

## Basic Commands

- Arbeitsordner festlegen (change directory)
  - Mac/Linux/Windows: `cd Desktop`
  - Windows mit Onedrive: `cd OneDrive Desktop`
- In einen Ordner höher wechseln
  - Mac/Linux/Windows: `cd ..`
- Zeige Arbeitsordner (print working directory)
  - Mac/Linux: `pwd`
  - Windows: `ch` oder `chdir`

17

## Basic Commands

- Inhalt des Ordners anzeigen (list)
  - Mac/Linux: `ls`
  - Windows: `dir`
- Neuen Ordner erstellen
  - Mac/Linux: `mkdir <folder name or full path>`
  - Windows: `md <folder name or full path>`

18

## Wichtige Zeichen zur Eingabe

- Mac OS X (Achtung: kann tastaturabhängig sein)
  - ~ Tilde → Tastenkombination: `option + N`
  - \ Backslash → Tastenkombination: `alt / option + shift + 7`
- Windows (Achtung: kann tastaturabhängig sein)
  - ~ Tilde → Tastenkombination: `alt + N`
  - \ Backslash → Tastenkombination: `alt + 9 + 2 / option + shift + 7`

19

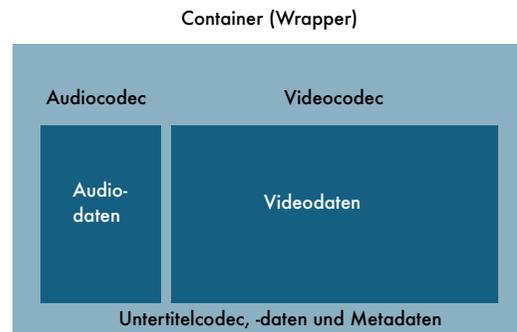
## Dateien

Einführung in FFmpeg  
Bern, 3. Februar 2025

20

## Dateiaufbau

- Container (Wrapper)
- Audiocodec
- Audiodaten
- Videocodec
- Videodaten



21

## Kompression

- nicht komprimiert
- verlustfrei komprimiert
- verlustbehaftet komprimiert
- komprimiert generiert
- Farbrunterabtastung

22

## Nicht komprimiert

- + Daten sind leichter zu bearbeiten
- + Software läuft schneller
- grössere Dateien
- langsames Schreiben, Übermitteln und Lesen der Dateien

Beispiele: TIFF, DPX, DNG, OpenEXR

23

## Verlustfrei komprimiert

- + kleinere Dateien
- + schnelleres Schreiben, Übermitteln und Lesen der Dateien
- Daten sind komplexer zu bearbeiten
- Software läuft langsamer

Beispiele: JPEG 2000, FFV1

24

## Verlustbehaftet komprimiert

- optimiert für Aufnahme und/oder Postproduktion
- optimiert für Zugang und Distribution

Beispiele Mezzanine:

ProRes 422, ProRes 4444, DNxHD, DNxHR

Beispiele Zugangsdatei:

H.264 (AVC), H.265 (HEVC), H.266 (VVC), AV1

25

## Komprimiert generiert

→ sowohl für Aufnahme als auch für Postproduktion optimiert

Beispiele: CineForm RAW, ProRes RAW, Blackmagic RAW

26

## Wichtige Grundsätze

Ein Archiv muss seine Dateien pflegen und handhaben können!

- Open Source Formate und Anwendungen
- einfache Bedienung und ausführliche Dokumentation
- weite Verbreitung der Formate

27

## Dateiformate

Formate für verschiedene Anwendungszwecke:

- Archivmasterformat
  - zur Erhaltung und Archivierung
- Mezzanine-Format
  - zur Bearbeitung und Postproduktion
- Distributionsformat
  - zur Verbreitung und Zugänglichmachung

28

## Archivmaster (heute)

### Film

- folder, TIFF, 2K, RGB, 4:4:4, 16 bit
- MXF, DPX, 2K, R'G'B', 4:4:4, 10 bit

### Video

- AVI, uncompressed, HD, Y'C<sub>B</sub>C<sub>R</sub>, 4:2:2, 10 bit
- MOV, uncompressed, HD, Y'C<sub>B</sub>C<sub>R</sub>, 4:2:2, 10 bit
- Matroska, FFV1, HD, Y'C<sub>B</sub>C<sub>R</sub>, 4:2:2, 10 bit

### Ton

- BWF, 96 kHz, 24 bit
- FLAC, 96 kHz, 24 bit

29

## Mezzanine (heute)

### Video

- ProRes 4444, 2K
- DNxHR, 2K
- ProRes 422 HQ, HD
- DNxHD 175x, HD

### Ton

- BWF, 48 kHz, 24 bit
- WAVE, 48 kHz, 24 bit

30

## Zugang (heute)

### MP4

### Video

- H.264, SD, yuv420p, „lossy“
- H.264, HD, yuv420p, „lossy“

### Ton

- AAC, 44.1 kHz, 16 bit
- AAC, 48 kHz, 16 bit

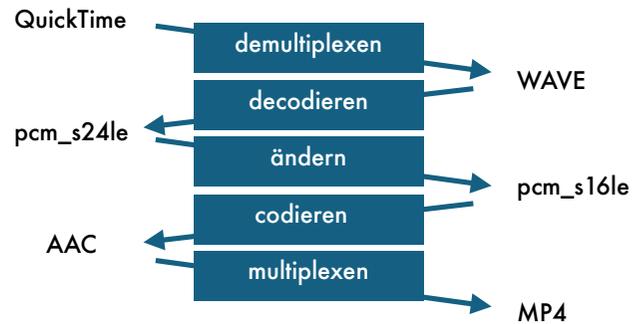
31

## Dateiumwandlungen

- Demultiplexen
- Decodieren
- Ändern
- Codieren
- Multiplexen

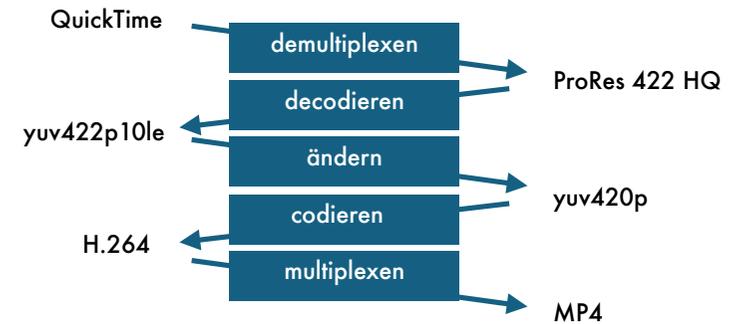
32

## Beispiel: Ton



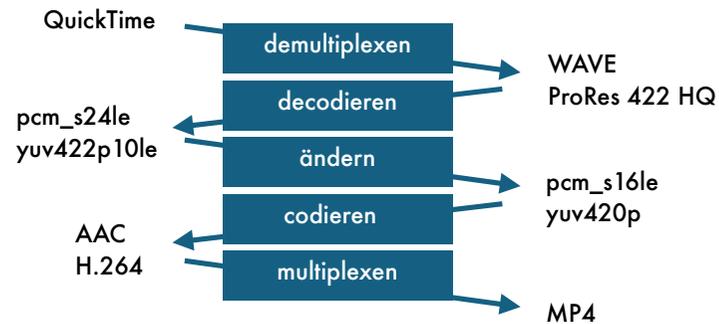
33

## Beispiel: Bild



34

## Beispiel: Bild und Ton



35

## Vor- und Nachteile von FFmpeg

Einführung in FFmpeg  
Bern, 3. Februar 2025

36

## Vorteile

- FFmpeg kann mit grosser Dateivielfalt umgehen
- Möglichkeit der Massenverarbeitung → Scripts
- Massgeschneiderte Lösungen → nicht eingeschränkt durch Programmvoreinstellungen
- Vielseitiges und leistungsstarkes Programm
- Bearbeitungsgeschwindigkeit
- Breite Nutzung und etabliertes Programm im Archivbereich
- FFmpeg-Commands können in einigen GUIs implementiert werden

37

## Nachteile

- Weniger benutzerfreundlich im Vergleich zu GUI
- Fehlerrisiko bei Unwissen und Tippfehler
- Risiken bei Benutzung der Kommandozeile (CLI)
  
- Vielseitiges und leistungsstarkes Programm
  - grosse Vielfalt an Commands
  - man muss sich genau mit Commands & Flags auseinandersetzen
  - oder sich auf Quellen wirklich verlassen können!

38

## FFmpeg Anwendungsbeispiele

Einführung in FFmpeg  
Bern, 3. Februar 2025

39

## Erste Anwendungsübungen

- Bilddatei erzeugen
- Bilddatei abspielen
- Tondatei erzeugen
- Tondatei abspielen
- Dateien zusammenführen

40

## Bilddatei erzeugen

```
ffmpeg  
-lavfi mandelbrot  
-t 10  
-c:v rawvideo  
-pix_fmt uyvy422  
mandelbrot.avi
```

41

## Bilddatei abspielen

```
ffplay  
mandelbrot.avi
```

42

## Tondatei erzeugen

```
ffmpeg  
-f lavfi -i "sine=frequency=440"  
-t 10  
-c:a pcm_s16le  
-ar 48k  
-ac 2  
la.wav
```

43

## Tondatei abspielen

```
ffplay  
la.wav
```

44

## Dateien zusammenfügen

```
ffmpeg  
-i mandelbrot.avi  
-i la.wav  
-c:v copy  
-c:a copy  
mandela.avi
```

45

## AV-Datei abspielen

```
ffplay  
mandela.avi
```

46

## Dateieigenschaften

Einführung in FFmpeg  
Bern, 3. Februar 2025

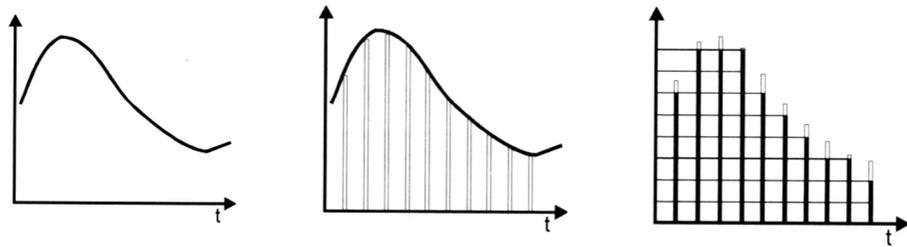
47

## Digitaler Ton

- Abtastung
- Quantisierung
- Kompression

48

## Digitaler Ton



Digitalisierung = Abtastung + Quantisierung

Abbildung: Agathe Jarczyk

49

## Abtastrate

- 44.1 kHz (CD Qualität)
- 48 kHz
- 96 kHz (gut geeignet für Archiv Master)
- 192 kHz

50

## Quantisierungsaufösung

- 16 bit ( $2^{16} = 65\,536$ )
- 24 bit ( $2^{24} = 16\,777\,216$ )
- 32 bit ( $2^{32} = 4\,294\,967\,296$ )

51

## Digitales Bild

- Bildauflösung
- Quantisierungsaufösung
- Linear, Potenzfunktion, logarithmisch
- Farbraum
- Kompression und Farbunterabtastung
- Normlicht

52

## Bildauflösung

- SD 480i / SD 576i
- HD 720p / HD 1080i
- 2K / HD 1080p
- 4K / UHD-1
- 8K / UHD-2
  
- Wofür stehen p und i?

53

## Interlaced oder progressive

- Interlaced → i
  - Anwendung bei analoger Monitortechnologie
  
- Progressive → p
  - Anwendung bei digitaler Monitortechnologie

54



55

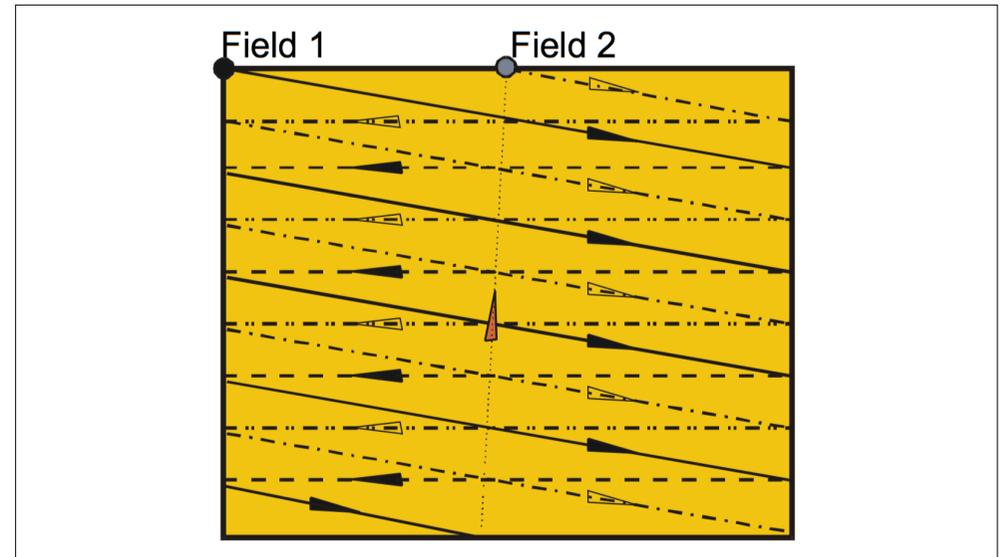


56

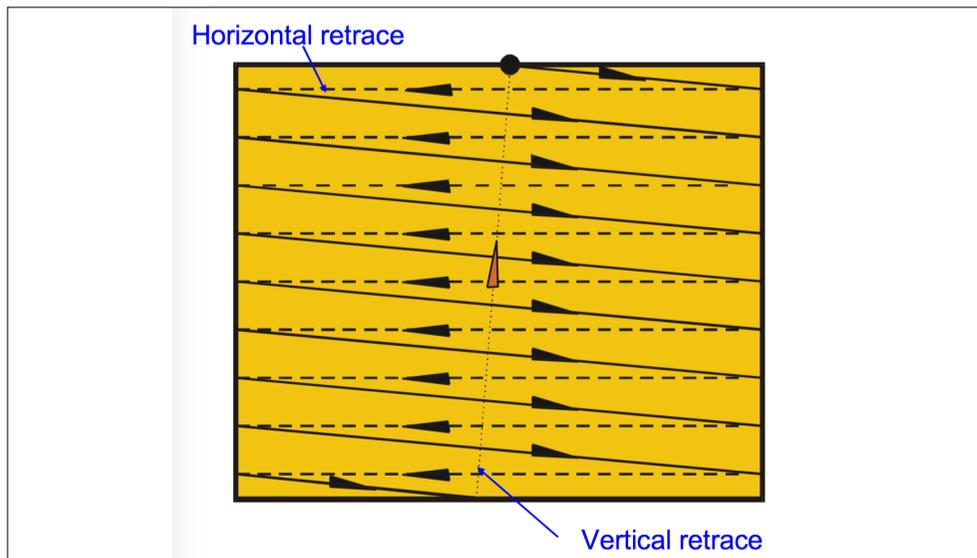
## Exkurs: Analoges Farbfernsehen

- PAL
- NTSC
- SECAM (französisch and solistisch)

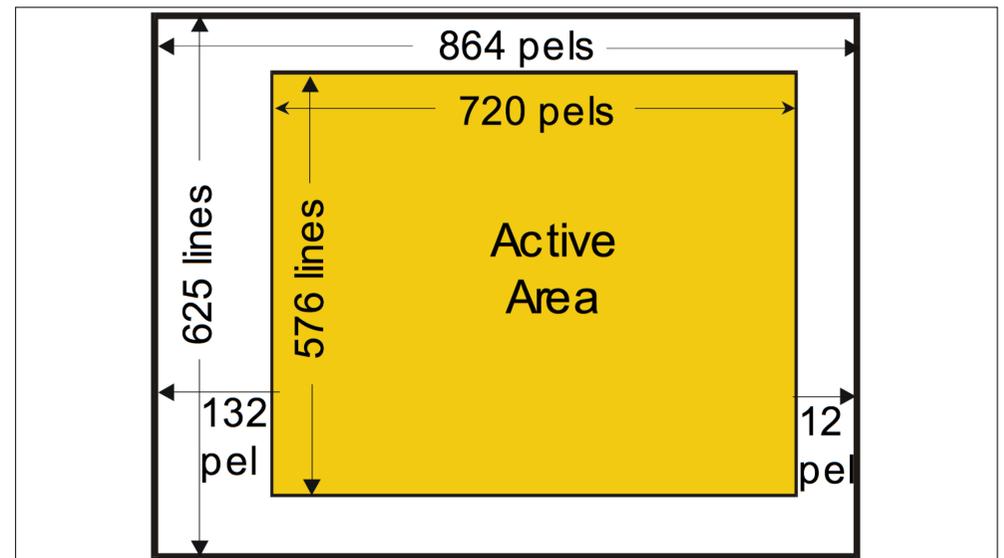
57



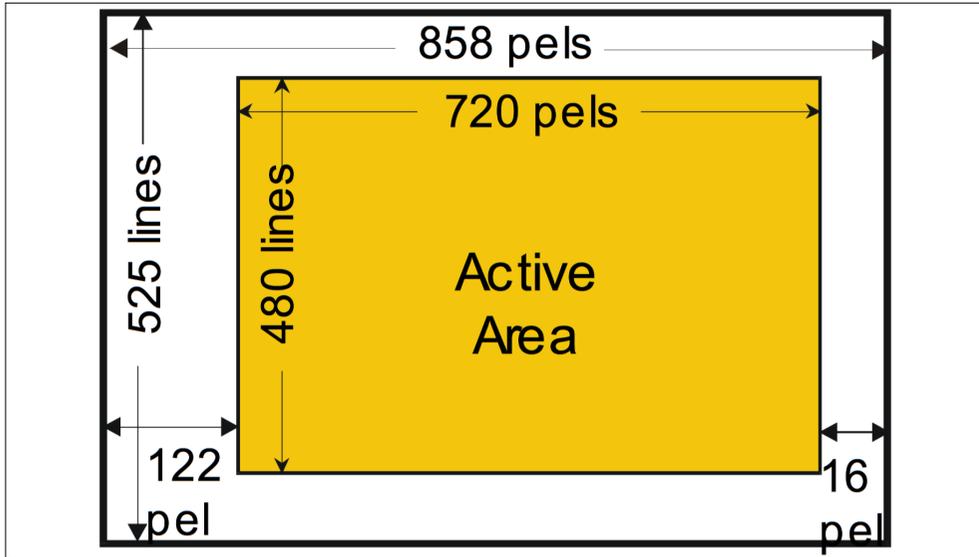
58



59



60



61



62



63



64



65



66

## Quantisierungsauflösung

- 8 bit ( $2^8 = 256$ )
- 10 bit ( $2^{10} = 1\,024$ )
- 12 bit ( $2^{12} = 4\,096$ )
- 16 bit ( $2^{16} = 65\,536$ )
- 24 bit ( $2^{24} = 16\,777\,216$ )

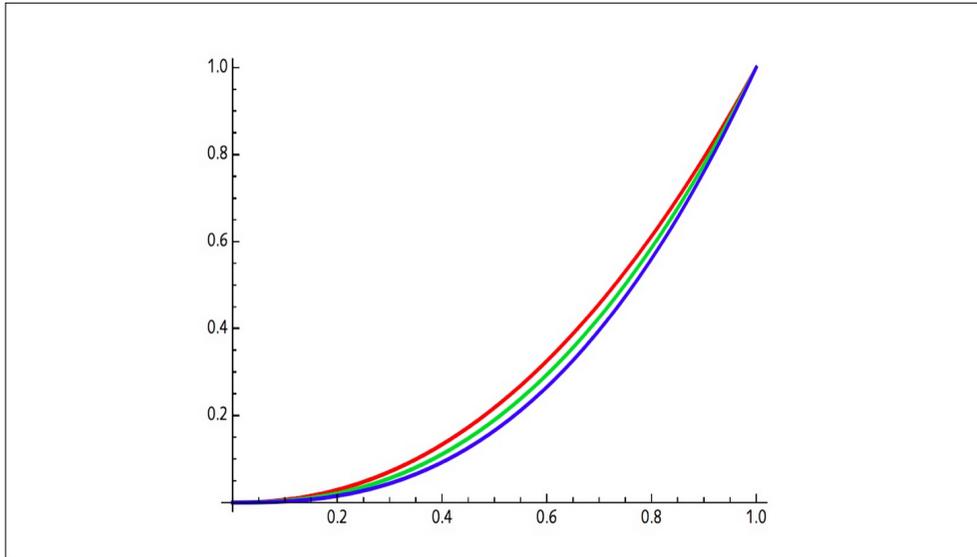
67

## Linear, Potenz, Logarithmus

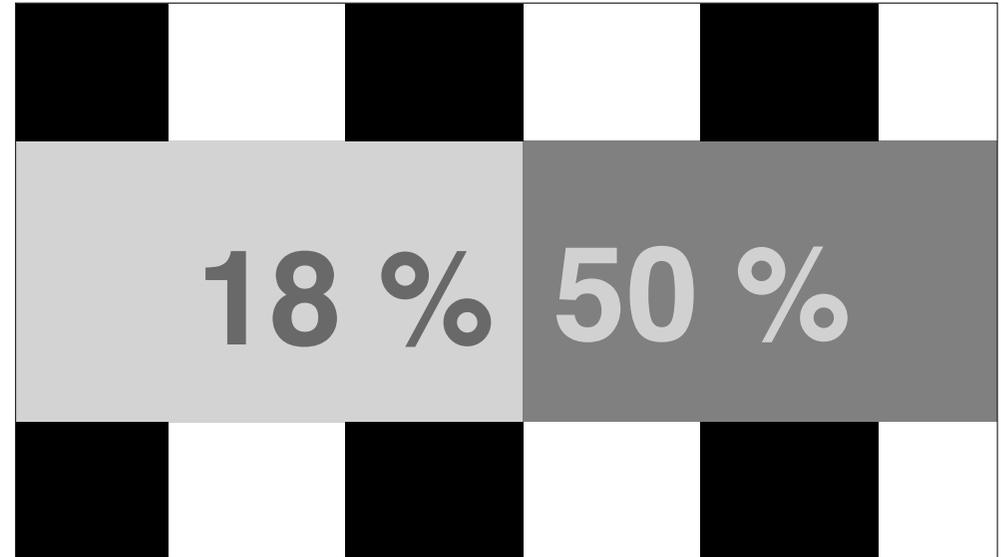
„Mittelgrau“

- Lineare Funktion: etwa 18 %
- Potenzfunktion: 50 %
- Logarithmusfunktion: 50 %

68



69

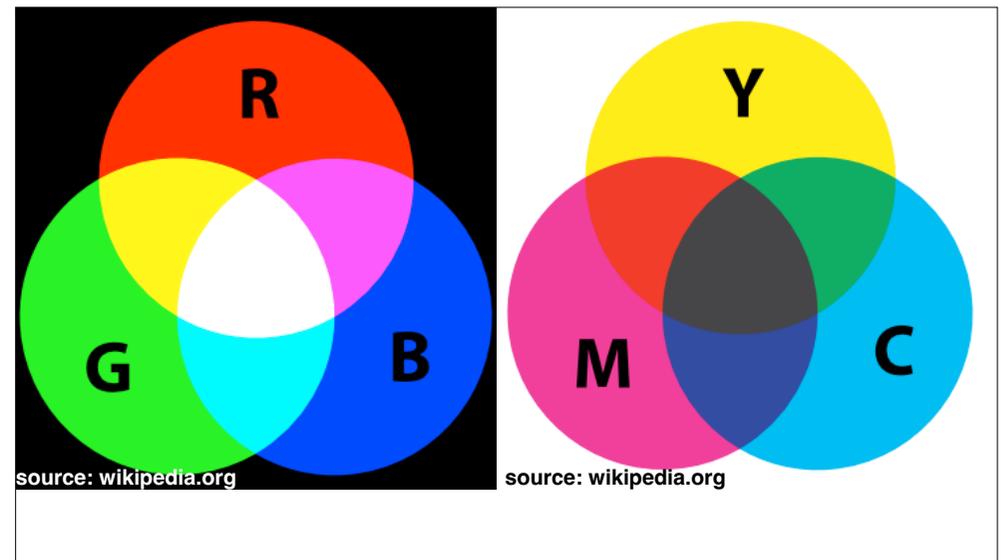


70

## Farbraum

- XYZ,  $L^*a^*b^*$
- RGB / R'G'B' / CMY / C'M'Y'
- Y'IQ / Y'UV / Y'D<sub>B</sub>D<sub>R</sub>
- Y'C<sub>B</sub>C<sub>R</sub> / Y'C<sub>O</sub>C<sub>G</sub>
- Y'P<sub>B</sub>P<sub>R</sub>

71



source: wikipedia.org

source: wikipedia.org

72

$$\begin{pmatrix} R' \\ G' \\ B' \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1.396523 \\ 1 & -0.342793 & -0.711348 \\ 1 & 1.765078 & 0 \end{pmatrix} \begin{pmatrix} Y' \\ C_B \\ C_R \end{pmatrix}$$

$$\begin{pmatrix} Y' \\ C_B \\ C_R \end{pmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ -0.168074 & -0.329965 & 0.498039 \\ 0.498039 & -0.417947 & -0.080992 \end{pmatrix} \begin{pmatrix} R' \\ G' \\ B' \end{pmatrix}$$

73

$$\begin{pmatrix} R' \\ G' \\ B' \end{pmatrix} = \begin{pmatrix} 1 & 1 & -1 \\ 1 & 0 & 1 \\ 1 & -1 & -1 \end{pmatrix} \begin{pmatrix} Y' \\ C_O \\ C_G \end{pmatrix}$$

$$\begin{pmatrix} Y' \\ C_O \\ C_G \end{pmatrix} = \begin{pmatrix} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & 0 & -\frac{1}{2} \\ -\frac{1}{4} & \frac{1}{2} & -\frac{1}{4} \end{pmatrix} \begin{pmatrix} R' \\ G' \\ B' \end{pmatrix}$$

74

## Farbraum

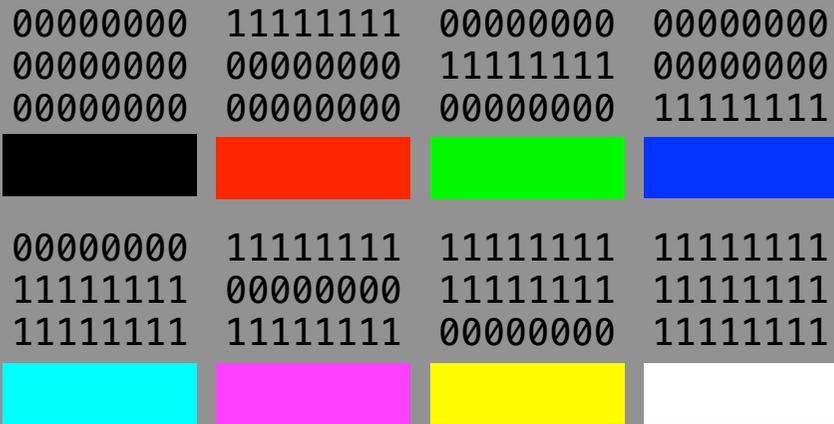
- XYZ, L\*a\*b\*
- RGB / R'G'B' / CMY / C'M'Y'
- Y'IQ / Y'UV / Y'D<sub>B</sub>D<sub>R</sub>
- Y'C<sub>B</sub>C<sub>R</sub> / Y'C<sub>O</sub>C<sub>G</sub>
- Y'P<sub>B</sub>P<sub>R</sub>

75



76

# RGB24

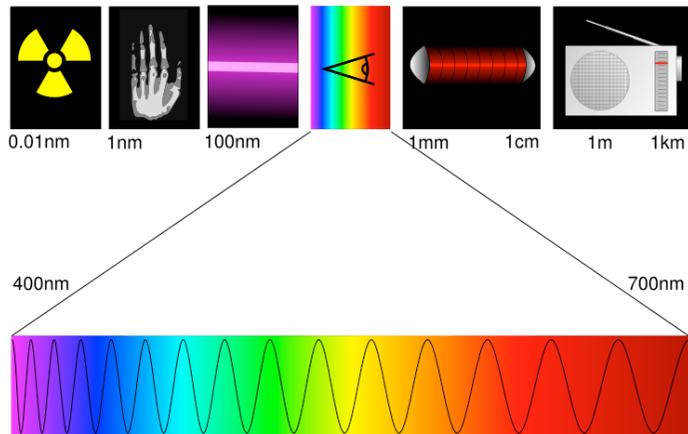


77

# Normlicht

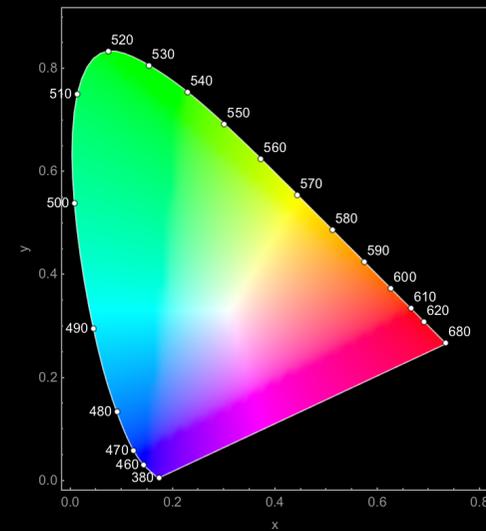
- D50
- D55
- D65
- D75

78

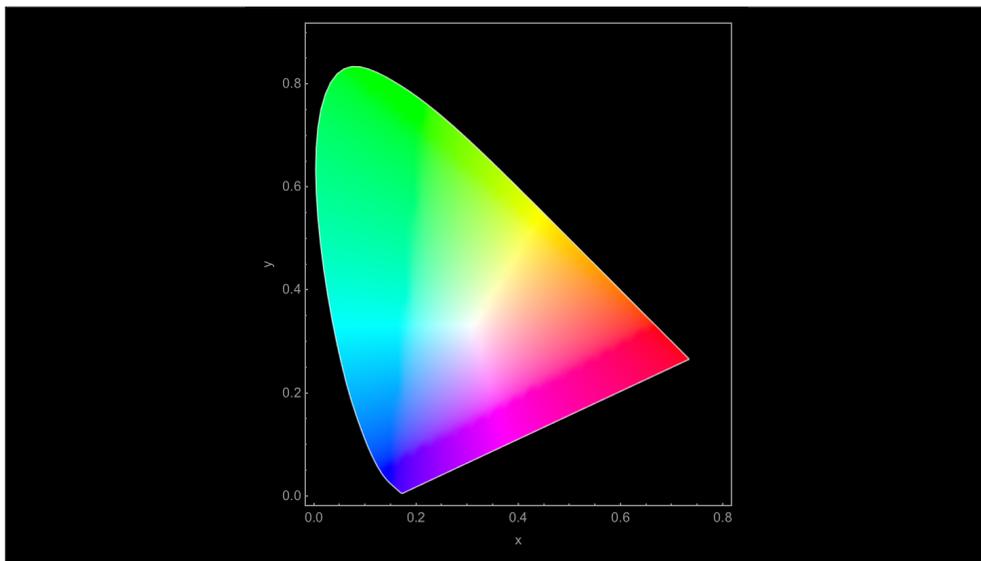


source: unknown

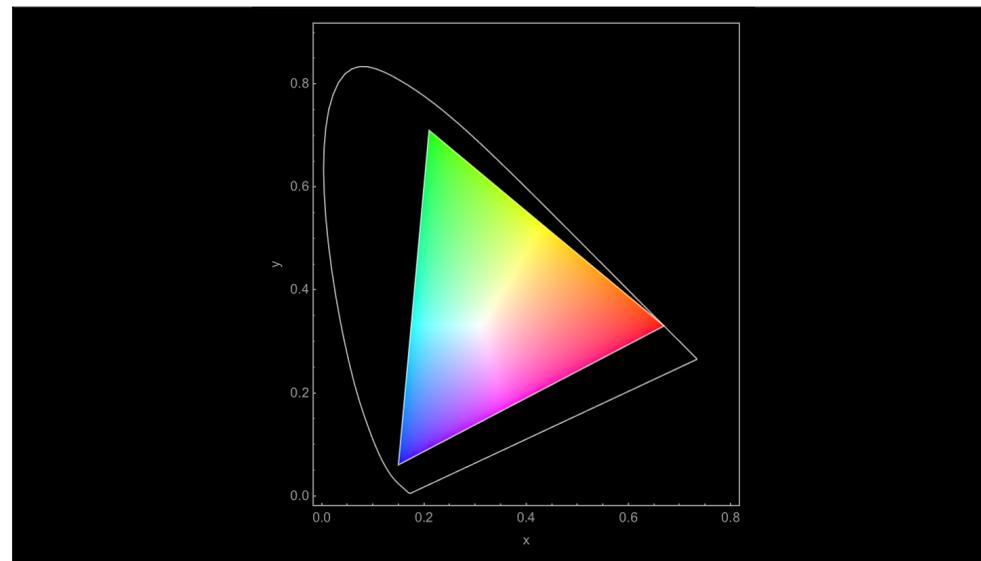
79



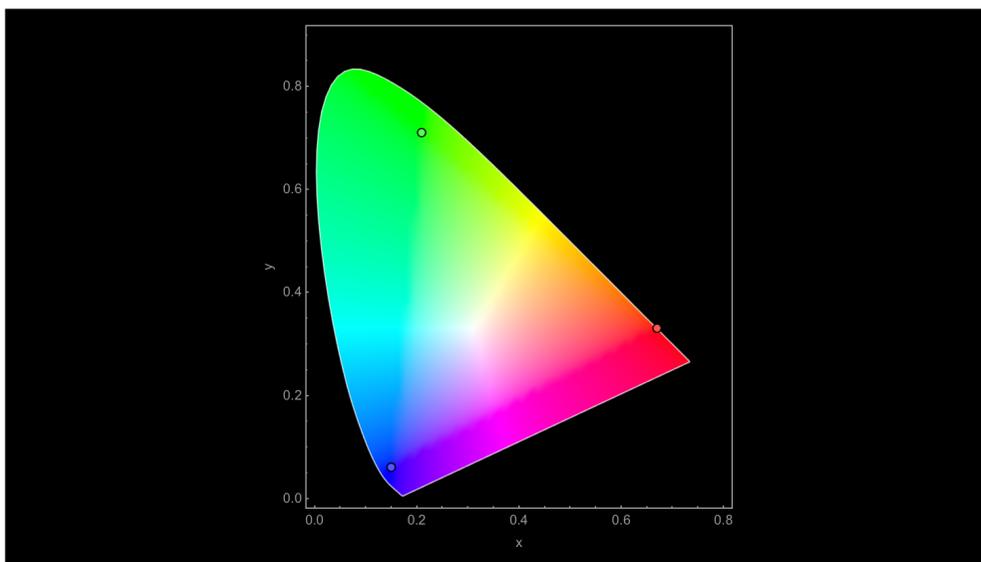
80



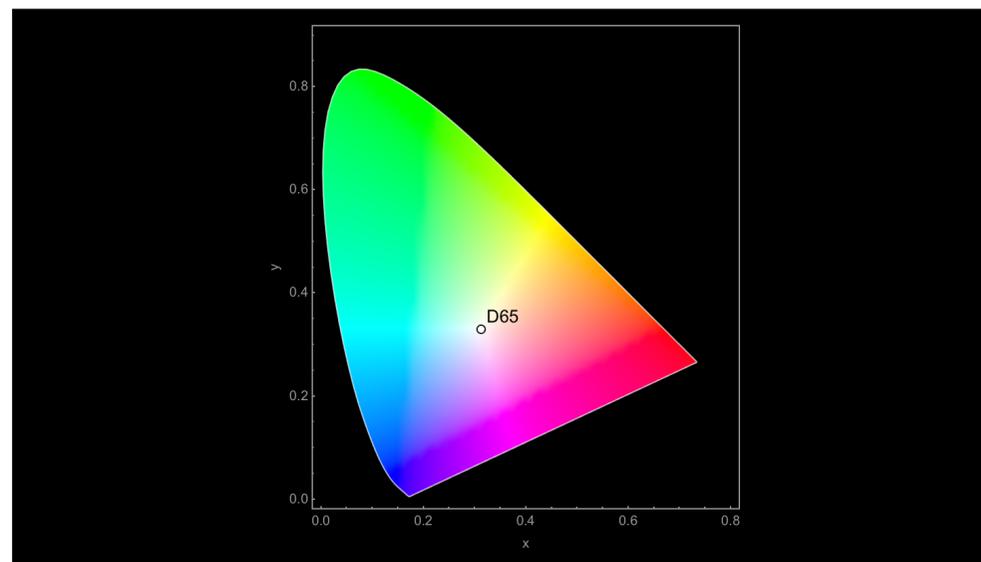
81



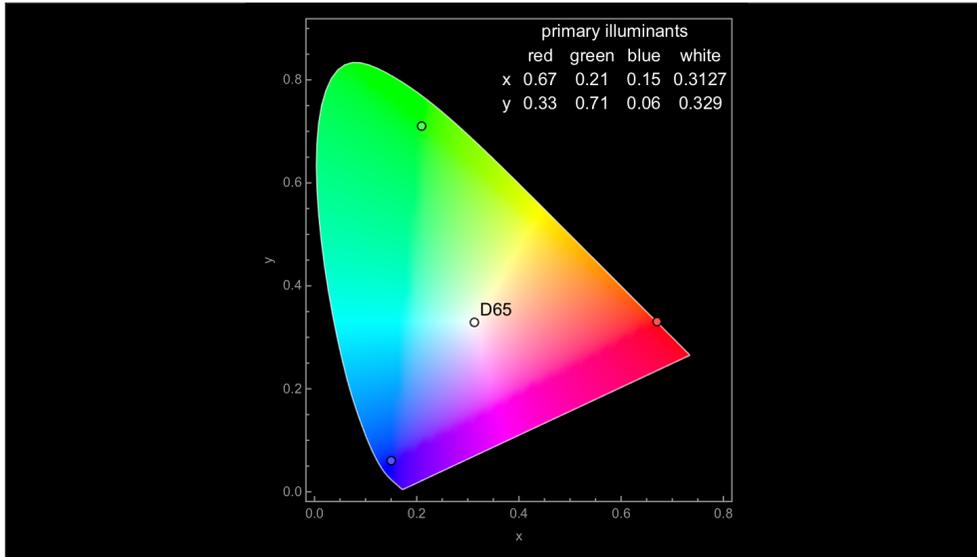
82



83



84



85

## Kompression

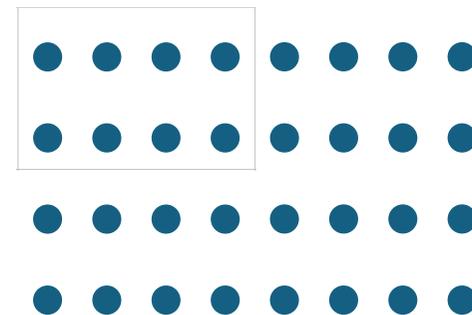
- nicht komprimiert
- verlustfrei komprimiert
- verlustbehaftet komprimiert
- komprimiert generiert
- **Farbunterabtastung**

86

## Farbunterabtastung

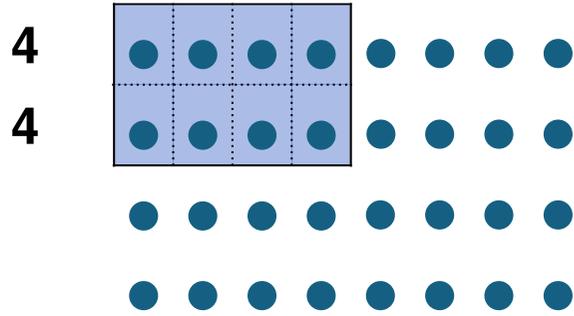
- 4:4:4
- 4:2:2
- 4:2:0 / 4:1:1

87



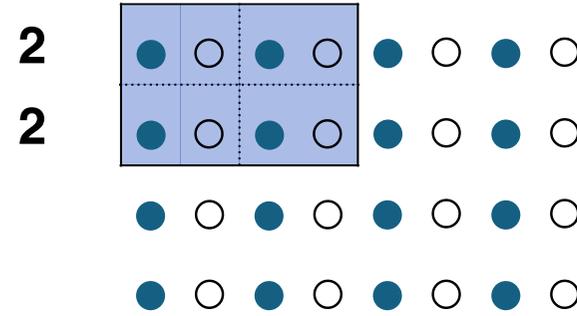
88

**4:4:4**



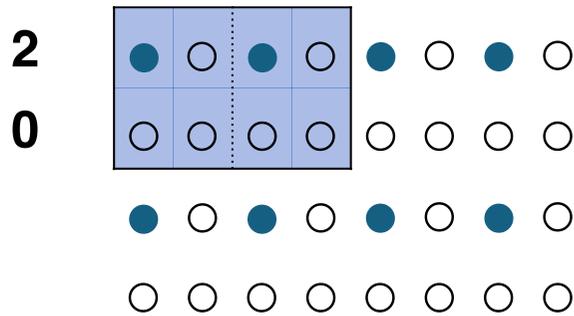
89

**4:2:2**



90

**4:2:0**



91

## Farbunterabtastung

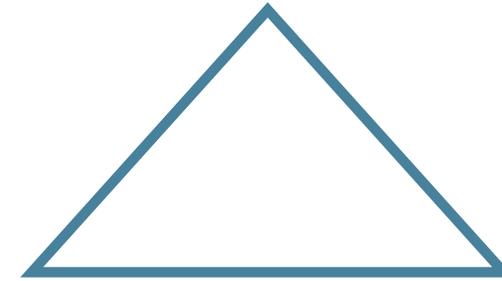
- 4:4:4
- 4:2:2
- 4:2:0 / 4:1:1

92



93

Bildqualität



Codierungszeit

Dateigrösse

94

## Audio Container und Audio Codecs

### Audio-Container

- WAVE
- AIFF
- BWF
- XMF

### Audio-Codec

#### Master:

- PCM
- FLAC

#### Zugang:

- AAC
- MP3

95

## Audiodaten

- pcm\_s16le
- pcm\_s24le
- pcm\_s32le

96

## Video und Bild Container

### Video-Container

- MP4
- MOV
- AVI
- MXF
- Matroska (.mkv)
- Flash

### Einzelbild-Container

- Folder
- TAR
- ZIP
- MXF
- Matroska (.mkv)
  
- Cinema DNG
- Motion JPEG

97

## Video und Bild Codecs

### Video-Codec

#### Master:

- 8 bit uncompressed
- 10 bit uncompressed
- ProRes RAW
- CineForm RAW
- HuffYUV
- FFV1

### Video-Codec

#### Mezzanine:

- ProRes 422,  
ProRes 4444,
- DNxHD, DNxHR

#### Zugang:

- H.264 (AVC),  
H.265 (HEVC),  
H.266 (VVC),  
AV1

### Bild-Codec

- TIFF
- DPX
- JPEG 2000
- OpenEXR
- DNG

98

## Videodaten

- rgb48le
- rgb24
- rgb72le
  
- bayer\_bggr16le
- bayer\_bggr24le
  
- yuv444p16le
- yuv422p10le
- uyvy422
- yuv420p
- yuv444p24le

99

## Praktische Anwendung

Einführung in FFmpeg  
Bern, 3. Februar 2025

100

## FFmpeg-Befehlsstruktur

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

### FFmpeg-Syntax der Argumente:

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

101

## FFmpeg-Syntax

```
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
```

102

## Metadaten extrahieren

```
ffprobe mandela.avi
```

- Container

```
→ ffprobe -show_format mandela.avi
```

- Codec

```
→ ffprobe -show_streams mandela.avi
```

- Metadaten formatieren

```
→ ffprobe -show_format -show_streams -print_format json  
mandela.avi
```

103

## Metadaten speichern

```
ffprobe  
-show_format  
-show_streams  
-print_format json  
mandela.avi  
> mandela.txt
```

104

## Dateiumwandlungen

- Container ändern und vergleichen
- von einer Masterdatei eine Mezzanine-Datei erzeugen
- von einer Masterdatei eine Zugangsdatei erzeugen
- von einer Mezzanine-Datei eine Zugangsdatei erzeugen

105

## Container ändern

```
ffmpeg  
-i mandelbrot.avi  
-c copy  
mandelbrot.mov
```

106

## Berechnen von MD5-Hashwerten

```
ffmpeg  
-i mandelbrot.mov  
-f framemd5  
mandelbrot_mov_framesmd5.txt
```

107

## Vergleichen von MD5-Hashwerten

### Mac/Linux

```
diff  
-s mandelbrot.avi_framesmd5.txt  
mandelbrot_mov_framesmd5.txt
```

### Windows

```
fc  
mandelbrot.avi_framesmd5.txt  
mandelbrot_mov_framesmd5.txt
```

108

## Parameter testen

- In Gruppen einen Command mit verschiedenen Parameter testen
  - Master (TIFF-Einzelbilder) > Mezzanine (ProRes/MKV)
  - Master (TIFF-Einzelbilder) > Zugang (H.264/MP4)
  - Master (uncompressed 10 bit) > Zugang (H.264/MP4)
- Wenn Zeit - weitere Commands suchen und testen
  - Dateieigenschaften ändern
  - Qualitätskontrolle: Splitscreen zweier Dateien/ Differenzdatei zweier Dateien

109

## Master > Mezzanine

Erzeugen einer ProRes-Datei aus TIFF-Einzelbilder und sie ins HD-Format korrekt einbetten.

Parameter:

- f es werden Einzelbilder verarbeitet
- i die Ausgangsdatei
- c:v der Video-Codec "Apple ProRes 422" wird ausgewählt
- profile:v die Variante HQ (High Quality) von ProRes ist 3
- filter:v die Grösse anpassen und ins HD Reinstellen
- an kein Ton ("audio no")  
der Name der zu erzeugenden Datei [ohne -o]

110

## Master > Mezzanine

Erzeugen einer ProRes-Datei aus TIFF-Einzelbilder und sie ins HD-Format korrekt einbetten.

```
ffmpeg
-f image2
-framerate 24
-i DUFAY_TIFF/Dufay_%06d.tif
-filter:v "scale=1440:1080:flags=lanczos,
pad=1920:1080:240:0"
-c:v prores_ks -profile:v 3 -an
Dufay_ProRes.mkv
```

111

## Master > Zugang

Erzeugen einer H.264-Videodatei aus TIFF-Einzelbilder und sie ins HD-Format korrekt einbetten.

```
ffmpeg
-f image2 -framerate 24
-i DUFAY_TIFF/Dufay_%06d.tif
-filter:v
"scale=1440:1080:flags=lanczos,pad=1920:1080:240:0"
-pix_fmt yuv420p
-c:v libx264 -preset veryslow -crf 30
Dufay_1_H264.mp4
```

112

## Mezzanine > Zugang

Erzeugen einer H.264 Videodatei aus einer ProRes/MKV-Videodatei.

```
ffmpeg
-i Dufay_ProRes.mkv
-pix_fmt yuv420p
-c:v libx264 -preset veryslow -crf 30
Dufay_2_H264.mp4
```

113

## Master Einzelbild > Master Video

Erzeugen einer FFV1-Datei aus TIFF-Einzelbildern. ([https://avpres.net/FFmpeg/sq\\_FFV1.html](https://avpres.net/FFmpeg/sq_FFV1.html))

```
ffmpeg
-f image2
-framerate frames_per_second
-i input_file_regex.extension
-c:v ffv1
-level 3
-threads nb_threads
-coder 1
-context 1
-g 1
-slices 24
-slicecrc 1
-c:a copy
output_file

ffmpeg
-f image2
-framerate 24
-i DUFAY_TIFF/Dufay_%06d.tif
-c:v ffv1
-level 3
-threads 8
-coder 1
-context 1
-g 1
-slices 24
-slicecrc 1
-an
Dufay_ffv1.mkv
```

114

## Master Video > Zugang

Erzeugen einer H.264/MP4-Videodatei aus einer FFV1/MKV-Videodatei.

[https://avpres.net/FFmpeg/im\\_H264.html](https://avpres.net/FFmpeg/im_H264.html)

[https://amiaopensource.github.io/ffmpegprovisr/#transcode\\_h264](https://amiaopensource.github.io/ffmpegprovisr/#transcode_h264)

```
ffmpeg
-i input_file
-c:v libx264
-preset preset_value
-crf constant_rate_factor
-pix_fmt yuv420p
-c:a aac
output_file

ffmpeg
-i Dufay_ffv1.mkv
-c:v libx264
-preset veryslow
-crf 18
-pix_fmt yuv420p
-c:a aac
Dufay_ffv1_h264.mp4
```

115

## Master Video Digitalisat > Zugang

Erzeugen einer H.264/MP4 Videodatei aus einer uncompresssed 10bit/MOV-Videodatei.

[https://avpres.net/FFmpeg/im\\_H264.html](https://avpres.net/FFmpeg/im_H264.html)

[https://amiaopensource.github.io/ffmpegprovisr/#transcode\\_h264](https://amiaopensource.github.io/ffmpegprovisr/#transcode_h264)

```
ffmpeg
-i input_file
-c:v libx264
-preset preset_value
-crf constant_rate_factor
-pix_fmt yuv420p
-c:a aac
output_file

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

116

## Ressourcen

FFmpeg Cookbook for Archivists  
→ [avpres.net/FFmpeg/](https://avpres.net/FFmpeg/)

ffmprovizr  
→ [amiaopensource.github.io/ffmprovizr/](https://amiaopensource.github.io/ffmprovizr/)

Liste von möglichen Parameter anzeigen

```
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
```

117

## Weitere Möglichkeiten mit FFmpeg

- Quantisierungsauflösung verändern
- Scanmodus ändern (interlaced > progressive)
- Datei zuschneiden
- Etc.
  
- Qualitätskontrolle: Differenzdatei zweier Dateien
- Qualitätskontrolle: Splitscreen zweier Dateien

118

## Beobachtungen

- Was haben Sie getestet?
- Was haben Sie dabei beobachtet, gelernt?
- Inwiefern verändern sich die Dateien (Metadaten und optisch)?

119

## Danke für Ihre Aufmerksamkeit!

- Feedback
  
- Aufbau Kurs
  - Wann: 13. März 2025
  - Inhalt: Anwendung von FFmpeg an konkreten Fallbeispiele & Möglichkeiten der Qualitätskontrolle
  - Welche Themen interessieren Sie für diesen Kurs?

120

## **AV Preservation by reto.ch Ødegård & Bernard Restaurierung**

**Reto Kromer:**

[reto.ch](http://reto.ch) / [info@reto.ch](mailto:info@reto.ch)

**Eléonore Bernard und Kamilla Ødegård:**

<https://atelier40a.ch/odegard-bernard-restaurierung-klg/>

[eleonore.bernard@atelier40a.ch](mailto:eleonore.bernard@atelier40a.ch) /

[kamilla.oedegard@atelier40a.ch](mailto:kamilla.oedegard@atelier40a.ch)