

## Anleitung zu *FFmpeg*

Dieses Dokument beschreibt die Verwendung von *FFmpeg* im Kontext der Lehrveranstaltung *Medientechnologie LB*.

### Grundlegende Kommandozeilensyntax

*FFmpeg* verlangt bei der Ein- und Ausgabe von Dateien zumindest einen Ein- und einen Ausgabedateipfad, wobei ersterer durch `-i` eingeleitet wird, z.B.:

```
ffmpeg -i input.wav output.mp3
```

Dateiendungen werden von *FFmpeg* die dazugehörigen Audio-, Video- bzw. Containerformate zugeordnet, sofern nichts Abweichendes spezifiziert ist. Im obigen Beispiel wird eine Audiodatei im WAVE-Format in eine Datei im in MPEG-1 Part 3 Layer III definierten Format umgewandelt.

Das Ausgabeformat kann mittels eines Formatparameters explizit angegeben werden. Um beispielsweise eine Datei im AAC-Format auszugeben, kann der Audio-Codec mittels des `acodec`-Parameters eingestellt werden:

```
ffmpeg -i input.wav -acodec aac output.aac
```

Es gilt zu beachten, dass sämtliche **Ausgabeparameter vor** dem jeweiligen Ausgabedateinamen, aber nach dem vorangegangenen Ein- oder Ausgabedateipfad zu stehen haben. Folgende grundlegende Parameter stehen zur Verfügung:

Kommandozeilenparameter	Bedeutung
<code>-acodec &lt;Format&gt;</code>	Audio-Codec
<code>-vcodec &lt;Format&gt;</code>	Video-Codec
<code>-f &lt;Format&gt;</code>	Containerformat

Eine Liste der unterstützten Containerformate kann wie folgt abgerufen werden:

```
ffmpeg -formats
```

Die verfügbaren Audio- und Video-Codecs können ebenfalls abgerufen werden:

```
ffmpeg -codecs
```

Der Pseudo-Codec *copy* sorgt dabei für ein Kopieren der Audio- bzw. Videospur, d.h. er verhindert, dass diese neu kodiert wird – sie wird unverändert aus der Eingabedatei übernommen.

Es ist möglich, mehrere Ein- und Ausgabedateien zu spezifizieren, z.B.:

```
ffmpeg -i video.264 -i audio.wav output.mp4 -vcodec copy copy.mp4
```

In diesem Beispiel werden ein Bitstrom nach H.264 Annex B als Videospur und eine WAVE-Datei als Audiospur eingelesen. Aus den beiden Spuren werden zwei Ausgabedateien im MP4-Format erzeugt. Für die erste werden zuerst die Videospur in H.264 und die Audiospur in AAC umgewandelt (Standard-Codecs für das MP4-Format). Wird für eine Spur vor der Ausgabe kein Codec angegeben, wird der Standard verwendet. Für die zweite Ausgabedatei wird die Eingabevideospur unverändert übernommen und die Audiospur in AAC umgewandelt.

## Datenratensteuerung

Die Datenrate der Ausgabedateien kann mit dem Parameter `-b` (der, wie oben beschrieben, **vor** dem Ausgabedateipfad zu stehen hat) geregelt werden. Analog zu den Formatparametern stehen folgende Datenratenparameter zur Verfügung:

Kommandozeilenparameter	Bedeutung
<code>-b:a &lt;Datenrate&gt;</code>	Datenrate des Audio-Codecs
<code>-b:v &lt;Datenrate&gt;</code>	Datenrate des Video-Codecs

Die Datenrate wird dabei in bit/s angegeben. Die praktisch relevanten SI-Präfixe werden unterstützt, z.B. entspricht der Datenratenparameter `-b:a 32k` einer Audiodatenrate von 32 kbit/s bzw. entspricht `-b:v 2000k` einer Videodatenrate von 2.000 kbit/s.

## Netzwerkein- und -ausgabe

Statt Dateipfaden können zur Ein- und Ausgabe alternativ auch Netzwerkadressen und -protokolle angegeben werden, z.B.:

```
ffmpeg -i file.mp4 -codec copy -f rtp_mpegts rtp://127.0.0.1:1024
```

In diesem Beispiel wird eine aus einer H.264-Videospur und einer AAC-Audiospur bestehende Datei im MP4-Format unverändert im MPEG-TS-Format über RTP an Port 1024 localhost (IPv4) gestreamt.

## Praktische Parameter

Auf folgende Parameter von besonderer praktischer Relevanz sei hingewiesen:

Kommandozeilenparameter	Bedeutung
<code>-an</code>	Audiospur wird entfernt (pro Ausgabe)
<code>-vn</code>	Videospur wird entfernt (pro Ausgabe)
<code>-y</code>	Ausgabedateien werden überschrieben
<code>-re</code>	Ausgaben erfolgen möglichst in Echtzeit
<code>-stream_loop -1</code>	Eingabedateien werden wiederholt/geloopt

Die unteren drei Parameter gelten global und sind vor dem ersten `-i` zu setzen.

## Weitere Informationen

Hilfe zur Bedienung von *FFmpeg* sowie eine Erläuterung der Kommandozeilenoptionen sind über

```
ffmpeg -h long
```

verfügbar. Desweiteren geben die man-Pages unter Linux detailliert Auskunft. Darüber hinaus sind unter <http://ffmpeg.org/ffmpeg.html> die meisten Parameter von *FFmpeg* mit Beispielen erläutert.