

## Aufgaben zu Transkodierung

Lösen Sie die nachfolgenden Aufgaben und bereiten Sie diese bis zum nächsten Lehrveranstaltungstermin vor. Unterstrichene Aufgaben sind nach Möglichkeit während der Lehrveranstaltung zu lösen.

### LB-TK 01.

- a) Laden Sie drei beliebige Musikstücke im „MP3“-Format von [http://freemusicarchive.org/music/Jahzzar/Travellers\\_Guide/](http://freemusicarchive.org/music/Jahzzar/Travellers_Guide/) herunter. Kodieren Sie diese mit *avconv* derart ins AAC-Format, dass die Qualität Ihrer Meinung nach gerade noch ausreichend ist, d.h., dass jede geringere Bitrate die kodierte Datei auf Grund der Verzerrungen quasi „unanhörbar“ macht. Verwenden Sie dabei eine konstante Bitrate und gehen Sie auf die Unterschiede zwischen den gewählten Musikstücken, den Bitraten sowie deren Zusammenhang ein.
- b) Kodieren Sie die im Rahmen von a) erstellten Dateien i) im AAC-Format mit einer Bitrate von 256 kbps, ii) im „MP3“-Format mit ihrer jeweiligen durchschnittlichen Originalbitrate. Vergleichen Sie die tatsächlich benötigte (von *avconv* ausgegebene) Bitrate mit der angegebenen und erläutern Sie die Unterschiede anhand der Qualität der kodierten Dateien.

### LB-TK 02.

- a) Laden Sie die MPEG-4-Part-2-kodierte Sequenz *AMOS TV* von <http://www.divx.com/en/devices/profiles/video> herunter. Untersuchen Sie das Video auf die bereits für JPEG in der Vorlesung behandelten Kompressionsartefakte, die für JPEG, MPEG-4 Part 2 und H.264 (und viele andere) typisch sind und geben Sie Bildbeispiele an.
- b) Kodieren Sie die Datei aus a) mittels *avconv* als H.264-Video mit AAC-Audio. Experimentieren Sie mit verschiedenen durchschnittlichen Bitraten und ermitteln Sie jene, bei denen Ihrer Meinung nach kein Unterschied mehr zum Ausgangsmaterial feststellbar ist, d.h., die Qualität quasi so gut wie jene des Ausgangsmaterials ist. Begründen Sie Ihre Wahl unter Berücksichtigung der Tatsache, dass MPEG-4 Part 2 im Allgemeinen bei gleicher Bitrate auf Grund seiner weniger weit fortgeschrittenen Kodierungswerkzeuge weniger effizient kodiert als H.264.  
Hinweise: Verwenden Sie der Kompatibilität halber die Dateiendung *mp4* für die Ausgabedatei, um *avconv* die Verwendung des gleichnamigen Containerformates zu signalisieren, das die Abspielbarkeit mit den meisten Playern sicherstellt. Kodieren Sie außerdem jeweils nur einige Dutzend Frames aus der Mitte des Ausgangsvideos, um die Qualität beurteilen zu können. Geben Sie dazu die Anzahl der zu kodierenden Frames sowie einen Offset an, ab dem kodiert werden soll (vgl. Hilfe zu *avconv*).

- c) Kodieren Sie im Rahmen von b) erstellte Datei ein weiteres Mal mit denselben Einstellungen. Begründen Sie, ob die Qualität dabei konstant bleibt oder sich verändert. Dokumentieren Sie, falls sie sich verändert, wie stark die Änderung ist und zeigen Sie beispielhaft Änderungsstellen auf.

### LB-TK 03.

- a) Richten Sie *Libav* (Quellcode unter <http://git.libav.org/?p=libav.git;a=snapshot;h=HEAD;sf=tgz>) unter Berücksichtigung der unten angeführten Ergänzungen ein und erläutern Sie ebendiese Einrichtung.
- b) Erstellen Sie ein minimales Beispielprogramm, das die Funktion `av_register_all()` der *Libav* aufruft.
- c) Erweitern Sie das Programm aus b) mit Hilfe der Dokumentation (<http://www.libav.org/DOXYGEN/master/index.html>) und der *Libav*-Beispielprogramme in `doc/examples` (im *Libav*-Repository) derart, dass es einen der Transkodierungsvorgänge aus LB-TK 01. a) nachstellt. Die Pfade der Eingabe- bzw. Ausgabedatei sollen dabei in ebendieser Reihenfolge als Kommandozeilenparameter angegeben werden können.

### Ergänzung zu LB-TK 03.: Zur Einrichtung von *Libav*

Kompilieren und installieren Sie *Libav* mittels `configure` und `make`. Aktivieren Sie dabei beim Aufruf von `configure` nur jene Encoder und Decoder, die Sie später tatsächlich verwenden bzw. benötigen, um die Kompilierungszeit möglichst kurz zu halten, z.B. (ohne Zeilenumbrüche!):

```
./configure --disable-everything --enable-demuxer=mp3
--enable-decoder=mp3 --enable-encoder=aac --enable-muxer=mp4
--enable-protocol=file --disable-programs
```