

BiaMovE documentation

Harishankar V

BiaMovE documentation

by Harishankar V

A short guide to video encoding using BiaMovE (via Mencoder)

Table of Contents

1. Introduction.....	1
1.1. Why Mencoder.....	1
1.2. Philosophy.....	1
2. Video encoding basics.....	2
2.1. The problems.....	2
2.2. Concepts.....	2
2.2.1. Bitrate.....	2
2.2.2. Framerate.....	2

Chapter 1. Introduction

BiaMovE (pronounced "by-a-movie") is a graphical user interface to Mencoder (<http://www.mplayerhq.hu/>), an extremely powerful and versatile cross-platform video encoding command-line utility. The problem with Mencoder is that it is very complex and non-trivial to lay users.

Adding to the complexity is the existence/non-existence of video/audio codecs, proprietary technologies and technical terminology related to audio and video. I had a very difficult time poring over the Mencoder documentation to find specific options so I decided that while I was learning how to use it, why not create a GUI front-end for it?

1.1. Why Mencoder

Why Mencoder and why not FFMpeg? Well, for one, Mencoder can wrap around FFMpeg libraries using `libavcodec`. Secondly I found using the FFMpeg command line much easier than Mencoder and felt that it would be better to write a GUI around Mencoder instead.

1.2. Philosophy

When I wrote BiaMovE I wanted to mainly explore the codec-specific options of Mencoder. I did so because video/audio codecs are extremely complex creatures. I cannot pretend that I understood everything I read in the Mencoder manual pages. With BiaMovE I've tried to hide a little bit of the complexity, but without restricting the advanced user.

BiaMovE is not a tool for the newbie though. It is just a thin (albeit useful) GUI wrapper to Mencoder. It does *not*:

- Check for the sanity of your Mencoder configuration, nor does it check for codec availability on your setup.
- Does not verify the sanity of your options. It just passes on your choices to the underlying Mencoder command line.

Along the way, I've been bumbling along, discovering by trial and error, by experimentation and research, the proper way to mix video and audio codecs and produce valid video streams. If you are a newbie to video encoding and want to learn more, I suggest your read the section on video encoding basics and also study the Mencoder manual pages in depth.

Chapter 2. Video encoding basics

Before you jump into video encoding, it's useful to understand a few video (and audio) related concepts. This will help you exploit the full power of BiaMovE and Mencoder. While most of the information here is common knowledge, I've documented some of my thoughts and discoveries in an easily accessible manner. However, before you dive in, I suggest you read the concepts section for definitions to common terms specific to digital audio/video technologies.

If you are of the "just want to rip my DVDs to my iPod" type, feel free to skip this section.

2.1. The problems

For most people simply using a command-line tool like Mencoder can be a daunting task. The Mencoder manual page alone is scary enough, because it documents both the player MPlayer as well as the encoder.

Apart from the command-line itself, video and audio codecs are complex and tricky subjects in themselves. Apart from the variety of standards and formats adopted by different organizations (for instance, MPEG-4, H.264, DivX, XViD etc.), there exists hundreds of specifications within even these formats. Worse still, many of these standards are not compatible with each other and many video players, particularly hardware players, do not recognize open standards codecs (the Apple iPod™ is a case in point).

Complicating it is the fact that not all muxers (container formats) work with all codecs and not all video and audio codecs are compatible with each other. Again, certain codecs provide very sane default parameters while other codecs require to be told exactly how to encode the underlying data. Last, but not least, some codecs are plain buggy and broken or don't work very well with Mencoder.

BiaMovE cannot overcome these problems for you for the reasons already mentioned earlier. However reading these notes should help you avoid some of the pitfalls.

2.2. Concepts

2.2.1. Bitrate

Bitrate is the number of bits which encode the information per unit of time. It is usually measured in kilobits per second (kbit/s) or megabits per second (mbit/s). In practical terms, higher the bitrate, higher the quality of the information stored (whether video or audio).

The best bitrate depends on several factors including desired video/audio quality versus size trade-off, the resolution of the video as well as the number of frames per second.

2.2.2. Framerate

Framerate is the number of frames per unit time. Frames may be audio or video frames. Think of a single frame as a still. Over the years, different video standards (PAL, NTSC etc.) have evolved and use

standard framerates (24.997 fps, 25 fps, 29.997 fps, 30 fps etc.) but theoretically any video can have any framerate or even variable framerates.

The higher the framerate, smoother the animation. Usually 25 to 30 fps is more than enough for the human eye.