VID2MP4 Guide

Author: Kim Miikki
Date: 15.9.2021

1 Introduction

This program converts a video file (e.g., h264 format video file) to mp4 format. *Vid2mp4* is a front-end application for *MP4Box*. When the record speed in FPS is known and there is a PTS (Presentation Time Stamp) file present, the correct FPS value for the mp4 can be calculated. This is done in *vid2mp4.py* for accurate playback speed.

2 System Requirements

Operating System: ALL

Program: MP4Box (RPI/Ubuntu installation: sudo apt install -y gpac)

Python 3

3 Program Usage

Vid2mp4 has some arguments which are used for FPS adjustments and disabling generation of analysis plot files:

```
$ vid2mp4.py -h
usage: vid2mp4.py [-h] [-rec REC] [-fps FPS] [-n] file

positional arguments:
  file        video file name

optional arguments:
  -h, --help  show this help message and exit
  -rec REC    recorded FPS as float
  -fps FPS    playback FPS as float
  -n          do not create graphs
```

Only file is a mandatory argument for this program. Playback FPS is as default 25, but it can be overridden with a -fps argument. Correct video speed (slowdown, normal or speedup) can be controlled with the -rec argument and a PTS file. Generation of PTS graphs can be disabled with -n switch.

4 Use Case

A short video was recorded with *fpsvideo.py* to demonstrate how to use this program. The video recorder created these files:

```
v-1440x768_100.0fps_10000ss_120s.pts
v-1440x768_100.0fps_10000ss_120s.h264
v-1440x768_100.0fps_10000ss_120s.rec
v-1440x768_100.0fps_10000ss_120s.mkv
v-1440x768_100.0fps_10000ss_120s.log
```

Originally the vide was captured in the h264 file, alongside with a pts file. These two files were required when converting to video to mp4 format with following command:

```
$ vid2mp4.py v-1440x768_100.0fps_10000ss_120s.h264 -rec 100
```

The stem for the pts file must be same as the stem for the video file. Otherwise the pts file is not read. Following files are created:

```
v-1440x768_100.0fps_10000ss_120s-histogram.png
v-1440x768_100.0fps_10000ss_120s-distribution.png
v-1440x768_100.0fps_10000ss_120s-intervals.txt
v-1440x768_100.0fps_10000ss_120s.mp4
v-1440x768_100.0fps_10000ss_120s-mp4.log
```

Interval distribution plots are generated and saved to two png files. The results are shown in Figure 1.

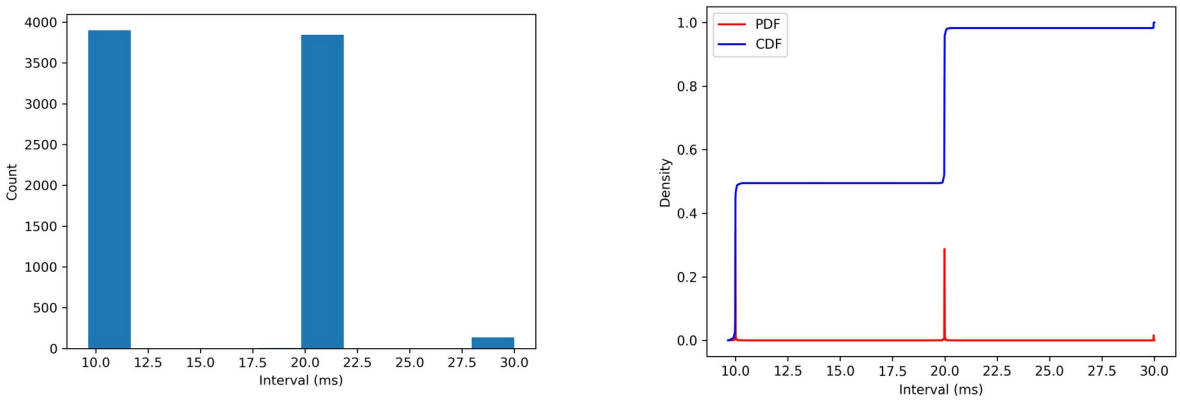

Figure 1. PTS file analyzed with graphs: interval distribution on left side, and probability and cumulative distribution function of the intervals on the right side.

The interval list is stored in a text file, where the stem is ended with “-intervals” string. A summary of selections and a statistical analysis can ar stored in the log file:

```
$ cat v-1440x768_100.0fps_10000ss_120s-mp4.log
vid2mp4.py log file

Log created on 2021.09.15-14:08:20

Program arguments:

FPS recorded: 100
FPS playback: 25

FPS: 16.434
Slowdown: 4.0x

Statistics:
Intervals: 7887
Frames   : 7888
Mean     : 15.21
Median   : 19.91
Variance : 28.37

MP4 command:
MP4Box -add v-1440x768_100.0fps_10000ss_120s.h264 -new v-
1440x768_100.0fps_10000ss_120s.mp4 -fps 16.434
```

A MP4Box command is generated and executed in order to convert the original video file to a mp4 format file. This command is also stored in the log file, as shown above.

Individual frames can be extracted with *vid2pic.py*, as shown in Figure 2.

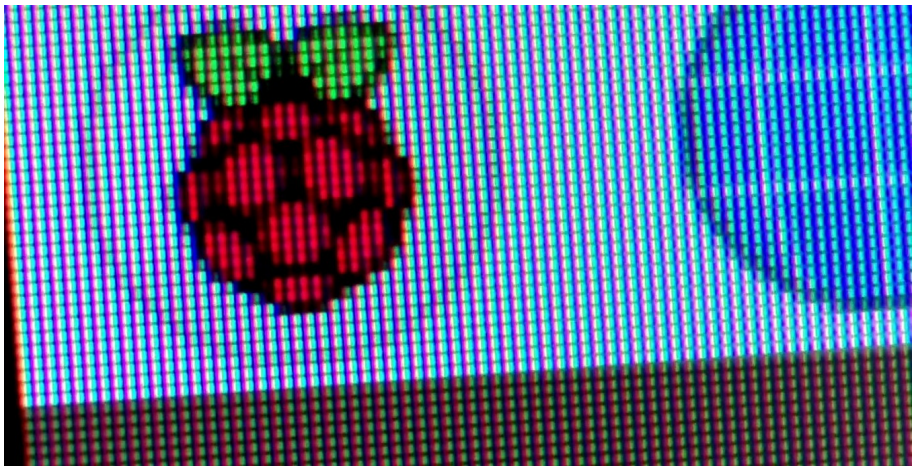

Figure 2. Frame 7830 extracted with vid2pic.py.

The converted video file properties can be extracted and listed with *mediainfo*:

```
$ mediainfo v-1440x768_100.0fps_10000ss_120s.mp4
General
Complete name                            : v-1440x768_100.0fps_10000ss_120s.mp4
Format                                   : MPEG-4
Format profile                           : Base Media
Codec ID                                 : isom (isom/avc1)
File size                                : 153 MiB
Duration                                 : 7 min 59 s
Overall bit rate                         : 2 672 kb/s
Encoded date                             : UTC 2021-09-15 11:08:11
Tagged date                              : UTC 2021-09-15 11:08:11

Video
ID                                       : 1
Format                                   : AVC
Format/Info                              : Advanced Video Codec
Format profile                           : High@L4.2
Format settings                          : CABAC / 1 Ref Frames
Format settings, CABAC                   : Yes
Format settings, ReFrames                : 1 frame
Format settings, GOP                     : M=1, N=60
Codec ID                                 : avc1
Codec ID/Info                            : Advanced Video Coding
Duration                                 : 7 min 59 s
Bit rate                                 : 2 672 kb/s
Maximum bit rate                         : 6 245 kb/s
Width                                    : 1 440 pixels
Height                                   : 768 pixels
Display aspect ratio                     : 1.85:1
Frame rate mode                          : Constant
Frame rate                               : 16.434 FPS
Color space                              : YUV
Chroma subsampling                       : 4:2:0
Bit depth                                : 8 bits
Scan type                                : Progressive
Bits/(Pixel*Frame)                       : 0.147
Stream size                              : 153 MiB (100%)
Title                                    : 0fps_10000ss_120s.h264@GPAC0.5.2-DEV-
revVersion: 0.5.2-426-gc5ad4e4+dfsg5-5
Encoded date                             : UTC 2021-09-15 11:08:11
Tagged date                              : UTC 2021-09-15 11:08:17
Codec configuration box                  : avcC
```