Screen capture tools to record online tutorials

This document is made to explain how to use ffmpeg and QuickTime to record mini tutorials on your own computer. FFmpeg is a cross-platform tool available for Windows, Linux and Mac. Installation and use process depends on your operating system. This info is taken from (Bellard 2016). Quicktime Player is natively installed on most of Mac computers. This tutorial focuses on Linux and Mac.

Table of content

1. Introduction .................................................................................................................................1
2. Linux ................................................................................................................................................1
   2.1. FFmpeg ......................................................................................................................................1
       2.1.1. installation for Linux ........................................................................................................1
           2.1.1.1. Add necessary components ......................................................................................1
           2.1.2. Screen recording with FFmpeg .....................................................................................2
           2.1.2.1. List devices to know which one to record .................................................................2
           2.1.2.2. Record screen and audio from your computer ..........................................................3
   2.2. Kazam ........................................................................................................................................4
       2.2.1. Installation for Linux Ubuntu ............................................................................................4
       2.2.2. Screen recording with Kazam ..........................................................................................4
3. Mac OS X ........................................................................................................................................5
   3.1. Requirements ............................................................................................................................5
       3.1.1. Homebrew ..........................................................................................................................5
   3.2. FFmpeg Installation ..................................................................................................................6
   3.3. Screen recording ........................................................................................................................6
       3.3.1. With Quicktime Player (Apple Inc. 2007) ........................................................................6
       3.3.2. With ffmpeg ....................................................................................................................6
           3.3.2.1. List devices to know which one to record .................................................................7
           3.3.2.2. Record screen and audio from your computer ..........................................................7
4. Windows ........................................................................................................................................8
   4.1. ActivePresenter for Windows ................................................................................................8
       4.1.1. Installation ..........................................................................................................................8
       4.1.2. Screen recording with ActivePresenter ............................................................................8
5. Shotcut: a free, open-source and cross-platform video editor and converter (Meltytech LLC 2016) .................................................................................................................................10
6. Conversion and format issues ........................................................................................................13
   6.1. Format issues ...........................................................................................................................13
       6.1.1. Using the HTML <video> element ..................................................................................13
7. Appendix .......................................................................................................................................13
   7.1. Use dedicated scripts with FFmpeg .........................................................................................13
   7.2. Conversion with FFmpeg .........................................................................................................14
7.3. Editing videos with ffmpeg

7.3.1. Cutting a video

7.3.2. Accelerate a part of the video

7.3.3. Concatenate the multiple parts to have one only video output

8. Bibliography
1. Introduction

FFmpeg is a powerful command line tool that allows you to record your computer screen and your voice. You can also easily convert videos to several formats. The tool is rich in functionality but these simple configuration instructions allows you to start easily. We will see here how to install and use this tool on Unix based systems (Mac OS X and Linux).

2. Linux

2.1. FFmpeg

2.1.1. installation for Linux

*These installation instructions have been made on Linux Ubuntu Trusty 14.04 LTS*

To install FFmpeg, type the three commands below in your terminal:

- add the PPA (Personal Package Archive) ppa:mc3man/trusty-media to your software sources with:

  ```sh
  sudo add-apt-repository ppa:mc3man/trusty-media
  ```

- then type `sudo apt-get update`

- and finally `sudo apt-get install ffmpeg`

Type `ffmpeg` in your terminal. If you see a text beginning with `ffmpeg version` that means FFmpeg is now installed on your computer.

*On Ubuntu, Libav (Libav 2016) is normally the native product but FFmpeg works well.*

2.1.1.1. Add necessary components

Now add the 3 packages which will be important for recording in good quality and for conversion purposes later. For example to convert a video in a .webm format (webmedia) to stream on the web. So, add:

- `sudo apt-get install libx264-dev` for H264. It allows you to record in a great video quality.
- `sudo apt-get install libx265-dev` for H265 which is another format to record in a very good quality.
- `sudo apt-get install libvpx-dev` which will allows you to convert a video in a .webm format.

Note that an alternative is also to compile FFmpeg from sources (Edgewall Software 2016a).
2.1.2. Screen recording with FFmpeg

Note that some text below is in French because of the settings of the computer used. According to your own settings, the display will be in English, French or other languages.

2.1.2.1. List devices to know which one to record

Type:

\texttt{arecord -l}

for a summary of your devices.

The terminal will normally return something like that:

\texttt{**** Liste des Périphériques Matériels CAPTURE **** carte 0: AudioPCI [Ensoniq AudioPCI], périphérique 0: ES1371/1 [ES1371 DAC2/ADC] Sous-périphériques: 1/1 Sous-périphérique #0: subdevice #0}

or:

\texttt{arecord -L}

for more details.

- default
  - Playback/recording through the PulseAudio sound server
- null
  - Discard all samples (playback) or generate zero samples (capture)
- pulse
  - PulseAudio Sound Server
- sysdefault:CARD=AudioPCI
  - Ensoniq AudioPCI, ES1371 DAC2/ADC
  - Default Audio Device
- front:CARD=AudioPCI,DEV=0
  - Ensoniq AudioPCI, ES1371 DAC2/ADC
  - Front speakers
- surround40:CARD=AudioPCI,DEV=0
  - Ensoniq AudioPCI, ES1371 DAC2/ADC
  - 4.0 Surround output to Front and Rear speakers
- iec958:CARD=AudioPCI,DEV=0
  - Ensoniq AudioPCI, ES1371 DAC2/ADC
  - IEC958 (S/PDIF) Digital Audio Output
2.1.2.2. Record screen and audio from your computer

In your terminal, go to the folder you want to put your video:

```
cd Path/to/my/videos
```

and type the command to record:

```
ffmpeg -video_size 1280x800 -framerate 30 -f x11grab -i :0.0 -f alsa -ac 2 -i hw:0 -c:v libx264 -qp 0 -preset ultrafast out.mp4
```

Press `Enter` to start recording.

Press `q` to stop recording.

Command options:

- `-video_size 1280x800` is your screen resolution. Adapt with yours.
- `-framerate 30` is the number of images/seconds
- `-f x11grab` is the tool to grab your screen on Linux
- `-i :0.0` is to specify which part of the screen you want to record. In this case (0:0) you record all the screen
- `-f alsa` is the tool for audio recording
- `-i hw:0` is to specify (with 0 in this case) which audio device you want to record. Please refer to the number related to your computer. You can find this number with the commands `arecord -l` or `arecord -L` saw above
- `-c:v libx264` is for H264 format for a good quality video
- `-qp 0` & `-preset ultrafast` are for a better quality recording
- `out.mp4` specifies the name and the format of the output you want
Please note that specific codecs are for specific output formats. For example, `libx264` is for .mp4 while `libvpx` is for .webm. If you try to record a video in .mp4 format with `libvpx` codec you will get an error.

2.2. **Kazam**

2.2.1. **Installation for Linux Ubuntu**

“Kazam provides a well designed and easy to use interface for capturing screencasts and screenshots” ([Canonical Ltd 2012](#)). To get it, download it from the Ubuntu apps official directory, or type “`sudo apt-get install kazam`” in a shell. Once it is done, launch Kazam from the applications folder and you should normally have a display as presented below.

![Kazam Interface](image.jpg)

2.2.2. **Screen recording with Kazam**

As one can see, it is possible to select a screencast or a screenshot as well as full screen, window, area or even multiple screens recordings. The Figure below shows the preferences which we can reach by clicking “`File → Preferences`”.

![Kazam Preferences](image.jpg)
It is possible there to choose the numbers of images per second and the format between .mp4, .avi or .webm. Then we click on “Capture” and the process begins. Once it is recording, a little camera icon with a red point appears in the menu bar as illustrated below.

Then we click on that icon to stop the recording. One can also select “pause” and restart from the point where we stopped. While stopping the recording, the software asks to save the file or edit it directly. We choose where to save the file, and the process is over, the video is recorded.

3. Mac OS X

To record your computer screen and your voice on Mac OS X you can use Quicktime Player. It is natively installed on most of Mac Computers. This is why you can find below a quick description about how to record your screen and audio with this software.

ffmpeg needs to install some dependencies on the computer. After several tries on different Mac computers it appears that if the computer is not up-to-date, it can create errors and issues during the installation process. Although, we recommend to try to follow at least the points in the section “FFmpeg installation” below for conversion purposes later.

3.1. Requirements

3.1.1. Homebrew

You will need to have homebrew installed on your Mac. Homebrew is a package manager for Mac (same as apt-get for Linux) which allows you to install several packages not present in the system. Get Homebrew from the site of Howell and Prévost (2016) and read the documentation. In brief, type:

```
/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

in your terminal. Follow the installation process (it could take some time) and normally Homebrew is now installed.

11.08.2016 5 Alexandre Racine
3.2. FFmpeg Installation

Once Homebrew is installed, then, it's just so easy to install FFmpeg in no more than one command. Go to Edgewall Software (2016b) under the FFmpeg through Homebrew section and copy / paste this command:

```
brew install ffmpeg --with-fdk-aac --with-ffplay --with-freetype --with-libass --with-libquvi --with-libvorbis --with-libvxpx --with-opus --with-x265
```

All the arguments like `--with-libvorbis` or `--with-libvxpx` are very important because we will need it later, for conversion purposes for example. With this command, your FFmpeg installation is normally now complete.

3.3. Screen recording

In order to have a neutral environment on your computer. Please first hide the desktop icons with those commands:

```
defaults write com.apple.finder CreateDesktop false
```

and then:

```
killall Finder
```

3.3.1. With Quicktime Player (Apple Inc. 2007)

Go to your applications folder and open Quicktime Player.

In the menu bar, select:

```
file → new screen recording
```

A new window is now open.

- Click on the little arrow in the right up corner and choose your input audio device
- Choose also if you want the mouse clicks to be shown in the video
- Choose the video quality
- Finally click on the recording button

Quicktime Player will ask you if you want to record the entire screen or just a part of it. Follow the instructions related to your choice and start the recording.

Click on the square button to stop recording. The video should now be where you specified to save it.

3.3.2. With ffmpeg

To capture your screen on Mac OS X with FFmpeg, type the following commands in your terminal:
3.3.2.1. List devices to know which one to record

Type:

```bash
ffmpeg -f avfoundation -list_devices true -i ""
```

The terminal will normally return something like that:

```
[AVFoundation input device @ 0x7fae4ac00460] AVFoundation video devices:
[AVFoundation input device @ 0x7fae4ac00460] [0] Caméra FaceTime HD (intégrée)
[AVFoundation input device @ 0x7fae4ac00460] [1] Capture screen 0
[AVFoundation input device @ 0x7fae4ac00460] AVFoundation audio devices:
[AVFoundation input device @ 0x7fae4ac00460] [0] Apowersoft_AudioDevice
[AVFoundation input device @ 0x7fae4ac00460] [1] Built-in Input
```

3.3.2.2. Record screen and audio from your computer

In your terminal, go to the folder you want to put your video:

```bash
cd Path/to/my/videos
```

and type the command to record:

```bash
ffmpeg -f avfoundation -i "1:1" -c:v libx264 -qp 0 -preset ultrafast out.mp4
```

Press `Enter` to start recording.

Press `q` to stop recording.

Now your video should be in folder `Path/to/my/videos`.

Command options:

- `-f avfoundation` is the tool to grab your screen on Mac OS X.
- `-i <screen device index>:<audio device index>` are the numbers corresponding to your devices listed with the command `ffmpeg -f avfoundation -list_devices true -i ""` above. In this case you can see it's 1 for the screen and 1 for the audio Built-in Input.
- `-c:v libx264` is for H264 format for a good quality video.
- `-qp 0` & `-preset ultrafast` are for a better quality recording.
- `out.mp4` specifies the name and the format of the output file.

Please note that specific codecs are for specific output formats. For example `libx264` is for .mp4 while `libvpx` is for .webm. If you try to record a video in .mp4 format with `libvpx` codec you will get an error.
4. Windows

4.1. ActivePresenter for Windows

4.1.1. Installation

First of all, download the software on its official website (Atomi Systems Inc. 2016) and follow the installation process. It is a standard process for Windows. It means double click on the .exe file downloaded and run the wizard. Once it is installed, the start page should be as illustrated in the Figure below.

4.1.2. Screen recording with ActivePresenter

To start, click on “New Capture” to create a new video. Then a pop-up window is open and allows to choose between different types of recordings. For example, the “Software Demonstration” is just a video, e.g. simple tutorial, while the “Smart Capture With Auto FMR” is a more advanced one, cropped into multiple slides. It is possible then to annotate the slides and create smart paths while browsing.
Then, it is possible to select whether one wants to record the entire screen or a defined area, as well as the audio devices. Finally, when the recording is done, go to the menu bar, click on the application's icon, and resume, stop or discard the recording. Below one can find an overview of the output provided in “Smart Capture With Auto FMR” mode.

As one can see the video is separated in three slides allowing thereby to manage the tutorial slide by slide by annotating it. Then in the bottom of the picture, it is possible
to edit the tutorial directly before publishing, by for example adding an audio track. Finally, the export provides one only video composed with the several edited slides as illustrated below.

5. **Shotcut: a free, open-source and cross-platform video editor and converter** *(Meltytech LLC 2016)*

Shotcut is available for Windows, Linux and Mac OS X. It is free. And it supports multiple file formats as well as various features which are interesting for such software in its range. It is quite complete so we will not see here all details but rather go through its main points of interest with the screen shots below. One can download Shotcut from its official website download's page. Once Shotcut is downloaded and opened, the layout is as such:
As one can see, by clicking “File → Open” it is possible to select the video file to edit. Then, click on the “Timeline” icon to open the timeline. Finally, drag and drop the file from the centre area to the timeline. The file is ready to be edited.

To remove the unnecessary seconds at the beginning or end of the recording, e.g. when you launch or stop the screen recording tool, you just have to crop at the right time slot + select the part to be removed + BACKSPACE or DELETE.

To crop the track, just position the cursor at the right time slot in the timeline and click on the “Split at playhead” icon as presented in Figure below. This Figure also shows that you can add various filters on the different parts of the track. As illustrated in the two Figures below, you can add, for example, a text area in the second part of the video.
Here you can see we added three supplementary video track for all the cropped parts of the video + a CERN logo. Finally, you can export the video in several formats by clicking on the export menu and selecting the appropriate parameters, e.g. .mp4 format with H264 video codec and aac audio codec.
6. Conversion and format issues

It can happen that video formats create issues while trying to play a video in a browser. This is mostly due to patent issues. For example, we identified on Firefox for Mac OSX that a .mp4 video could not be watched correctly sometimes. In the meantime, the same video played without any problems on Firefox for Ubuntu. Below are some steps to be sure your video is well supported by every browser.

.mp4, .webm and .mov are three formats currently supported.

6.1. Format issues

This information is taken from w3schools (2016) and kentuckyfriedtakahe (2016)

6.1.1. Using the HTML <video> element

To show a video in HTML, use the <video> element like this example:

```
<video width="320" height="240" controls>
  <source src="movie.mp4" type="video/mp4">
  <source src="movie.mov" type="video/mov">
  <source src="movie.webm" type="video/webm">
Your browser does not support the video tag.
</video>
```

As you can see there are three <source> elements for the video. Multiple <source> elements can help if one format is not well supported, the browser will play the first recognized video. So at this point we recommend to have the video available in more than one format to prevent issues.

.mp4, .webm and .mov are three formats currently supported by <video> element. The controls attribute add buttons control like play, pause and volume. The text between the two elements is shown only if your browser does not support the video tag.

7. Appendix

7.1. Use dedicated scripts with FFmpeg

To simplify user's experience some basic shell scripts automate the process of tutorial recording. For the moment these scripts can be found on the Github page of Racine (2016).

In this Github repository you will find two main folders. Folder named “video_with_audio” contains two scripts (video.sh and cut_convert.sh). They allow you to record the video (video.sh) in the same process (audio & video altogether) and
then cut your file at your preferred time slots and convert file in both .webm and .mov formats to stream well on every browser (cut_convert.sh). Then finally rename your final file as you want.

For several reasons, it could be better to record the audio and the screen separately. For example to limit the CPU usage and avoid listening the noise of your computer’s ventilation. Or for example if you need to do some editing tasks (e.g. accelerate) on your video before to stream it.

To do this, folder named “video_audio_split” contains four scripts (video.sh, audio.sh, merge.sh and cut_convert.sh). They allow you to record the video in two stages. First the image (video.sh) and then the audio stream (audio.sh). Then you merge (merge.sh) the video and audio files. Finally you can cut your file at your preferred time slots, convert it in both .webm and .mov formats to stream well on every browser (cut_convert.sh) and rename your final file as you want. In brief with this process in two stages the structure is as such :

1. Run the video.sh script to record only the video stream
2. Run the audio.sh script to record only the audio stream while watching the video previously recorded. The audio file should be in the same folder as the video file previously recorded to allow merging files in a second step.
3. Run the merge.sh script to have one only final video.
4. Run the cut_convert.sh script to cut the original merged .mp4 file at the time slots you desire. Then it will convert automatically the cut.mp4 video in the both .webm and .mov formats to stream correctly in every browser. Finally it will rename your final three files (.mp4, .mov, .webm) as you want while keeping the original (non-cut) audio and video files.

The scripts are interactive and will ask you fore some informations (e.g. your screen resolution) in order to adapt the ffmpeg commands according to your Operating System (Linux or Mac). If you don't know some of these values, you are prompted with some friendly commands to discover them. When the scripts ask you to enter the path to save your files, please use auto completion or do not forget to end the path with a slash “/”.

7.2. Conversion with FFmpeg

Note that those commands are for ffmpeg but the conversion process can also be done in iMovie for Mac users or with Shotcut as just mentioned above.

These commands work for Linux and Mac.

Type those commands :

```
ffmpeg -i path/to/out.mp4 -c:v libvpx -crf 18 -b:v 0 -c:a libvorbis path/where/you/want/out.webm
```

for a video in .webm format or :

```
11.08.2016  14  Alexandre Racine
```
ffmpeg -i path/to/out.webm -c:v libx264 -crf 18 -b:v 0 -c:a aac path/where/you/want/out.mp4 for a video in .mp4 format

Note in the commands above, libvpx is for .webm and libx264 is for .mp4

Press Enter to start converting.

The conversion process takes several minutes.

Command options :

- `-i` is the name of your input (in this case out.mp4 or out.webm)
- `out.mp4 / out.webm` are the names of the inputs
- `-c:v libvpx` or `-c:v libx264` are the codecs to convert in .webm / .mp4 format
- `-crf 18` is the Constant Rate Factor. It changes the quality. 0 is for the best quality but is really too long to convert. A good practice is around 18 - 20
- `-c:a libvorbis` is the codec for the audio in the .webm video
- `-c:a aac (for example)` is the codec for the audio in the .mp4 video
- `out.webm / out.mp4` are the names and the formats of the outputs you want

### 7.3. Editing videos with ffmpeg

#### 7.3.1. Cutting a video

These commands work for Linux and Mac.

To avoid showing terminal at the beginning and at the end of the videos you can cut it at the time slots you want. You can also do this if you simply want to have just a part of a video. To do this type this command :

`ffmpeg -i path/to/your/video.mp4 -ss HH:MM:SS -t HH:MM:SS -async 1 path/to/your/cut/video.mp4`

Command options :

- `-i` specifies you want to give an input file
- `path/to/your/video.mp4` is the path and the name of your video
- `-ss [start time]` is the starting point where you want to cut
- `-t [duration sequence]` is the duration you want (for example if you want to have the video cut from the 4th second to the 20th second of the original file, you have to specify `-ss 00:00:04 -t 00:00:16` (i.e. $4 + 16 = 20$)
- `-async 1` is to keep the audio while cutting the video
- `path/to/your/cut/video.mp4` is the path and the name of your cut video
7.3.2. **Accelerate a part of the video**

If your video is too long maybe you would like to accelerate some parts of it which are less interesting. You can do this by cutting your video in multiple parts as saw above, accelerating the desired parts and then concatenate again all the parts to have one only video as output. Here is the command which allow to accelerate:

```
ffmpeg -i path/to/your/video.mp4 -filter:v "setpts=[speed]*PTS"
path/to/your/accelerated/video.mp4
```

Command options:

- `-i` specifies you want to give an input file
- `path/to/your/video.mp4` is the path and the name of your video
- `-filter:v` specifies you want to accelerate only the video and not the audio
- "setpts=[speed]*PTS" e.g. "setpts=0.3*PTS" specifies how speed you want to accelerate the video. For example “0.5” is 2x. Shorter the number is, speeder the video is. As opposed, if you want to slow down, you have to specify a number bigger than 1.

- `path/to/your/accelerated/video.mp4` is the path and the name of your accelerated video

7.3.3. **Concatenate the multiple parts to have one only video output**

After for example cut in multiple parts and speed up the needed parts, concatenate the parts to obtain one only video output. To do this, type this command:

```
ffmpeg -f concat -i <(for f in ./part*.mp4; do echo "file '$PWD/$f'"; done) -c copy output.mp4
```

For any further information about ffmpeg, refer to the official website ([Bellard 2016](https://www.ffmpeg.org/)) and the official wiki documentation ([Edgewall Software 2016c](https://ffmpeg.org/documentation.html)).

For any further information about Quicktime Player, refer to the official documentation ([Apple Inc. 2007](https://www.apple.com/quicktime/)).

For any further information about video formats issues on the web, refer to those pages ([w3schools 2016](https://www.w3schools.com/video/) and [kentuckyfriedtakahe 2016](https://www.kentuckyfriedtakahe.com/videoformats/)).
8. Bibliography


