

# Producing Video Tutorials With Open Source Tools

Abe Kazemzadeh

University of St. Thomas, St. Paul, MN  
For SCALE18x, Pasadena, CA

2020-03-08



# Motivation

- I've been teaching about 1.5 years
  - University of St. Thomas (St. Paul) Grad Programs in Software
- Before that, software industry here in Southern California
- I needed to learn how to record my lectures
- I also needed to make software tutorials
- I wanted to use Open Source tools

# Goals

- Share experiences about making video tutorials and lectures
- Two-way sharing of experiences is ideal: please share your experiences as well

# Overview

- 1** Intro
- 2** Use Cases
- 3** Techniques
- 4** Tools
- 5** Discussion

# Recording a Lecture

- A normal classroom experience but it's recorded
  - Students can consume lectures remotely if they are sick or travelling
  - Students can rewatch difficult material
  - Beneficial for English-as-a-Second-Language (ESL) students
  - Since my university provides classrooms equipped with recording tools and proprietary software, I won't go deep into this

# “Flipped” Lecture

- Normal lecture: lecture in class, homework/project at home
- “Flipped” lecture: recorded lecture watched at home, homeworks and projects in class.
- Seems like a good idea but I didn't have any video recording or editing skills

# Video Tutorial

- Video tutorials illustrate a skill using video instead of text
  - Our scenario: the tutorial is about using some software so the base case is screen capture + voice
  - I'll be using this case as an example:
    - Example video posted on [YouTube](#): how to make a Sozi animation
    - Sozi is a way to make animated presentations, e.g. this [overview of my presentation](#)

# Inspiration: Citizen Journalism

- Public citizens play the role of traditional, professional journalists
  - E.g. Arab Spring, Japan Tsunami, Ferguson unrest, Occupy Wall Street, Haiti Earthquake, etc.
  - Often relies on open source tools, new media platforms, and mobile phones
  - USC Annenberg project: a smart phone app that guides the user to record specific shots, in order to foster the creation of higher quality video content

# Video Screen Capture

- Video screen capture is a recording of the screen and usually includes audio narration
- Useful for demonstrating features of software
  - Especially useful for teaching software that has GUIs (graphical user interfaces)
  - Also useful for filing bug reports
- By default, screencasts are real-time, but after-the-fact editing can be useful
  - Fixing mistakes, shorten the video
  - Adding other footage to make it more interesting, e.g. a talking head

# Video Editing

- Video editing involves cutting, arranging, and other transformations of video shots
- Editing is a big aspect of storytelling using video
  - Difference between plot and story
- Editing video tutorials can make your videos shorter
  - Saves your viewers' time...
  - ...but costs you your time

## Open Question

How much value is added from editing and adding other footage besides screen+audio?

# B-Roll

- “B-Roll” refers to supplementary or alternative footage
  - The main footage is called “A-Roll”
  - Our case: the screen recording is the A-Roll
- B-Roll can be other video or still images
  - E.g. talking head to go with narration, hands/pointing, logos/branding, etc.
- Helps establish context and adds dramatic support
- Helps prevent jump cuts, i.e. two shots that are very similar that are edited to be temporally adjacent
  - The result with B-Roll in between the cut is called a “cut-away shot”

## Tip:

Collect B-roll footage. B-roll footage can give context and depth to the video and it can break up jump cuts.

# Shot Sequences

- Combining A-Roll and B-Roll builds up shot sequences
  - Like software design patterns for cinema and television
- The “5-Shot Sequence” is a formulaic sequence in journalism composed of the following 5 shots
  - ① A close-up of the hands
  - ② A close up of the face
  - ③ A wide shot of the environment with the subject in it
  - ④ An over-the-shoulder shot showing the action from the subject’s perspective
  - ⑤ A creative shot to capture an unusual perspective or something else the viewer should know
- I was inspired by an app done by Melissa Loudon and Andrew Li at USC Annenberg School of Communication and Journalism
- Example

# Animation

- Screen capture of is itself a type of animation
  - E.g. mouse/cursor movement and slide shows
- Sozi is an open-source version of Prezi, which uses an SVG canvas to build animated presentations
- Otherwise animation is a large topic and we won't cover animation outside of screen capture

# Audio Synchronization

- If you are using multiple streams of video, you'll want to make sure that they are all synchronized
- It's easier to synchronize video using audio
  - Audio sampling rate  $>>$  video sampling rate, so there is more resolution to work with
- A clap (impulse noise) can be used to mark the synchronization point

## Tip

Two claps can be used to mark an error that you want to edit out

# Room Tone

- If you record in different acoustic settings, the difference background noise creates noticeable changes when editing
  - E.g., heating/ventilation, traffic, lights/electrical
  - This gives the video an unpleasant, amateurish feel
  - Especially noticeable when going from ambient recording to complete silence

## Tip:

Record 30 seconds of “room tone” to smooth out audio during silences.

# Effort Estimation

- Effort estimation for video production seems to be similar to software engineering, i.e., it's difficult to estimate

## Tip:

Though it is difficult to estimate the time required to edit video, the effort required roughly increases with the ratio of input footage to output footage.

- Setting things up and rehearsing or having a script is much easier than trying to edit multiple takes

# Microphone

- An external microphone will help record better audio
  - If you use your computer microphone, you may hear typing noises, especially for software tutorials

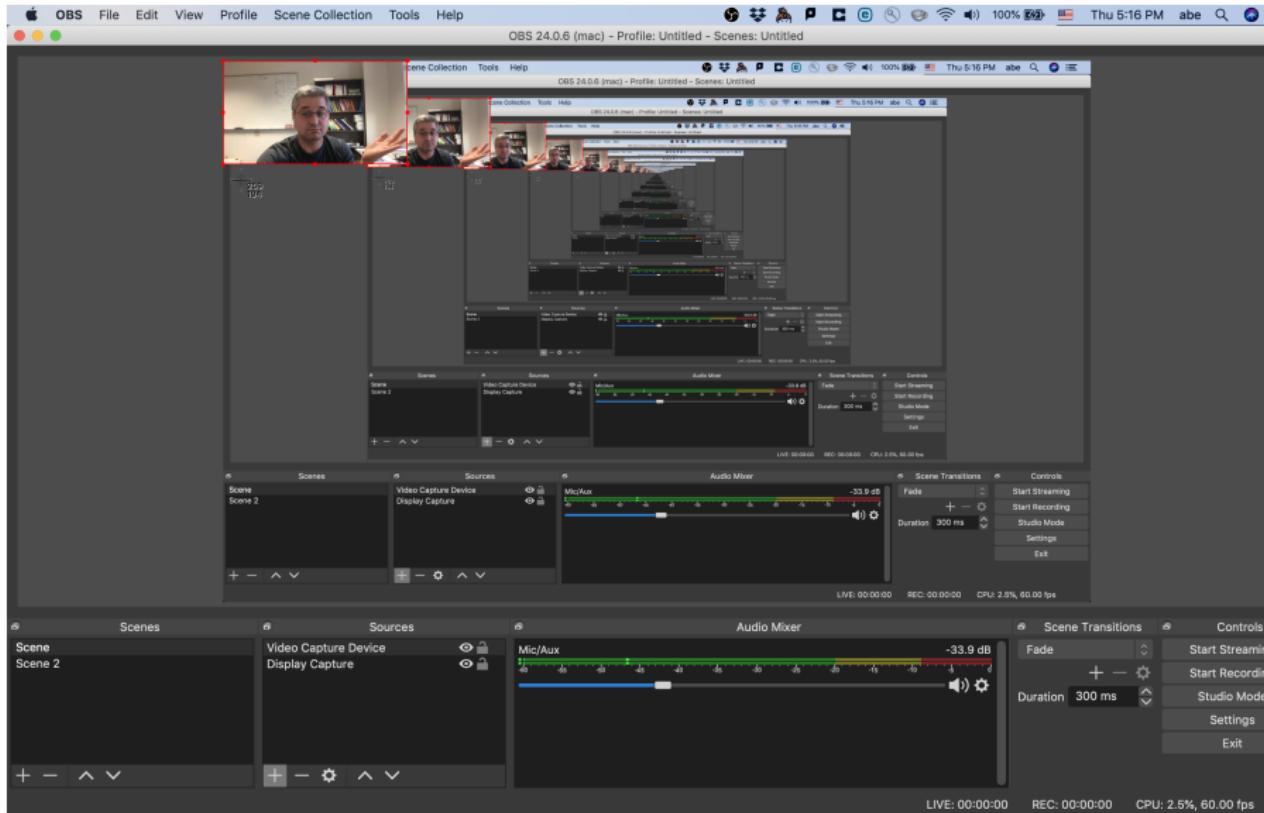
# Microphone



# Clapper



# OBS Studio



# OBS Studio

- Open Broadcaster Software (OBS) Studio is an open source screen capture software
  - Has options for both screen capture and streaming (live screencast)
  - Sponsored by Twitch, among others
- Outputs a video file (mkv-format) or streams to a server
- Multiple sources are-mixed into a single file or stream

## Tip:

To be able to record two separate filesstreams (screen+webcam) using OBS Studio, you need to start two instances of OBS Studio, and start the second instance on the command line.

- The more I used OBS Studio, the more I realized having two streams wasn't necessary

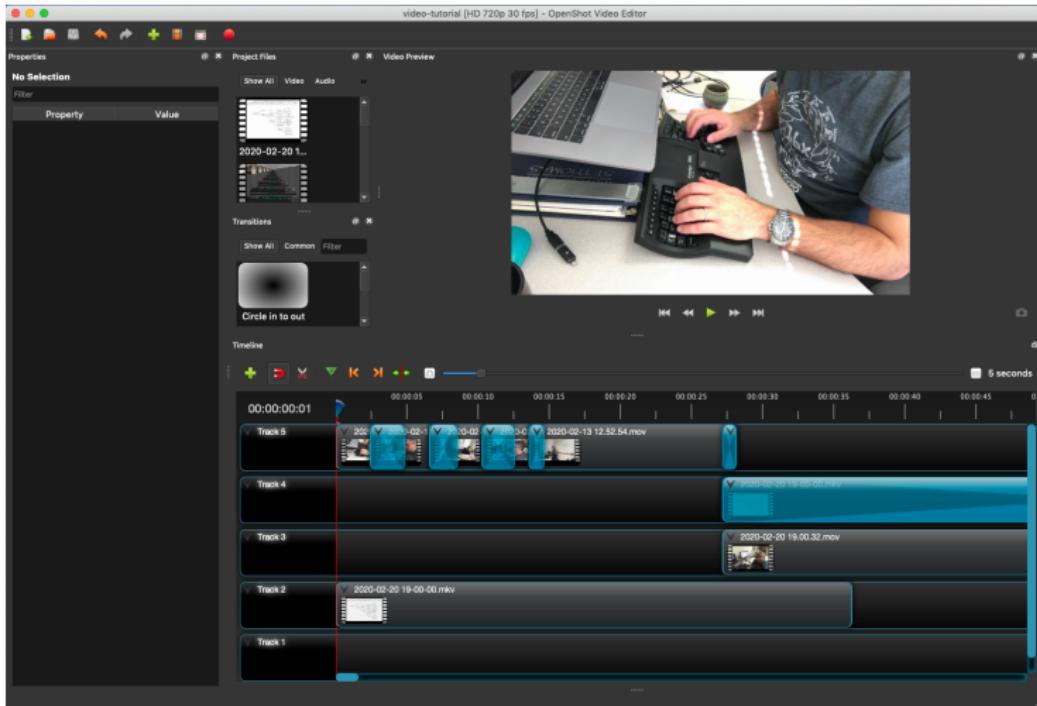
# OBS Studio

- Another issue I had was showing the mouse when using window capture (as opposed to screen capture)

## Tip:

To record mouse movements, use whole screen capture instead of individual window capture.

# OpenShot



# OpenShot

- **OpenShot** is an open source video editing tool
- Editing is necessary if...
  - ① you want to mix multiple video sources (e.g., 5-shot sequence)  
or
  - ② you want to cut out unnecessary material (i.e., not live or real-time)
- Some editing type features are available when recording live from OBS Studio (e.g., switching from one scene to another)

# Smart Phone

- This is needed if you decide to produce B-Roll footage
- Using a file synchronization service like Dropbox can make editing the B-Roll footage from the phone very convenient because it doesn't need to be manually moved to the computer

# Tripod



# Tripod

- A tripod will help with B-Roll footage if you are producing the video by yourself
- There are many options for “selfie” type tripods for phones
  - Comes with Bluetooth remotes for starting/stopping recording
  - ~\$25

# Sozi

The screenshot shows the Sozi application window. On the left, there is a large white canvas area where a flow diagram is displayed. The diagram consists of several rectangular nodes connected by arrows, representing a process flow. The nodes are labeled: brain, idea, editor, distill, commandline, svgimage, sozi, htmljson, and browser. Arrows indicate the flow from brain to idea, idea to editor, editor to distill, distill to commandline, commandline to svgimage, svgimage to sozi, sozi to htmljson, and finally htmljson to browser.

On the right side of the window, there are several configuration panels:

- Frame**:
  - Title: New frame
  - Id: frame6385
  - Timeout (seconds):
- Layer**:
  - Copy layer: Select a layer to copy
  - Outline element Id:
  - Layer opacity: A slider set to 100%
- Transition**:
  - Duration (seconds):
  - Timing function: Linear
  - Relative zoom (%):

At the bottom of the window, there is a toolbar with various icons for file operations like Open, Save, Print, and a zoom slider. Below the toolbar, there is a row of buttons labeled 1 through 13, each followed by the text "New frame". The button for frame 13 is highlighted in blue, indicating it is the active frame. At the very bottom, there is a footer bar with a "Default" tab and other unlabelled tabs.

# Sozi

- **Sozi** creates animated diagrams from SVG images for snazzy presentations
  - Given SVG image(s) create animations that pan, zoom, and rotate using the images as layers
- Open source version of Prezi
  - User interface takes some getting used to: good candidate for a video tutorial
- See demo video tutorial @ t=769

# Graphviz

- Graphviz is an open source graph visualization system
- Generates graph diagrams from a declarative specification
- Used to create the SVG input to Sozi
- Consists of a graph specification language, DOT, and command line tools to generate/render output graphs
  - Also useful in the L<sup>A</sup>T<sub>E</sub>X toolchain
- See demo video tutorial @ t=204

# Graphviz

The screenshot shows a Mac OS X desktop environment with several open windows. In the top dock, there are four terminal windows labeled 'bash', 'emacs-26.2', 'ec2-user@ip-172...', and 'emacs-26.2'. Below the dock, an Emacs buffer titled '1. emacs-26.2' is visible, containing the following code:

```
# brain -> idea -> editor -> dotfile -> commandline ->
# svgimage -> sozi -> htmljson -> browser

digraph sozi {
    rankdir = LR
    node [shape = record]

    brain -> editor [Label="idea"]
    editor -> commandline [Label="dotfile"]
    commandline -> sozi [Label="svgimage"]
    sozi -> browser [Label="htmljson"]

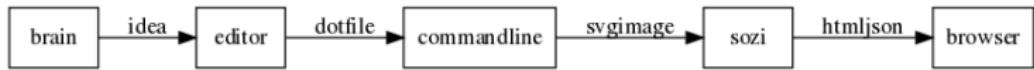
}
```

At the bottom of the screen, the terminal window title bar displays the file name '-UU-:----F1 sozi-process.dot' and the status bar shows 'All L1 Git-master (Fundame'.

# Graphviz

- Graphviz is an open source graph visualization system
- Generates graph diagrams from a declarative specification
- Used to create the SVG input to Sozi
- Consists of a graph specification language, DOT, and command line tools to generate/render output graphs
  - Also useful in the L<sup>A</sup>T<sub>E</sub>X toolchain
- See demo video tutorial @ t=204

# Graphviz



# Graphviz

- Graphviz is an open source graph visualization system
- Generates graph diagrams from a declarative specification
- Used to create the SVG input to Sozi
- Consists of a graph specification language, DOT, and command line tools to generate/render output graphs
  - Also useful in the L<sup>A</sup>T<sub>E</sub>X toolchain
- See demo video tutorial @ t=204

# \LaTeX/Beamer

- \LaTeX and Beamer were used for this presentation slides
- Fits more with the online lecture format but could also be used for video tutorial format
  - E.g., switch between the main screen (tutorial subject) and the slide deck
  - The slide deck can help keep a video tutorial on track

# Acknowledgements

- Matthew Lynn: visual effects specialist
- Melissa Loudon, Andrew Li: citizen journalism inspiration
- Eric Level: teaching and classroom video
- SCaLE organizers, esp. A/V team

# Conclusion

- It's fun and not difficult to make video tutorials with open source tools
- I hope that this presentation makes it easier to see a general lay of the land and one specific end-to-end example

# Discussion Topics

- Related experiences to share
- Questions
- Opinions:
  - Is extra footage apart from screen capture useful (e.g. talking head)?
  - Video tutorials vs text/readme
- If there's time, we can watch [the demo video](#) or drill down into specific topics