Report
"Creating a Python environment for video editing".

11.06.2021

Table of contents

1 Project Objective

The aim of the project was to create a Python environment for film editing with the help of available bilbians. The environment can be used to generate movies in different forms and with different effects.

2 Libraries used

2.1 MoviePy

MoviePy is a library that allows you to make cuts, joins, transitions between video clips/photos. It allows you to insert subtitles to movies or music.

2.2 numpy

Numpy is a library for using various mathematical operations (such as sine).

2.3 matplotlib

Matplotlib is a library mainly used for drawing mat- graphs. It has been used to generate animations of functions and vectors.

2.4 librosa

Librosa is a library for managing music files and includes tools for handling them.

2.5 PyQt5

Library used to generate GUI (graphical user interface).

2.6 cv2

cv2 is a library containing a collection of algorithms for image processing.

2.7 random

random is used to generate random numbers

3 Workstation configuration

The workstation should be configured before starting work.
Minimum Requirements:

- Windows, Linux
- 4GB Ram (more recommended)
- Python 3.8 + (Anaconda 4.10.1 was used)
- The above libraries (you can use pip install #package-name) Optional

add-ons:

- Jupyter Notebook (a useful tool used in data-science)

- Image-Magick (if you want to use subtitles in your movie)
  https://www.imagemagick.org/script/index.php

4 Description of the main library

Of the above libraries, the MoviePy library is the most important because it is what generates the movies.

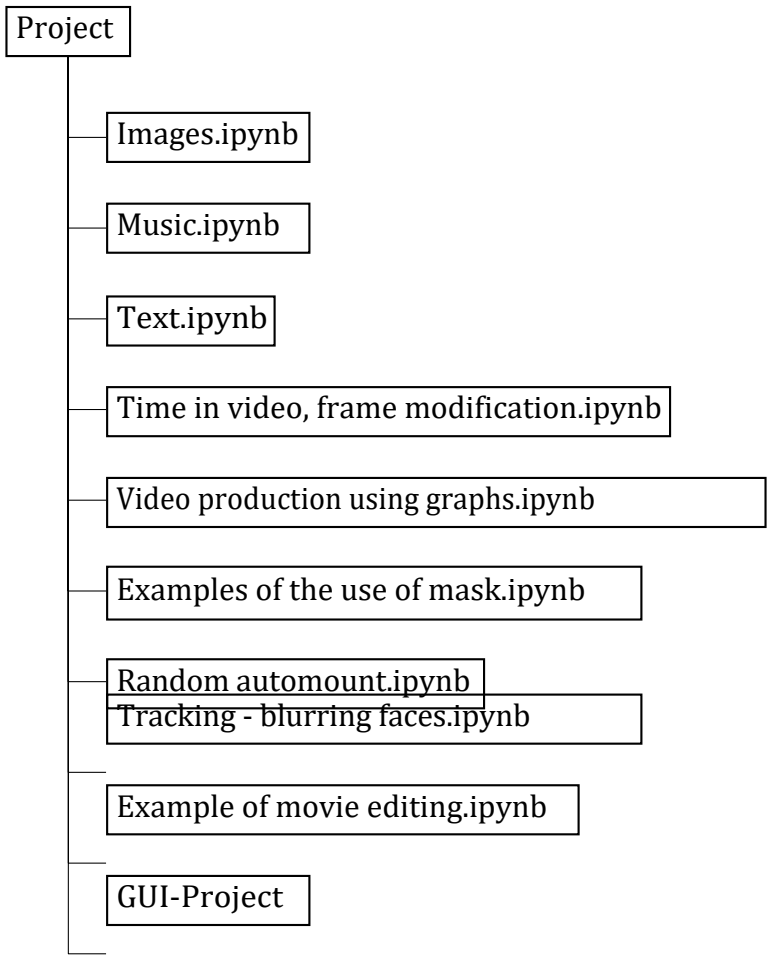
MoviePy is a library that allows you to perform various multimedia functions through programming. It allows you to perform image manipulation such as: cutting, merging, inserting subtitles, video processing (angular processing), creating your own special effects, as well as non-linear video editing or "nesting" of several images in one picture. Additionally, it allows you to manage and modify sounds to the extent sufficient for the selected theme.

5 MoviePy library functions

Examples of the capabilities of the functions provided by the MoviePy library:

1. Tracking and blurring a person's face in a video
2. Combining multiple video clips/photos into one movie
3. Modifying audio in a video (adding, removing, changing)
4. Adding a watermark/subtitle to your video
5. Create transitions between video clips/photos

6 Symbolic program tree



7 Executed programs

7.1 Images.ipynb

Simple research on the use of images/clips in editing.

7.2 Music.ipynb

Examples of music use, experiments with music.

7.3 Text.ipynb

Generating text on video.

7.4 Time in video, frame modification.ipynb

Experiments on the behavior of time in a movie. A change of time in a movie based on a graph was performed. Single frames were also captured and edited (graphic effects, frame reflections)

7.5 Video production using graphs.ipynb

Generating animations of mathematical functions on a movie. I also wrote a short program which makes a music video - it draws vectors to the rhythm of the music.

7.6 Examples of the use of mask.ipynb

Using a mask cut out with GIMP. It was used to generate text that scrolls underneath it, which gives a nice visual effect.

7.7 Tracking - blurring faces.ipynb

Tracking implementation. Specify the tracking coordinates of an object, and then a tracking object (such as a face blur) is applied to the screen.

7.8 Random automount.ipynb

An implementation of the idea where the user selects video clips and music, provides configuration parameters, and then the program, using randomization, edits the entire movie along with the music. A different movie comes out each time, which is interesting and you can get an interesting end result.

7.9 Example of movie editing.ipynb

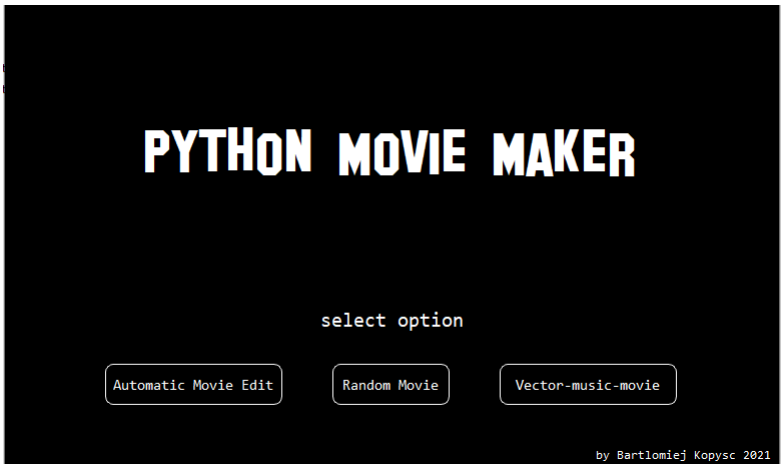
This program, is a pre-built movie montage. It serves to show that you can edit a movie without a graphical editing program, but using only *MoviePy* code and library

7.10 A program with a GUI to edit a movie

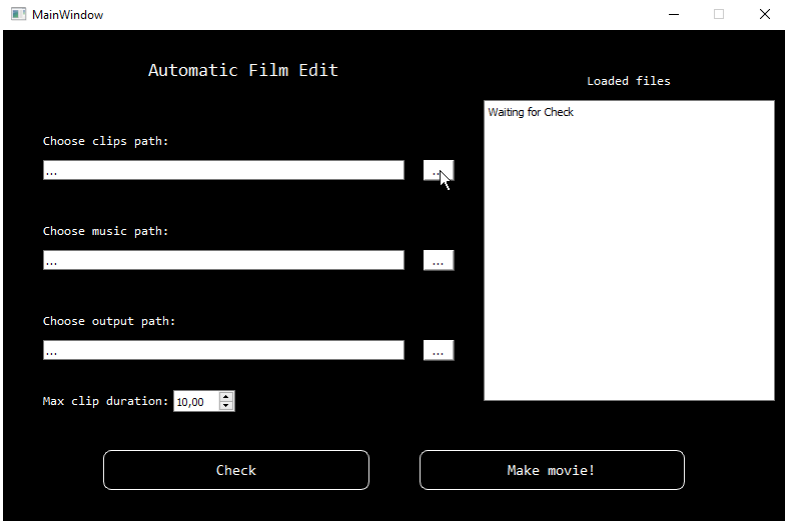
A program concept, where the above examples would be implemented. Easy and clear for the user, it could be used as an interesting tool for achieving various effects with the use of video and music. It implements a method of generating a movie, based on files taken by the user (the so-called auto-assembly). User intervention is minimal. The GUI has been designed using PyQt5.

7.10.1 Brief operating instructions:

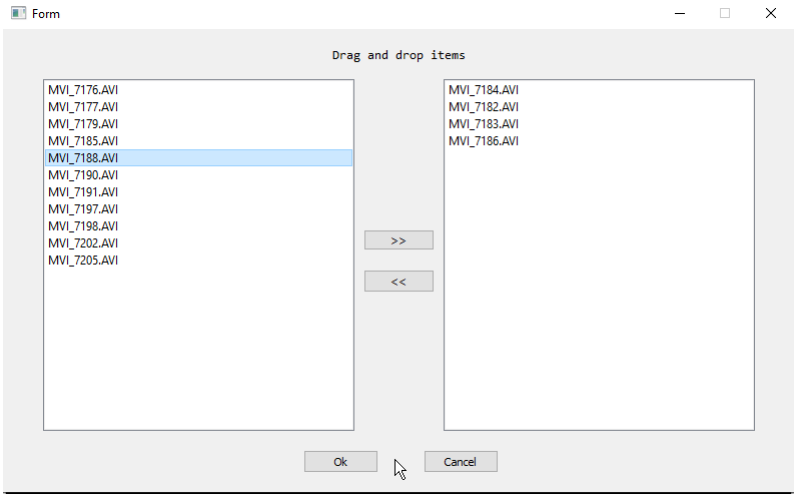
1. Run main.py (or build)
2. Select "Automatic Movie Edit"



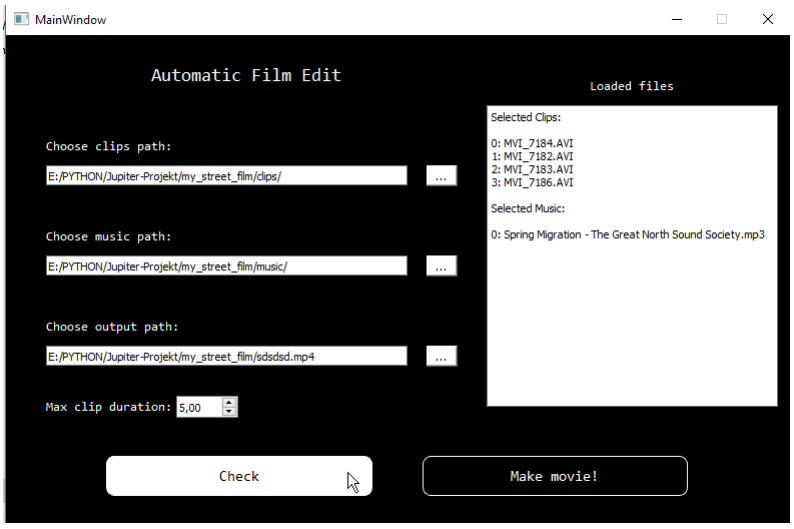
3. Fill in the fields with paths to folders



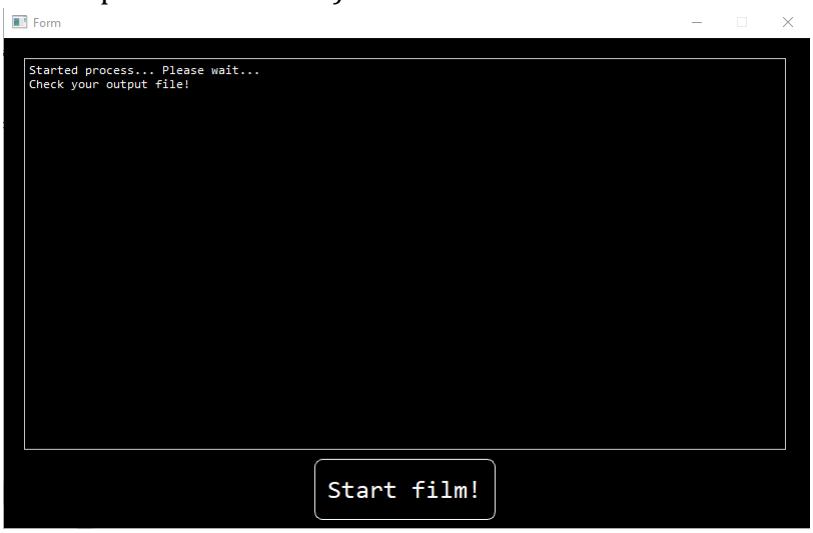
4. Drag elements (files) from the left to the right. Arrange them in the order you want



5. Enter the maximum duration of one clip
6. Click the "Check" button to check the loaded items



7. Select the "Make movie!" button.
8. Start the production of the film with the "Start Film!" button.
9. Wait for the return message, and navigate to the desired location (where the output file is located)



10. The video is already edited.

8 Results

The results are evident in:

- GitHub Repository (code): https://github.com/BKopysc/MoviePython
- YouTube (videos): YouTube link

9 Conclusions

9.1 Results achieved

The project primarily enabled:

- learning a new library
- Partial understanding of information about video codecs
- learning to edit an image as a matrix of pixels
- getting to know the PyQt5 library and creating a GUI in Python
- A broader understanding of the Python language

9.2 What could be improved?

I think it could be streamlined:

- Creation of a full GUI and implementation of different ideas
- Improved RAM management
- Using multithreading to speed up processes
- Improvement of some code elements
- Increased knowledge of GUI building language and tools

10 Glossary

- Video clip - an element that is part of a longer video.
- Watermark - a partially transparent (or not) layer/object applied to an image
- Transition - the effect of switching between one clip and another. The simplest transition is the switch itself. More advanced transitions are usually used to add variety to a movie.
- Face/Object Blur - Blurs the object in a photo/video to hide detailed information about it. There are many algorithms for this, the most commonly used is Gaussian blur.
- Object tracking - Follows the superimposed element (e.g. blur) to a specific object (set of pixels) in the image.
- Non-linear video editing - Access any part of a clip at any time and from anywhere during editing.

11 Documentation

- MoviePy: https://zulko.github.io/moviepy
- CV2: https://pypi.org/project/opencv-python/
- PyQt5: https://pypi.org/project/PyQt5/
- librosa: https://librosa.org/doc/latest/index.html
- matplotlib: https://matplotlib.org
- NumPy: https://numpy.org