CMPT365 Project

Mattias Stroman

mstroman@sfu.ca

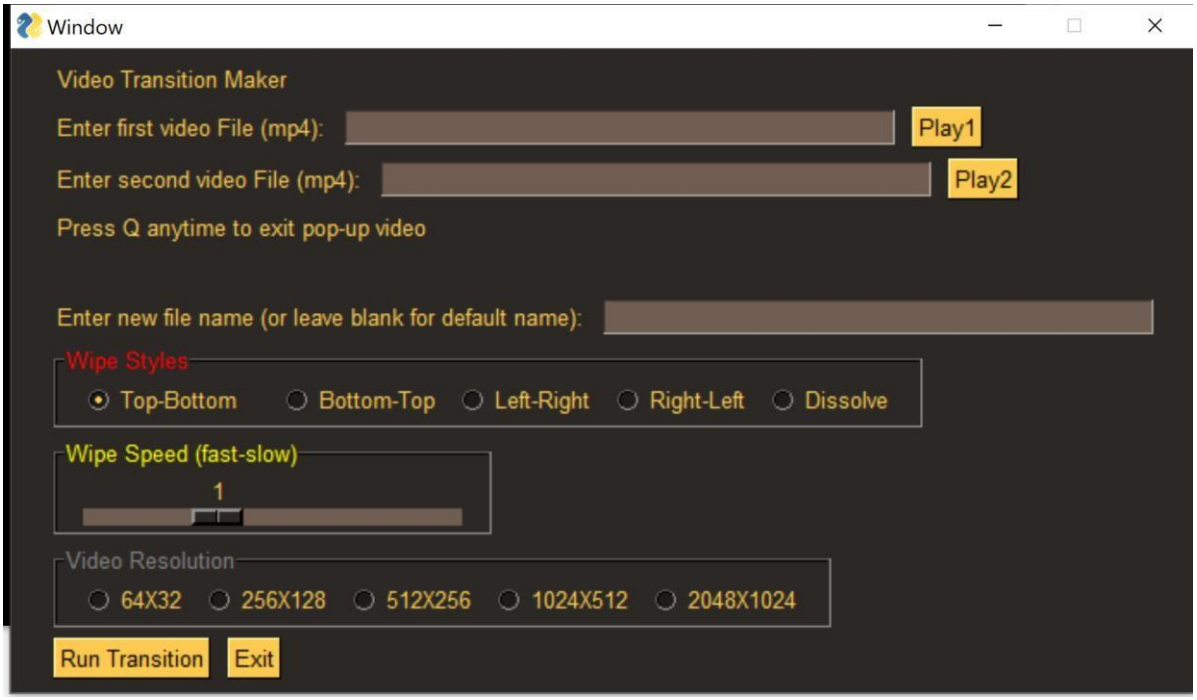
Part 1

Executable File: google_drive **About**

the program:

I used python to code the project. Using common libraries such as numpy, opencv2, and pysimplegui.

These powerful libraries aided me in the making of the project as well as teaching me more about the process of video transitioning. Pysimplegui gave me the tools to make a well organized and simple to use GUI. This GUI makes the process of making transitions in videos easier. Numpy, the most common library of the three I used allows for the use of opencv2 as well as manipulating arrays and matrices in much easier fashion. Lastly, opencv2 gave me the tools to access the individual pixels of each frame of a video. It also made controlling the process of video manipulation streamlined. These libraries were painless to use and easy to code with after the beginning few mistakes.



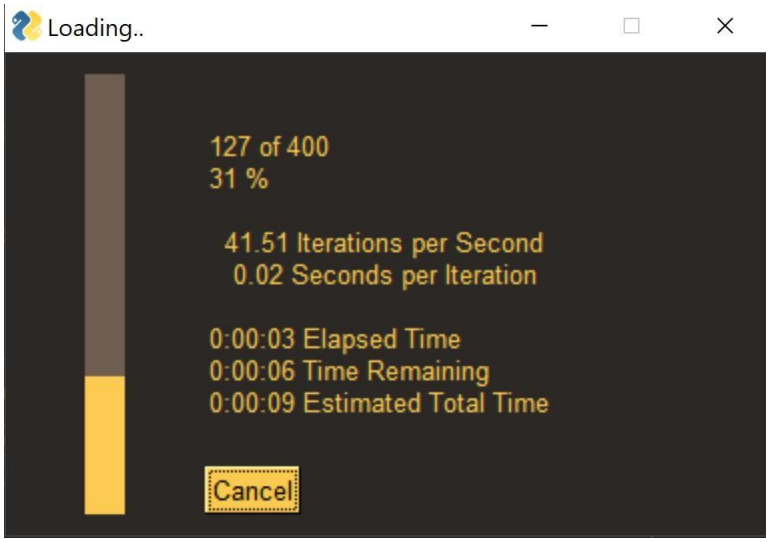
Graphic User Interface:

My goal with making the GUI was to give the most options when creating a transition while still making the design user friendly. By using pysimplegui it gave me the tools to make a powerful yet straightforward interface.

Considering the design:

- Two boxes for video file entry
- Option to play either video file
- Choice to name new video file
- Easy button-based choice to pick desired wipe style
- Simple slider to effectively control transition speed
- Easy button-based choice for resolution
- Simple Run/Exit controls

Loading Screen



Combined with a simple but informative loading pop-up to give accurate information on the current state of the transition process.

Transition Options:

The choices of transitions I made was directly decided by what I though were the most used and therefore the most relevant.

The choices:

- Top-Bottom Wipe
- Bottom-Top Wipe
- Left-Right Wipe

- Right-Left Wipe
- Dissolve

The **Wipes** were all made using similar processes, taking columns of each frame and adding them to another frame of a different video. This process was straight forward enough and allowed for a effective way of transitioning.

Dissolve was trickier to accomplish. What I needed to do was take a spread of the desired frames array and black out pixels then blacking out all the exclusive disjunction of the other frames array, followed by adding one to the other to gain a new frame. This process gave me a better understanding of what it takes to accomplish a proper dissolve effect on a video transition.

Attempted Transition:

I tried to do an eye blinking transition. Why I couldn't pull it off was to do with my inability to map a sin() function to a frames array effectively. What this would take would either to make a more complicated mapping process or gain a deeper understanding of array manipulation.

Part 2

How hard was this project?

I would consider this project to be medium if being completed in a bare bone's way. If a student chooses to push themselves then this project can become intense to do. Whether the area they choose to push themselves is an amazing GUI, large selection of options, or many transitions. This can be a difficult project.

State briefly how would you improve this project?

I would give a list of different transitions that could be done, with accompanying difficulty to each type of transition. This gives students a better idea as what are expectations.

Any ideas for a different project at the same difficulty level? Would you enjoy more difficulty? Less?

A simple photo editing GUI that can-do simple tasks like crop, draw, stretch, etc. This would be about the same difficulty and I think would be fun as well. I think more difficulty could be a good thing if there was a bit more help and pointers as how to accomplish the task.