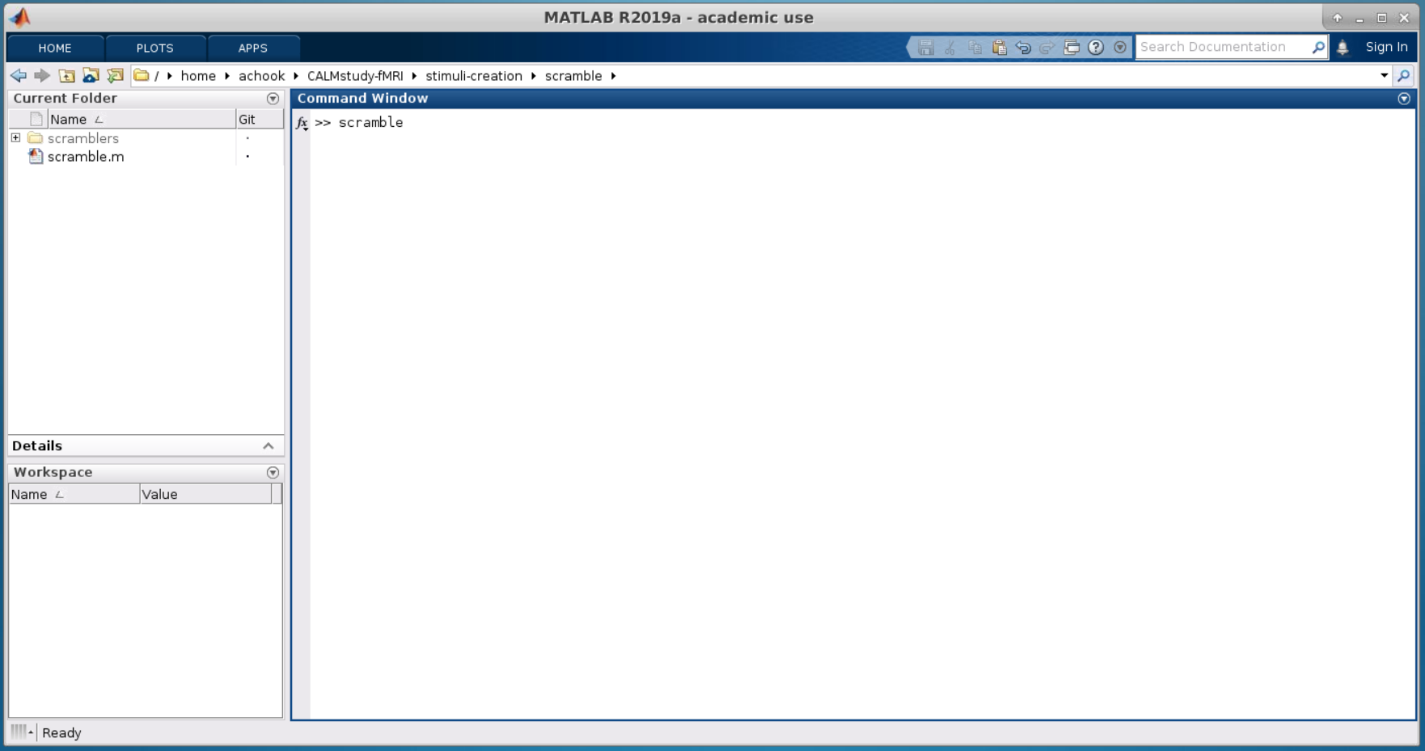
**Instructions for Using the Video Scrambler (*beta version*)**

This scrambler is designed to obfuscate both the image and sound of videos in order to make their contents unrecognizable while still maintaining some low-level perceptual features. The intent is to produce higher quality control conditions for experiments. The sound is obscured using a phase scrambling approach while images are distorted with the diffeomorphic transformations described in Stojanoski & Cusack (2014)[1]. Below are detailed instructions how to use this program written for those who find the idea of running code anxiety provoking. We welcome any feedback for improving the clarity and accessibility of this document.

**SETUP:**

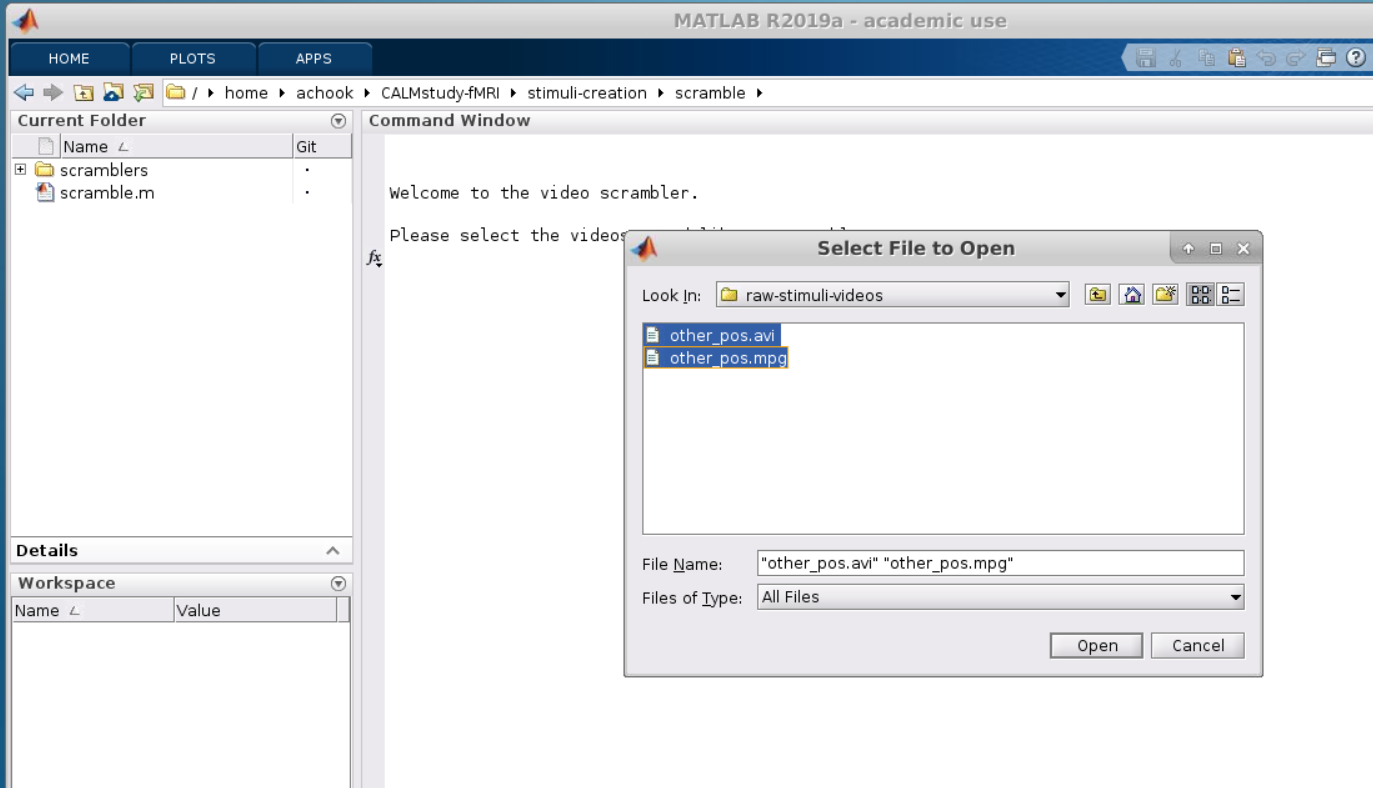
1. Install MATLAB. This code also requires the Computer Vision System Toolbox so make sure you install this too. This version of the code was built using MATLAB version 2019a and
2. Download the scrambler code available at (get code from Meg for now) and save it somewhere on your computer.
3. Open MATLAB. Use the file navigator pane to click to the folder containing the scramble.m file.
4. Type scramble in the command window and press enter to start the program.



**4.**

**3.**

**USING SCRAMBLE:**

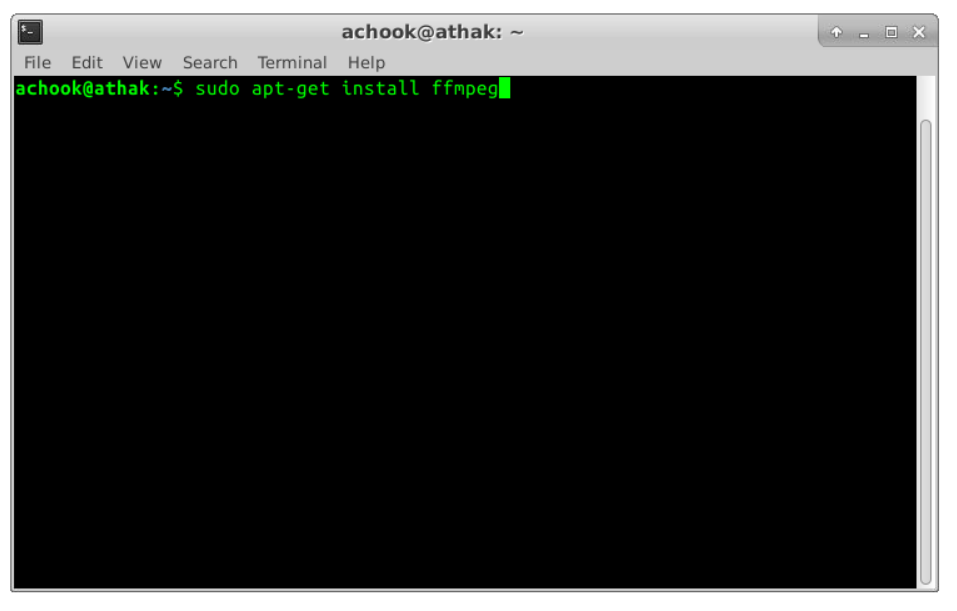
1. Once the program starts you will be asked to select which files to scramble. Multiple files can be selected.
   1. Notice in the screen shot that text will also be printed to the command window. This will often have extra useful information. Sometime warnings will also be displayed, however most of them are only meant to provide helpful troubleshooting suggestions if you find that scrambling has failed. Under most cases these can be safely ignored.
   2. Video formats (*e.g.* avi, mp4, etc) that are supported are determined by the If you find that your video format

**1.**

1. Next you’ll be asked to pick a folder to save the scrambled videos to. All scrambled files will use the original filename with “scrambled-“ prepended.
2. Then it will ask you what warp parameter to use. This is somewhat arbitrary, and some suggested values are provided, however you’ll have to play with this to get the look you want. Numbers must be whole positive value.
   1. *Note that per the original experiments reported in Stojanoski & Cusack (2014), faces will require higher amounts of distortion than other objects to become unrecognizable.*
3. Finally, you will be asked what amount of phase shifting to apply to the audio. Again, this will be something you will need to experiment with the get the sound you want.
   1. Although you can specify larger values, essentially 2\*pi is the largest amount of distortion you can apply. This is equivalent to creating noise out of all the component frequency/amplitudes present in you roriginal sound. To obscure speech with this approach, you will likely need to use this amount of phase shifting.
4. Now the program should pretty much do its own thing as it works on each of your video files. There will be a lot of stuff that get printed to the command window however most of this is just information about what steps are being done. A message will print in the command window to let you know that everything is finished.
5. One important point however is that in order to compress the video, the program attempts to make a system call to an outside program called ffmpeg. If this is not installed on your system, then the files will main uncompressed.
   1. It is fairly easy to install the ffmpeg libraries. If you are using Ubuntu and have permissions to install on your system, you can open a terminal and type:

sudo apt-get update

sudo apt-get install ffmpeg



* 1. However, if you do not wish to or are unable to install the ffmpeg libraries, there are many good free options worth exploring.

1. Also be aware that under the current implementation this code is not well designed to handle very large movies. You may find that you get an *Out of Memory* error. If this happens, you can try scrambling just one file at a time and/or restarting MATLAB between runs which will clear the data currently in memory.
   1. There are also some cases of video that have not had code written to handle them. This will throw and error which asks you to submit a feature request. If either of these become issues for you please let me know, and I’ll do my best to push a more robust version to you!

**References**

Stojanoski, B., & Cusack, R. (2014). Time to wave good-bye to phase scrambling: creating controlled scrambled images using diffeomorphic transformations. Journal of vision, 14(12), 6-6.