Table of Contents:

Requirements
Basic Walkthrough
Addendum
Addendum 2
Addendum 3
Default Settings

Requirements

This guide requires that you have R Studio and Blender already installed, with HandBrake as an optional (but highly recommended) install as well. All three programs are free to download and use, from the links supplied below.

R Studio can be downloaded here (select the free option):
https://www.rstudio.com/products/rstudio/download/
Blender can be downloaded here:
https://www.blender.org/download/
HandBrake can be downloaded here:
https://handbrake.fr/

Note: this manual assumes that your .db is formatted in such a way that there is a table called cohorts, with columns cohortID, year, species, trees, diameter, and height. The R script's SQL calls may need to be edited if this is not the case.

Basic Walkthrough

Step 1: R Studio

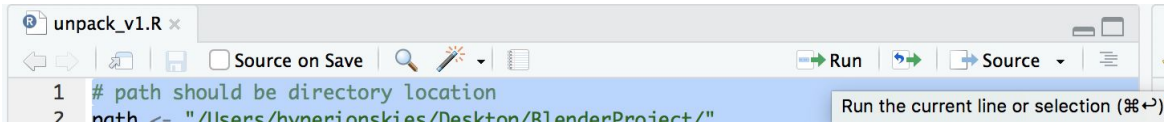
I. Open unpack_v1.R in R Studio
II. Check the *path*, *output*, and *database* variables. Make sure *path* is the path to the directory your database is in and where you want your text file (.txt) to go. *Output* and *database* should have the file names for your desired .txt and the database you're using, as shown in green below:

```
# output adds desired output name; should end in .txt for text file
output <- paste(path,"/matrix_file_nr.txt",sep="")


# set up database access/library calls/driver
database <- paste(path,"/db_cohorts.db",sep="")
```

III. Select all of the code; you can do this quickly with ctrl+A on Windows or cmmd+A on Mac.

IV. Click “Run.”



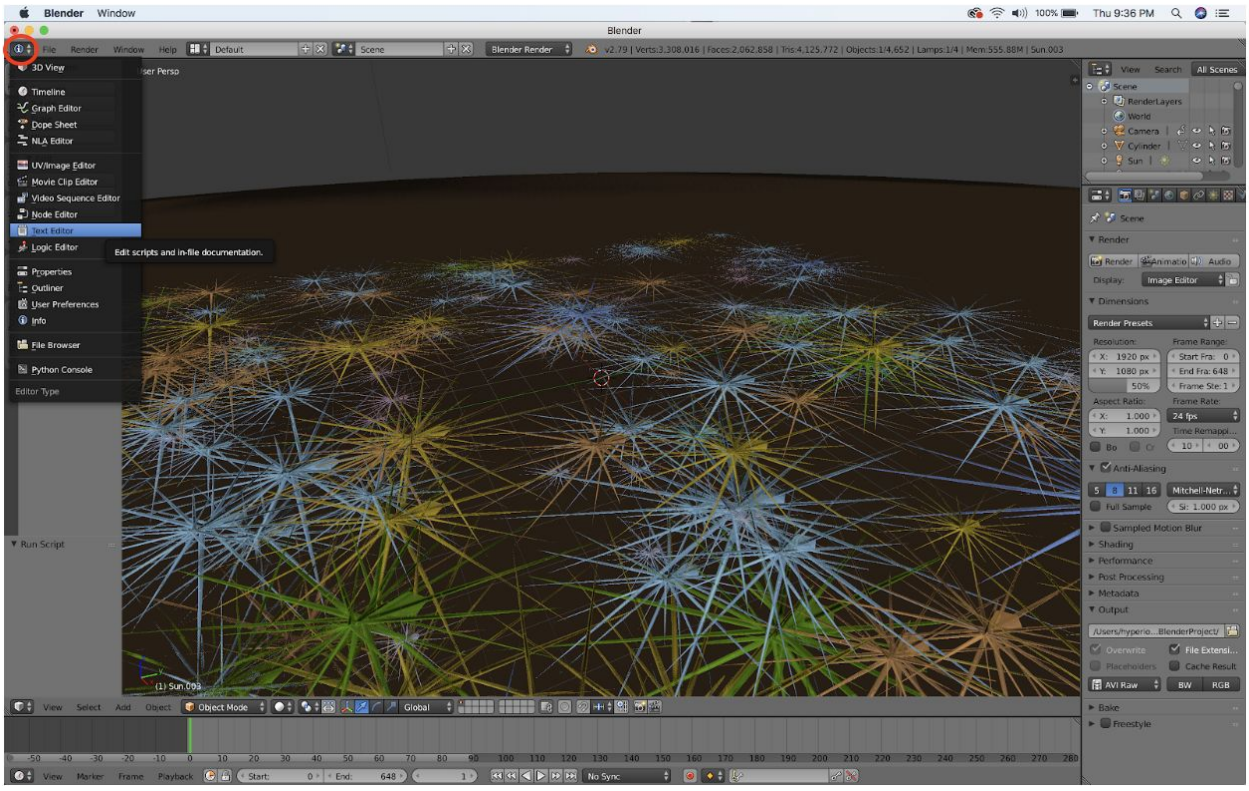
- Note: This script will install the DBI and RSQLite library packages if they are not already installed.

V. You now have a .txt file with the necessary data for the last step.

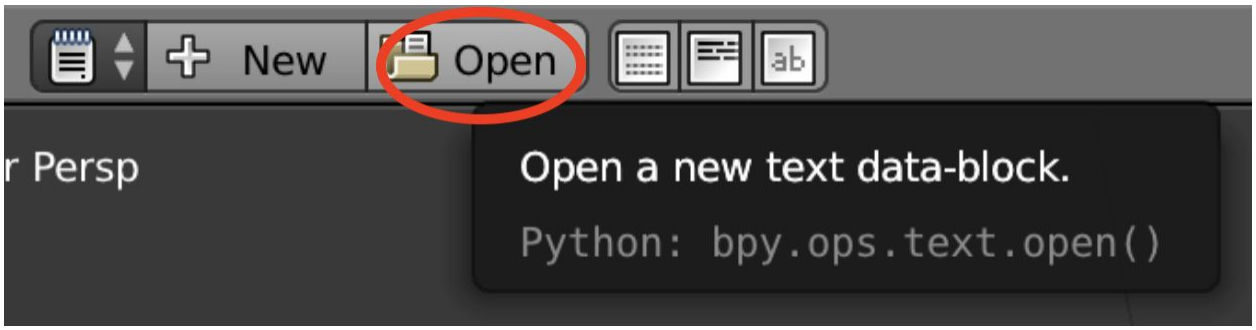
Step 2: Blender

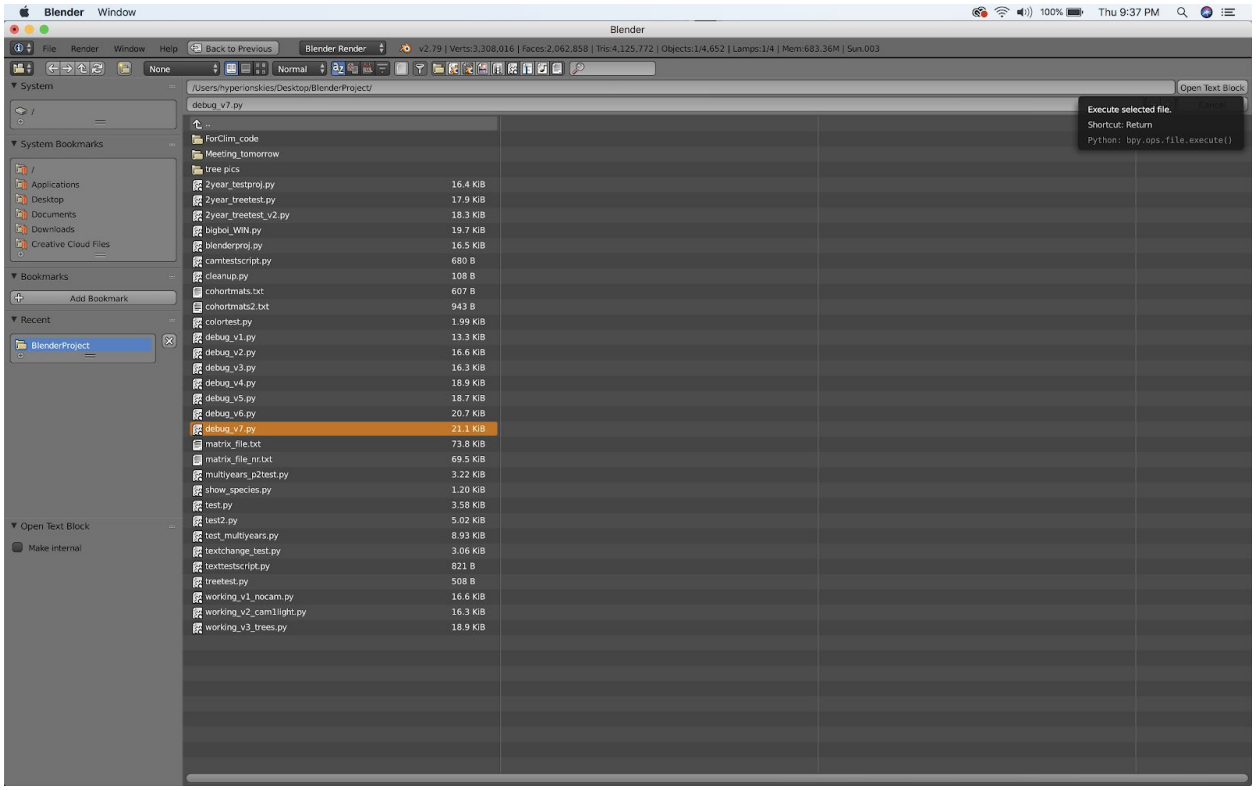
I. Open Blender.

II. Switch the top bar to Text Editor, as shown below; the circled blue i icon in the image opens the menu shown.

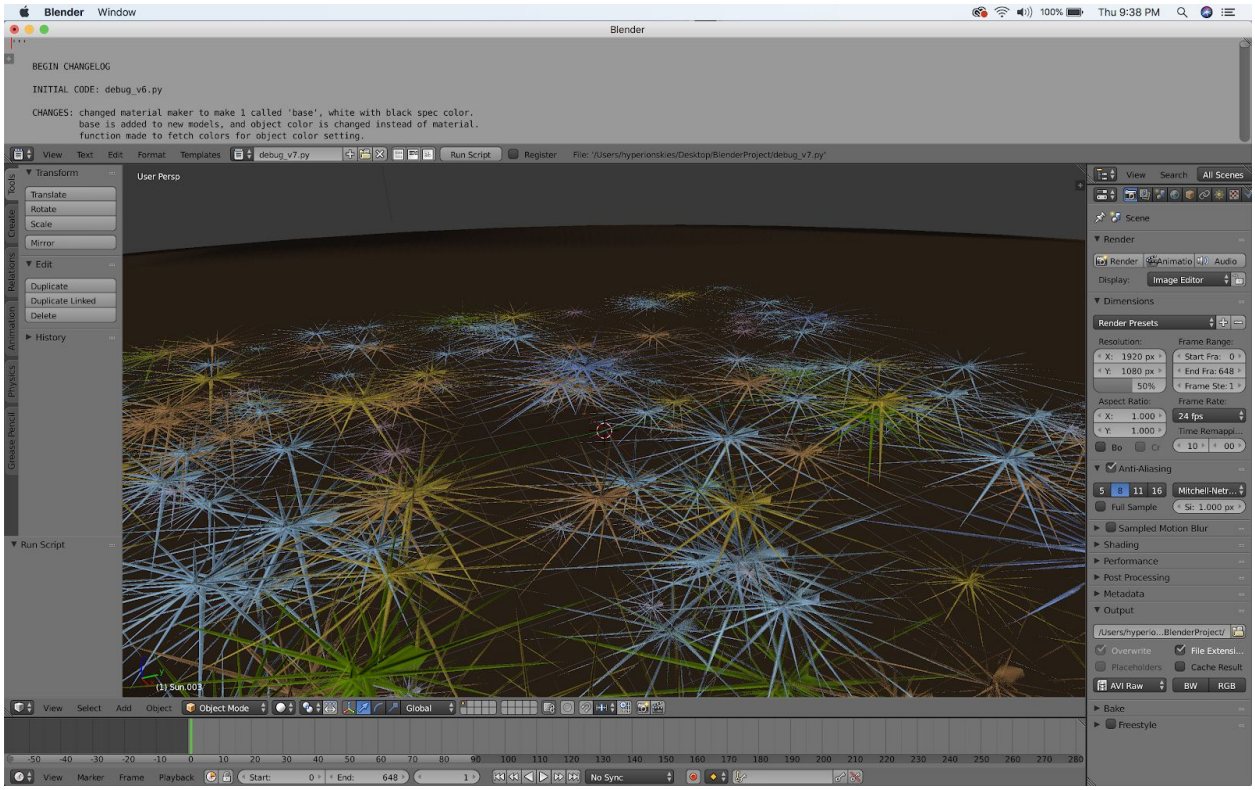


III. Click Open, navigate to your desired folder, and select animation_v1.py. Click Open Text Block.


New
Open
r Persp
Open a new text data-block.
Python: bpy.ops.text.open()

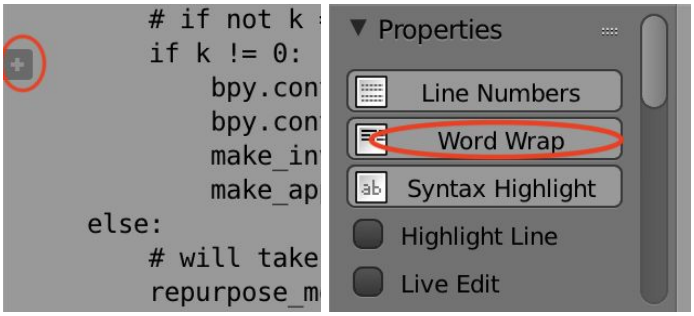

Blender Window
Thu 9:37 PM
Blender
File Render Window Help
Back to Previous
Blender Render
v2.79 | Verts:3,308,016 | Faces:2,062,858 | Tris:4,125,772 | Objects:1/4,652 | Lamps:1/4 | Mem:683.36M | Sun.003
None
Normal
System
/
System Bookmarks
/
Applications
Desktop
Documents
Downloads
Creative Cloud Files
Bookmarks
Add Bookmark
Recent
BlenderProject
Open Text Block
Make internal
/Users/hyperionskies/Desktop/BlenderProject/
debug_v7.py
Open Text Block
Execute selected file.
Shortcut: Return
Python: bpy.ops.file.execute()
..
ForClim_code
Meeting_tomorrow
tree pics
2year_testproj.py 16.4 KiB
2year_treetest.py 17.9 KiB
2year_treetest_v2.py 18.3 KiB
bigboi_WIN.py 19.7 KiB
blenderproj.py 16.5 KiB
camtestscript.py 680 B
cleanup.py 108 B
cohortmats.txt 607 B
cohortmats2.txt 943 B
colortest.py 1.99 KiB
debug_v1.py 13.3 KiB
debug_v2.py 16.6 KiB
debug_v3.py 16.3 KiB
debug_v4.py 18.9 KiB
debug_v5.py 18.7 KiB
debug_v6.py 20.7 KiB
debug_v7.py 21.1 KiB
matrix_file.txt 73.8 KiB
matrix_file_nr.txt 69.5 KiB
multiyears_p2test.py 3.22 KiB
show_species.py 1.20 KiB
test.py 3.58 KiB
test2.py 5.02 KiB
test_multiyears.py 8.93 KiB
textchange_test.py 3.06 KiB
texttestscript.py 821 B
treetest.py 508 B
working_v1_nocam.py 16.6 KiB
working_v2_cam1light.py 16.3 KiB
working_v3_trees.py 18.9 KiB

IV. Expand Text Editor window by hovering under the bar until your mouse pointer becomes arrows, and then drag downwards until you are satisfied with the panel size.

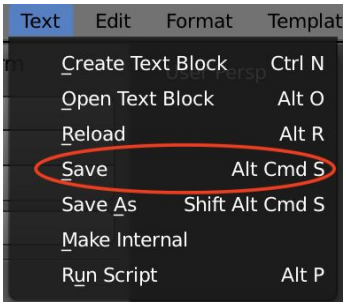


V. At this point, you will need to edit the .py file. You can either do that directly in Blender, now that it is open, or you can open the .py file in a text editor; Blender allows quick updating of open files, so either method can be used.

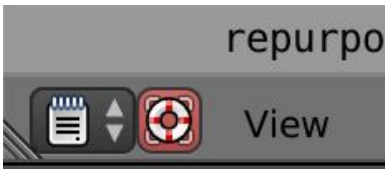
- If using Blender to edit the file, you may need to scroll side to side to see all text for some lines, or open the Properties panel and select Word Wrap.

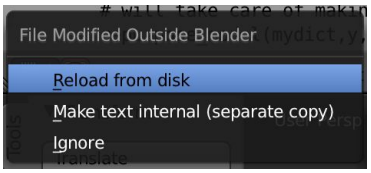


To save your changes to the original file, select Text and click Save.



- If using a text editor to edit the .py file, you may need to right click the file name and select your editor under the list of programs to open it with; some computers may try to run the file by default, which will appear to do nothing. Once you are done editing, Save your file (do not use Save As). Switch back to the Blender window, and click anywhere within the text window; a red square icon should appear. Click on it and select Reload from Disk; you will then have your updated file version in Blender.

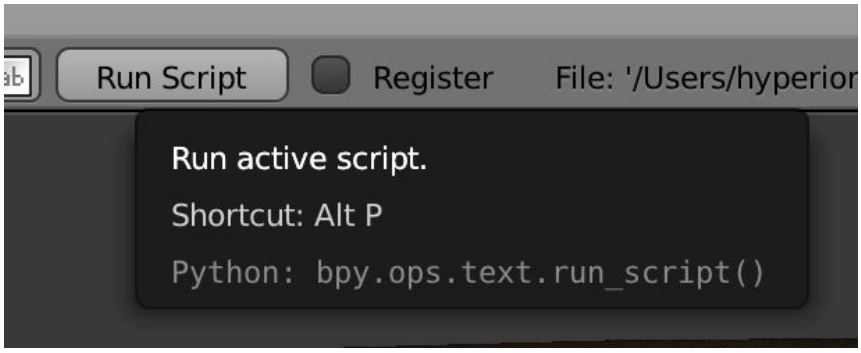




VI. Edit variables as necessary; *path*, *renderpath, colorkey*, *yearjump,* and *blendfile* should be checked. *Path* must go to the text file you created in Step 1, *renderpath* should go to the folder you want the exported video in, *colorkey* should go to a text file (which doesn't have to already exist), and *yearjump* should be edited to reflect your data; *blendfile* must be the path to Trees.blend, which should be included with the .R and .py files.

- The desired *yearjump* value can be found by opening your .txt file and looking to see by what amount the simulation years change.
- If you would like to change the model used for the trees, please see the Addendum at the end of this guide. You will also put the path to your desired .blend file as the path for *blendfile* instead of the path to Trees.blend.

VII. When you are satisfied with the variable setup, click Run Active Script.



VIII. Blender will become unresponsive. This is normal. Do not exit Blender or log out of your computer until Blender becomes responsive again. Large data sets may take several

hours to run. Should Blender become responsive, but there is not a new video file and a line in the Text Editor is highlighted in red, an error has occurred.

- Check that *path*, *renderpath*, *blendfile,* and *yearjump* are accurate; check any other variables that have been changed from the default version.
- If there are more than 24 species in your data, you may need to add additional colors to the *colors* list, under the Color List Creation heading. Please see Addendum 2 for more information on *colors*.
- If there are more than (*maxcoord**2)^2 trees in one year at a time, you may need to increase *maxcoord* (default of 50 can represent up to 10,000 trees). Please see Addendum 3 for a short script that will tell you the maximum trees per year in your dataset, along with instructions for its use.
- If you do not see red highlights but there is no movie file, change the Text Editor window to Info; if it failed, there will be a red notification, as below:

```
Python script fail, look in the console for now...
```

IX. Once Blender has completed with no errors, you have a .avi movie of your data animated. However, .avi files do not play well at large sizes; consider using Step 3 to convert your movie to .mp4 to ensure it is playable. A text file with the RGB values for each species in the animation will be in the location and under the name you specified with *colorkey*.

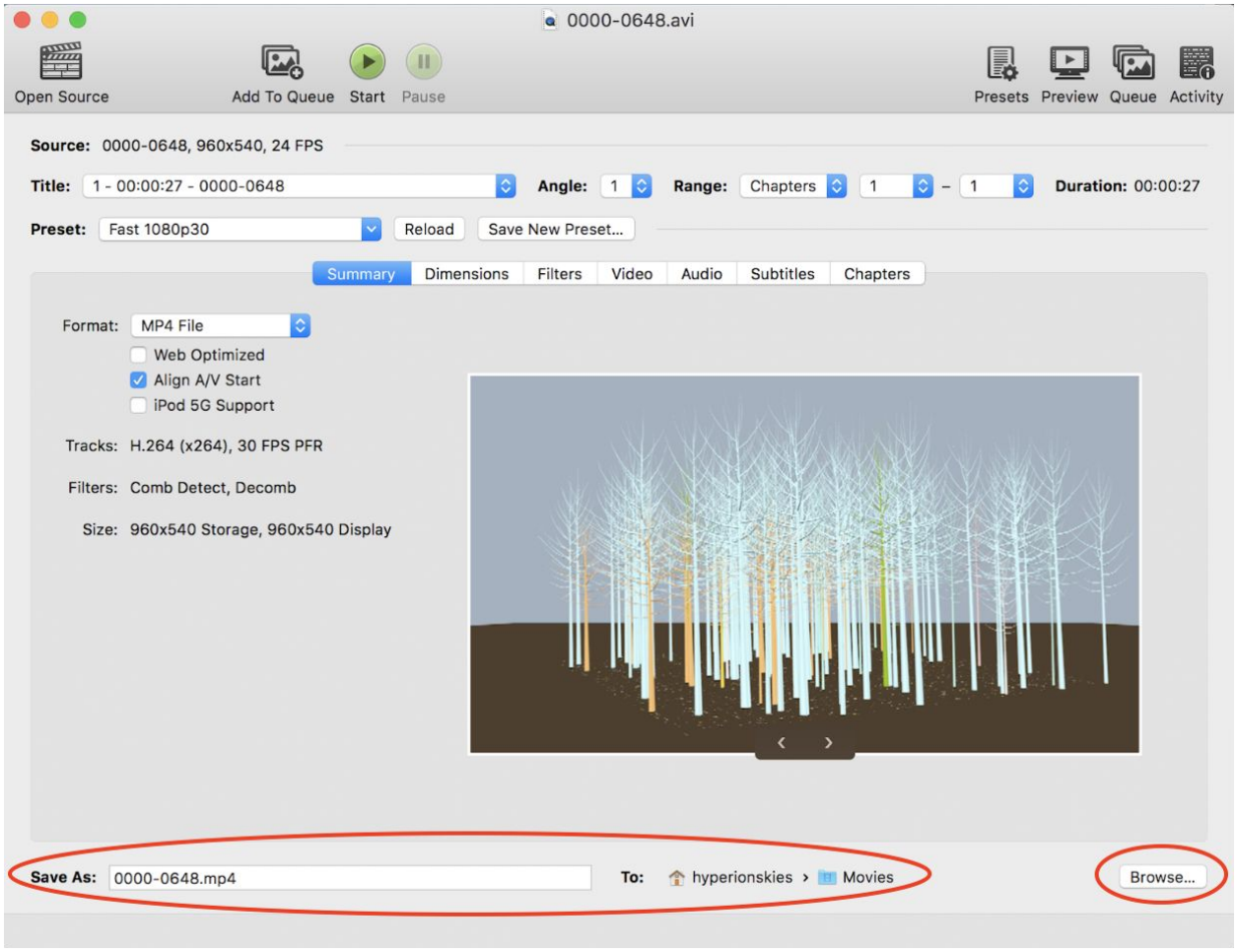
Step 3: HandBrake

I. Blender AVIs are typically very large, and .avi files larger than 1 GB may not play as-is. HandBrake can be used to convert these files to be smaller and playable.

II. Open HandBrake. It will immediately direct you to select a file to open; navigate to your Blender export file. It will be the most recent .avi file in the folder you set via *renderpath* in Step 2.

III. Make sure Format is set to mp4. At the bottom of the window, you can customize the file name, and select what folder to put the new video into via Browse. Other settings can be

ignored.



IV. Once everything is set to your satisfaction, click the green Start button. HandBrake will show a progress bar at the bottom, along with a time estimate. Leave HandBrake open and do not log out until it has completed the conversion.

V. Your new video should be in the folder you set via Browse, or the default movie folder for your computer.

Addendum

I. You may wish to change the model used in the animation. This is possible, though it must be noted that this program is designed to use a single model for all data given; it is not set up for multiple models (ex. a conifer tree and a deciduous tree in the same animation). Additionally, they must be an object, not text, a camera, or lighting. To begin with, you will need a .blend file with the model you wish to use; you can create this yourself, or look online for free-to-use licensed models.

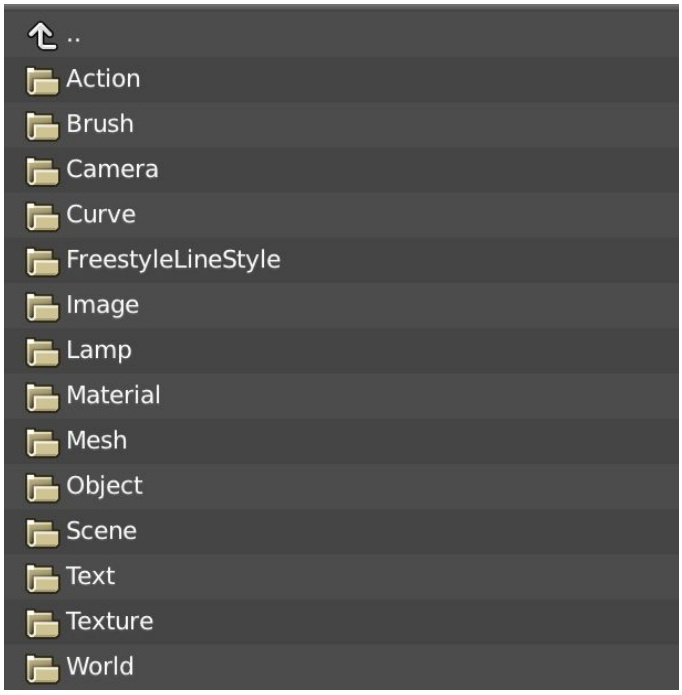
II. Once you have your .blend file, you will need to find the necessary data for the *section* and *obj* variables, under the Model data heading. To do this, open your .blend file in Blender.

III. With your file open, you should see one or more models on screen. Right click the one you would like to use, so that it is highlighted in orange. At the bottom left corner of the 3D viewport, the name of the active object (in bright orange) is shown. This will be what you place in your *obj* variable. In this case, it is “tree”.

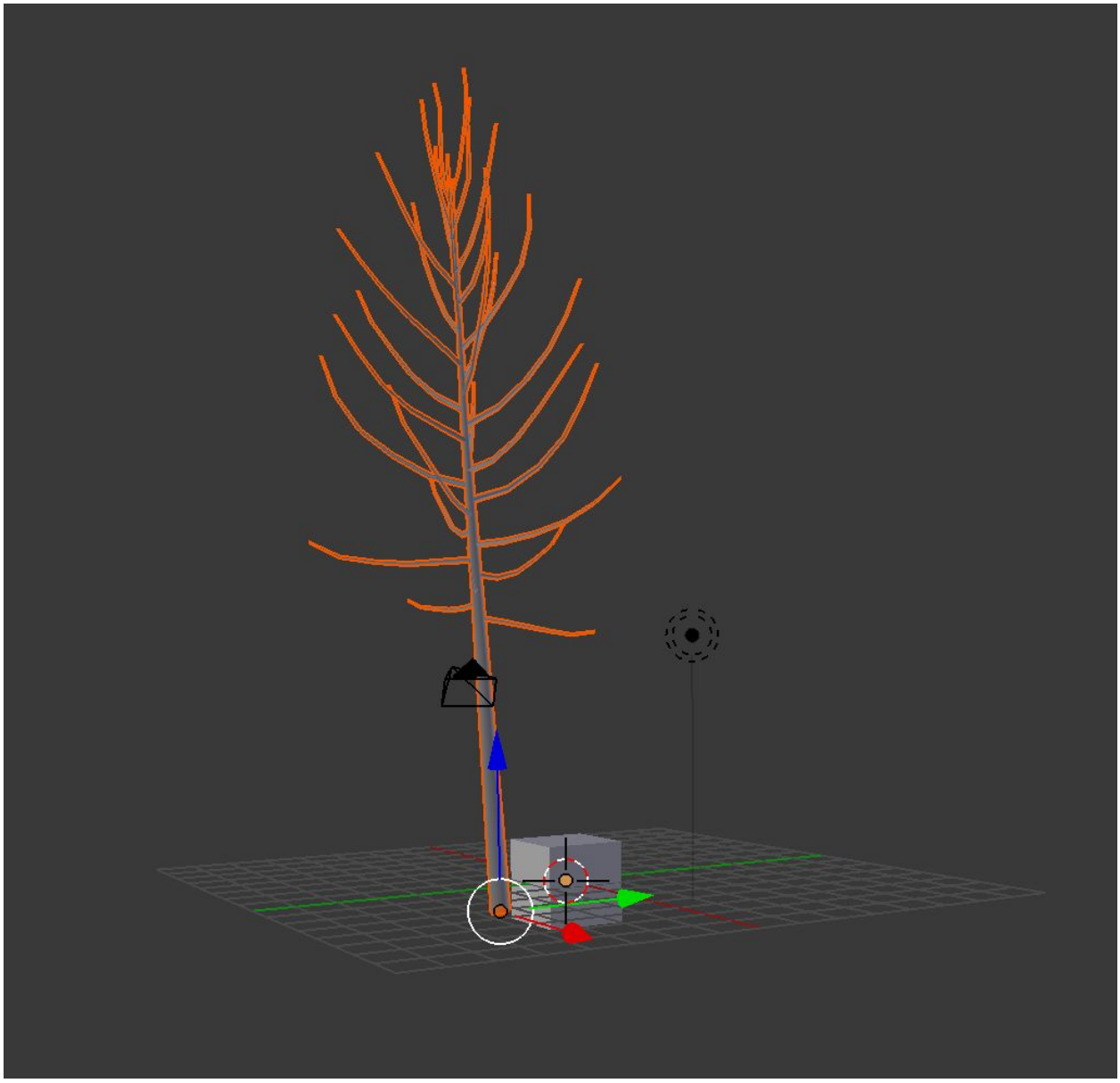
IV. The *section* variable should not need to be changed, assuming that you are using an object for your model. Should you run into an error in the script, you can open a new Blender file, and then go to File > Append, and select your file in the menu; it will bring you to a series of folders.

V. Look in these folders for the name of your model, that you found in step IV of this Addendum. Typically, it is in the Object folder, but if it is in another folder, such as Curve,

Mesh, or Text, *section* will need to be changed.



VI. Note: Text cannot be used with this program. Curve and Mesh have not been tested, and may not work. To be safe, always choose a model in the Object folder. Most grey, 3D models are designated as Objects. Additionally, names such as “Camera” or “Sun” typically belong to camera or lamp objects, which cannot be used as your models. To make sure that “Sun” or “Camera” is an object, if you are attempting to use a sun or camera model, select the name you are planning on using and click “Append from Library”. This will place it in the scene as below:



If your model is a grey, 3D object from the Object folder, you are good to go. If, instead, it is primarily black lines, as also shown above, you have selected a camera or lamp object; this is not the object you want. Keep looking until you find the appropriate name.

Addendum 2

There are 24 colors provided by the script to map to individual tree species. These are provided in a list called *colors*, where each color is listed as three numbers in brackets (ex. [1.0,1.0,1.0]).

You can add more colors to the list by simply adding more sets of three numbers in brackets, with commas between each set as per the rest of the list, but there are several considerations.

I. Blender uses a variation on RGB. You can find the standard RGB values for a particular color using tools, such as w3school's easy to use widget here; you will note that the values for R, G, and B (red, green, and blue) are between 0 and 255. Blender uses a scale of 0 to 1.0 instead; and so you will need to take the standard values and divide them by 255 to see what you should be typing into the .py file. Note that these numbers tend to be very long decimals, but you can shorten them to the third decimal place.

   - Example 1: Bright red, with RBG values (255,0,0), would be typed in as [1.0,0,0]. 255/255 = 1.0, 0/255 = 0.

   - Example 2: A medium grey with values (144,144,144), would be typed in as [0.565,0.565,0.565]. 144/255 is 0.56470…., but we can round it to 0.565.

II. Colors will need to be selected to ensure that trees are visually distinct. There are already 24 colors provided, over a wide range of colors; pastel, medium, and dark variations of grey, red, pink, blue, teal, green, yellow, and orange were selected, to create a wide range of colors, but this means it may be difficult to find distinct colors from these. You may find that creating your own list is easier; either by hand, or using a site such as this one. (To use the linked site, click the Refine tab, adjust Threshold in the top right corner to roughly the amount of colors you need (it will not be precise), and then go to the Results tab to see a list of swatches and values.)

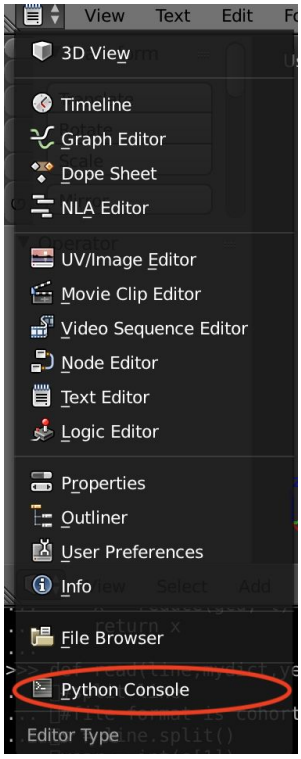
III. The following is a listing of the colors included by default in the .py script. Note the swatches may be off due to limitations of Google Docs (it does not represent darks or pastels well). The swatches are just to give a basic idea of what is represented.

Color swatch	RGB	Blender entry
	rgb(255,255,255)	[1.0,1.0,1.0]
	rgb(161,161,161)	[0.633,0.633,0.633]
	rgb(0,0,0)	[0.0,0.0,0.0]
	rgb(42,2,42)	[0.168,0.008,0.168]
	rgb(169,42,169)	[0.665,0.0168,0.665]
	rgb(178,60,178)	[0.701,0.235,0.701]
	rgb(56,2,2)	[0.220,0.008,0.008]

	rgb(218,11,11)	[0.855,0.044,0.044]
	rgb(207,66,66)	[0.815,0.262,0.262]
	rgb(6,2,53)	[0.023,0.010,0.209]
	rgb(15,6,162)	[0.060,0.023,0.638]
	rgb(60,45,184)	[0.235,0.175,0.723]
	rgb(2,40,42)	[0.008,0.156,0.168]
	rgb(5,159,171)	[0.022,0.624,0.672]
	rgb(51,199,207)	[0.200,0.780,0.814]
	rgb(2,42,3)	[0.008,0.168,0.010]
	rgb(6,175,8)	[0.024,0.687,0.032]
	rgb(48,171,171)	[0.188,0.672,0.672]
	rgb(43,47,2)	[0.171,0.185,0.007]
	rgb(178,194,4)	[0.701,0.761,0.017]
	rgb(178,184,75)	[0.701,0.723,0.292]
	rgb(42,20,2)	[0.168,0.080,0.007]
	rgb(207,88,3)	[0.815,0.347,0.010]
	rgb(166,97,23)	[0.651,0.381,0.093]

Addendum 3

I. Select the Info or Text Editor icon, and from the dropdown menu select Python Console.



II. Copy the text from the table below and paste it into a text editor. Change *path* to be the path to the .txt file created with the R script.

```
from math import cos, sin, pi, gcd
from functools import reduce

path = '/Users/hyperionskies/Desktop/BlenderProject/matrix_file_nr.txt'

def find_gcd(l):
    x = reduce(gcd, l)
    return x

def read(line,mydict):
        #split line
        #file format is cohortID, year, species, trees, diameter, height
        s = line.split()
        year = int(s[1])
        trees = int(s[3])
        #assign cohort to mydict
        if not year in mydict:
                mydict[year] = 0
        mydict[year] = mydict[year] + trees

mydict = {}
```

```
db_file = open(path,'r')
test = db_file.readline().rstrip()
while not test is '':
        read(test,mydict)
        test = db_file.readline().rstrip()

treelist = []
for y in mydict:
        treelist.append(mydict[y])

maxtrees = max(treelist)

print(maxtrees)
```

III. Copy all of the text from your text editor (so that it has the correct path) and paste it into the Python Console, then hit enter.

IV. The number written in blue is the maximum number of trees in a single year of your dataset. The .py script will require that this number is lower than (*maxcoord**2)^2, and so you will need to adjust *maxcoord* if the number is too high.
    - If you take the number you got from the console, divide it by 4, and then square root it, the result is the smallest number that *maxcoord* can be set to for the program to work. It is recommended to use that number rounded up to the next highest whole number as your value for *maxcoord*, unless it is smaller than the default of 50.
    - If *maxcoord* is changed, the .py script will automatically adjust scaling factors to account for this difference, in an attempt to keep all trees visible in the animation. If you find that some trees are cut out (i.e. the rotation of the camera does not catch some trees at all), you may need to adjust *scale* or *cscale*. Do not make *cradius* any higher, or background trees may not show in the animation.

Default Settings

The following is the contents of the default Global Variables heading in animation_v1.1.py. Should you need to reference or reset the script to its original state, the entirety of the original settings is in text below for your convenience.

```
########################################
#---        Global Variables        ---#
########################################

#--- Data file
```

```
# mac directory path sample
path = '/Users/hyperionskies/Desktop/BlenderProject/matrix_file_nr.txt'
# windows directory path sample
# path = 'C:/Users/zc600012/Desktop/Cohort_Database/matrix_file_big.txt'


#--- Outputs

#--- Folder
# mac directory path sample
renderpath = '/Users/hyperionskies/Desktop/BlenderProject/'
# windows directory path sample
# renderpath = 'C:/Users/zc600012/Desktop/Cohort_Database/'

#--- Color key
# mac directory path sample
colorkey = '/Users/hyperionskies/Desktop/BlenderProject/colorkey.txt'
# windows directory path sample
# colorkey = 'C:/Users/zc600012/Desktop/Cohort_Database/cohortmats.txt'


#--- year change

yearjump = 25


#--- Model file

# Should point to Trees.blend, unless another model is desired;
# if not using Trees.blend, change 'section' and 'obj' in Model data.
# see manual for more info.

# mac directory path sample
blendfile = '/Users/hyperionskies/Desktop/BlenderProject/Trees.blend'
# windows directory path sample
# blendfile = 'C:/Users/zc600012/Desktop/Cohort_Database/Trees.blend'


#--- Keyframing

# NOTE: Script does not affect Blender's fps; default is 24.

# keyincr - still = number of keyframes for change between years
# num keyframes still (x, for k+x in loops)
still = 24
# num keyframes between years
keyincr = 72
```

```
#--- Coordinates

# coordinates will be between -(maxcoord-1) and maxcoord
maxcoord = 50
# cscale automatically adjusts sizes of model and coordinate "squares"
cscale = maxcoord/50

#--- Scaling

# Tree scaling
scale = (0.0254)*(16/28.3)*0.3*cscale
# Additional scaling for specific tree model
xmod = 12
#--- xmod used for radius value instead of separate x/y values
#--- z scale modifier (full precision: 0.310608)
zmod = 0.311


#--- Camera/Text orbit

# camera orbit radius
cradius = 55.0
# text orbit radius
tradius = 25.0

# NOTE: to change speed of orbit, adjust numk and/or kf
# number of locations on camera orbit to keyframe (will be evenly distr)
numk = 60
# frequency of camera keyframing
kf = 12


#--- Background

# radius of circle floor
floorscale = 60
# RGB color for floor
floorcolor = (0.046,0.028,0.015)
# RGB color for sky
skycolor = (0.315,0.390,0.477)


#--- Model data

# section and obj should not need to be changed unless dealing with a different
.blend
```

```
section = '\\Object\\'
obj = 'tree'

filepath  = blendfile + section + obj
directory = blendfile + section
filename  = obj
```