OptoGraphs: Creating gif images for Optomotor and Optokinetic responses in larval fish.

Erin Frolli • Fuiman Lab

May 08, 2017

Contents

1. Introduction

Two R functions were designed to create two simple assays for testing the Optomotor and optokinetic response of larval fish. This is short vignette to describe the steps of two methods to create a **.gif** image in R. The first method, described in the Circular Assay, is by creating several **.png** images and turning them into **.gif** or **.avi** files using ImageJ. The second method, described in Stripe Assay, utilizes ImageMagick package by creating a **.gif** through the command line. Please note that each assay has code for both types of saving methods.

2. Circular Assay

In this section we will describe how to make an animated .gif for the Circular Assay using both R and ImageJ. We will start by opening R. Then load OptoCirclesF() function. Don't forget to use the correct file path if it is different from your working directory.

```
> # Note: write the file path to the folder that you have the R file saved in.
> source(' C:/Users/Desktop/OptoCirclesF.R')
```

Next we will need to set our working directory to a specific folder or folders.

```
> #Set your working directory to working folder
> setwd('C:/Users/Desktop)
>
> # Create a destination folder
> dir.create("ClockWiseCircles") # Name this folder whatever you like
> dir.create("CounterClockWiseCircles") # Name this folder whatever you like


> # Note you can skip this step if you have already created destination folders
> # And just move onto the next line of code.
> # Then set our directory to the new folder
> setwd('C:/Users/Desktop/ClockWiseCircles)
```

Now you are ready to run the function. This will create a series of 36 images in that designated folder that will identical but slightly different from each other in angle orientation.

```
> OptoCirclesF(20) # Clockwise
```

This creates 36 images that have 20 wedges of equal size and when you collate the images they will rotate clockwise. Note: You can change the number of wedges with any number, but if you chose an odd number it will make it an even number in order to maintain the color pattern.

Repeat the above steps if you want to create counter clockwise images also.

```
> # Then set our directory to the new folder
> setwd('C:/Users/Desktop/CounterClockWiseCircles)
> OptoCirclesF(20,Reverse = TRUE) # Counter Clockwise
```

** Note you can also change the colors of the graph in order to test color contrast.

```
> OptoCirclesF(20, ColorSr= c("black","red"),Reverse = TRUE) # Counter Clockwise
```

Once you are happy with the images you have created its time to make the .gif or a video. We will do this in ImageJ. So open ImageJ. Then to upload the images we just created:

Go to File → Import → Image Sequence…

Navigate to the folder that has all the images in it. Then click on the first image in the sequence. It will have a number that looks like this: 0001plot.png

Next a popup window will open. Make sure the following are done:

Number of images: 36
Starting image: 1
Increment: 1
Scale images: 100%
Convert to RGB: is checked
Sort names numerically: is checked

Once all of the above are satisfied press the OK button. This should upload your images into a single window. You can press the play button to see your images in action as a sequence. To adjust the speed of the sequence:

Go to Images → Stacks → Tools → Animation options…

A new popup window will open. You can change the speed in the Speed box. This is number of frames per second. So the smaller the number the slower it goes and visa versa. Then press OK.

Once you are satisfied with your image you can save it in two formats. First format is a video format.

Go to File → Save As → AVI…

There will be a pop up box where you can adjust the compression and frame rate.

Compression: JPEG
Frame Rate: (Should be the same as your speed)

Then press OK. Then the save window will pop up. Rename and save your movie.

To save your images as a .gif file. First you need to change the image format:

Go to Image → Type.. → 8-bit color

Then save the file:

Go to File → Save As → Gif…

Then press OK. Then the save window will pop up. Rename and save your gif.

2. Stripe Assay

In this section we will describe how to make an animated .gif for the Stripe Assay using just R with the help of ImageMagick an animation support package. We will start by opening R. If you don't already have ImageMagick you need to install it. This can be done in R using the package installr:

```
# Install installr pakage & load it
> install.packages("installr")
> require(installr)
# Then install ImageMagick
> install.ImageMagick()
```

Then open the OptoStripesF() function in the R console. Don't forget to use the correct file path if it is different from your working directory.

```
> file.edit('C:/Users/Desktop/OptoStripesF.R')
```

Then you need to find where ImageMagick downloaded in your file directory. Once you find it you need to update line 77 to match your file directory for the convert function. Then edit this line of code to match your file directory. Note only replace the area in yellow.

```
# Replace the yellow portion of line 77 to match your computer's directory
```

```
GifPath = paste('"C:/Program Files/ImageMagick-7.0.5-Q16/convert.exe" -delay
',Speed, ' *.png ',R,'Stripes',BarWidth,'_Speed',Speed,'.gif',sep = "")
```

Once you have ImageMagic all set up you need to load your OptoStripesF().

```
> # Note: write the file path to the folder that you have the R file saved in.
> source('C:/Users/Desktop/OptoStripesF.R')
```

Next we will need to set our working directory to a specific folder or folders.

```
> #Set your working directory to working folder
> setwd('C:/Users/Desktop)
>
> # Create a destination folder
> dir.create("Stripes") # Name this folder whatever you like
>
> # Note you can skip this step if you have already created destination folders
> # And just move onto the next line of code.
> # Then set our directory to the new folder
> setwd('Stripes')
```

Now you are ready to run the function. This will create an animated .gif in that designated folder that will play on loop a series of strips moving across the image.

```
> # You can have the image movie either left or right
> # Right handed .gif
> OptoStripesF(20,10)
> # Note the first variable is the bar width, the second variable is speed, and
the third is reverse True/False note the default is FALSE.

> # Left handed .gif
> OptoStripesF(20,10,Reverse = TRUE)
> # Note the first variable is the bar width, the second variable is image speed,
and the third is reverse True/False.
```

Note: You can change the size of the stripes but the values have to be 0> X <=50. Also the speed variable the lower the value the faster the gift will animate. You can also change the colors of the graph in order to test color contrast.

```
> OptoStripesF(20,10,ColorSr= c("black","red"),Reverse = TRUE) # Counter
Clockwise
```