

Package: camcorder (via r-universe)

September 15, 2026

Type Package

Title Record Your Plot History

Version 0.1.0.9000

Description Record and generate a 'gif' of your 'R' sessions plots.

When creating a visualization, there is inevitably iteration and refinement that occurs. Automatically save the plots made to a specified directory, previewing them as they would be saved. Then combine all plots generated into a 'gif' to show the plot refinement over time.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.2.1

URL <https://thebioengineer.github.io/camcorder/>,

<https://github.com/thebioengineer/camcorder>

BugReports <https://github.com/thebioengineer/camcorder/issues>

Imports ggplot2, gifski, tools, magick, rsvg, jsonlite, rlang, grDevices, svglite

Suggests knitr, rmarkdown, markdown, ragg, grid, covr, testthat (>= 3.0.0), patchwork, withr, dplyr, systemfonts, scales, extrafont

VignetteBuilder knitr

Config/testthat/edition 3

Config/pak/sysreqs libfontconfig1-dev libfreetype6-dev libfribidi-dev libharfbuzz-dev libmagick++-dev gsfonts libicu-dev libpng-dev librsvg2-dev libssl-dev libclang-dev

Repository <https://thebioengineer.r-universe.dev>

Date/Publication 2026-09-15 21:30:30 UTC

RemoteUrl <https://github.com/thebioengineer/camcorder>

RemoteRef HEAD

RemoteSha fd1938f0b6ba0c265aa6fe7093a7cf3d619df5ef

Contents

gg_record	2
record_polaroid	4
Index	6

gg_record	<i>Record and generate plot histories</i>
-----------	---

Description

Record plots created over time and generate a GIF of the plots made in the 'R' session. Overrides the print methods for ggplot and patchwork objects from the 'ggplot2' and 'patchwork' packages respectively.

resize the film for recording, reprints and saves last plot

Stop recording images with camcorder.

Usage

```
gg_record(  
  dir = NULL,  
  device = c("png", "pdf", "jpeg", "bmp", "tiff", "emf", "svg", "eps", "ps"),  
  scale = 1,  
  width = NA,  
  height = NA,  
  units = c("in", "cm", "mm", "px"),  
  dpi = 300,  
  limitsize = TRUE,  
  device_ext = NULL,  
  bg = NULL  
)  
  
gg_playback(  
  name = NULL,  
  first_image_duration = 16,  
  last_image_duration = 20,  
  frame_duration = 0.25,  
  loop = TRUE,  
  image_resize = 600,  
  background = "black",  
  width = NULL,  
  height = NULL,  
  progress = interactive(),  
  playback = TRUE,  
  stoprecording = FALSE,  
  last_as_first = TRUE,
```

```

    ...
  )

  gg_resize_film(height = NA, width = NA, units = NA, dpi = NA)

  gg_stop_recording()

```

Arguments

<code>dir</code>	directory to save the intermediate plots in. Defaults to a temporary directory
<code>device</code>	Device to use. Can either be a device function (e.g. <code>png()</code>), or one of "png", "pdf", "jpeg", "bmp", "tiff", "emf", "svg", "eps", "ps".
<code>scale</code>	Multiplicative scaling factor.
<code>width, height, units</code>	Plot size in units ("in", "cm", "mm", or "px"). If not supplied, uses the size of current graphics device.
<code>dpi</code>	Plot resolution. Also accepts a string input: "retina" (320), "print" (300), or "screen" (72). Applies only to raster output types.
<code>limitsize</code>	When TRUE (the default), <code>ggsave()</code> will not save images larger than 50x50 inches, to prevent the common error of specifying dimensions in pixels.
<code>device_ext</code>	file extension to use for images created. Does not usually need to be populated manually.
<code>bg</code>	Background colour. If NULL, uses the <code>plot.background</code> fill value from the plot theme.
<code>name</code>	name of gif.
<code>first_image_duration</code>	n units of <code>frame_duration</code> to show the first image for.
<code>last_image_duration</code>	n units of <code>frame_duration</code> to show the last image for.
<code>frame_duration</code>	n seconds each plot should be shown.
<code>loop</code>	if the gif should be repeated. Set to FALSE to only play once, or a number to indicate how many times to repeat after the first.
<code>image_resize</code>	size to rescale images to in pixels.
<code>background</code>	color to set the background of the gif. A valid color string such as "navyblue" or "#000080". Use "none" for transparency. Does not impact the background of images.
<code>progress</code>	print some verbose status output
<code>autoplay</code>	Boolean, should the recording start playing after it is turned into a gif? defaults to TRUE.
<code>stoprecording</code>	Boolean, should the plots stop being recorded? defaults to TRUE.
<code>last_as_first</code>	Should the last plot be displayed at the beginning too?
<code>...</code>	Other arguments passed on to the graphics device function, as specified by device.

Value

Used initialize recording, nothing returned

Returns nothing. Used to generate the gif.

Returns the last plot generated, resized to new dimensions

Returns nothing. used for side effect.

Functions

- gg_playback():
- gg_resize_film():
- gg_stop_recording():

Examples

```
if(require(ggplot2) & interactive()){
  gg_record(dir = file.path(tempdir(),"recording"))
  ggplot(data.frame(x = 1, y = 1), aes(x=x, y=y)) + geom_point() + ylim(0,4)
  ggplot(data.frame(x = 1, y = 2), aes(x=x, y=y)) + geom_point() + ylim(0,4)

  ## resize canvas of the last plot
  gg_resize_film(height = 10, width = 5, dpi = 350)

  ggplot(data.frame(x = 1, y = 3), aes(x=x, y=y)) + geom_point() + ylim(0,4)

  gg_playback(tempfile(fileext = ".gif"))
}
```

record_polaroid

Record Plots - generic

Description

For plot types that don't have a special print method, use this function to capture what has been printed to the current graphics device and save it using the current camcorder settings

Usage

```
record_polaroid()
```

Value

No return value. Used for the side effect of capturing the current graphics device and saving it to the set directory from gg_record.

Examples

```
library(grid)

gg_record(device = "png", width = 10, height = 8, units = "in", dpi = 320)

## make a plot using grobs
grid.draw(rectGrob(width = 2, height = 2, gp = gpar(fill = "green")))
grid.draw(textGrob("Hello world"))

record_polaroid()

gg_stop_recording()
```

Index

`gg_playback (gg_record)`, [2](#)
`gg_record`, [2](#)
`gg_resize_film (gg_record)`, [2](#)
`gg_stop_recording (gg_record)`, [2](#)
`record_polaroid`, [4](#)