

Package: impac (via r-universe)

August 26, 2024

Title Make packed image mosaics

Version 0.1.0.9000

Description impac takes a list of images or a function that generates images and packs them onto a canvas using a greedy algorithm, while respecting transparency.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.2

Imports imager, magrittr, progress, rlang (>= 0.1.2)

Suggests roxygen2, rmarkdown, Rvcg, rgl, rphylopic (>= 1.0.0), testthat (>= 3.0.0), rtweet

Config/testthat/edition 3

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

RemoteUrl <https://github.com/rdinnager/impac>

RemoteRef HEAD

RemoteSha f729be6f823c1502f8a1811b9b41470e692620dc

Contents

impac	2
impac_clear_cache	3
impac_recover	4
impac_resume	4
impac_write	5

Index	6
--------------	----------

 impac

Make a packed image mosaic

Description

A simple greedy algorithm tries to pack as many images into a larger image as possible, taking into account transparency, if available (recommended).

Usage

```
impac(
  im,
  width = 1024,
  height = 800,
  mask = NULL,
  weights = NULL,
  preferred = NULL,
  max_num_tries = 100,
  scales = c(rep(0.5, 2), rep(0.25, 4), rep(0.15, 8)),
  scale_fun = function(s, i, c) { if (c < (i * 0.5)) { mscale <- min(s)
    c(s, rep(mscale/2, floor(1/mscale))) } else { scales } },
  max_images = 1000,
  min_scale = 0.05,
  bg = "transparent",
  show_every = 25,
  progress = TRUE,
  start_image = NULL,
  ...
)
```

Arguments

im	Can be either a character vector of image file names (format must be compatible with <code>imager::load.image()</code>), a list of <code>imager::cimg</code> objects, or a function that generates an image when evaluated. The function can take a single argument, which is the current iteration of the packing algorithm. Can also be specified as an <code>rlang</code> style lambda syntax (see <code>rlang::as_function()</code>).
width	Width in pixels of produced image
height	Height in pixels of produced image
mask	An optional masking image.
weights	Vector of Weights to apply to each image. Higher weighted images will be packed first and so will tend to be larger. This vector will be recycled.
preferred	An alternate way to specify images to pack first, as a character vector of names or file names (only works if <code>im</code> is a vector of image file name or a list of <code>imager::cimg</code> objects).

max_num_tries	Maximum number of times to try packing an image onto the canvas before giving up.
scales	A vector of starting scaling factors to randomly choose from for each image.
scale_fun	An function that takes three arguments, which correspond to the current vector of scaling factors, the current iteration of the algorithm, and the count of the number of packed images so far, respectively (e.g. $f(s, i, c)$), and returns a new vector of scaling factors to use.
max_images	The maximum number of images to pack before stopping.
min_scale	The minimum scale factor to use. If the algorithm generates a scale factor this small (via <code>scale_fun</code>), packing will stop.
bg	The background colour for the canvas, default: "transparent"
show_every	Show the intermediate packed image after every <code>show_every</code> images are packed. Set to 0 to not show intermediates.
progress	Should progress be printed as the algorithm runs?
start_image	An optional image to start the packing with. If not NULL, the width and height arguments will be ignored and the dimensions of the starting image used instead. Can be an <code>imager::cimg</code> object, a path to an image in png or jpg format or an <code>impac</code> object.
...	Further arguments passed on the <code>im</code> , if it is function.

Value

A packed image mosaic, as a `imager::cimg` object.

Examples

```
plot(
  impac(
    function(i) imager::draw_circle(
      imager::imfill(500, 500, val = c(0, 0, 0, 0)),
      250, 250, radius = runif(1, 150, 250),
      color = matrix(grDevices::col2rgb(sample(grDevices::rainbow(100), 1), alpha = TRUE), nrow = 1)
    ),
    width = 400, height = 400,
    max_images = 10, bg = "white"
  )$image
)
```

`impac_clear_cache` *Clear any cached impac objects*

Description

Clear any cached `impac` objects

Usage

```
impac_clear_cache()
```

Value

No return value

Examples

```
impac_clear_cache()
```

impac_recover	<i>Function to try and rescue</i>
---------------	-----------------------------------

Description

Function to try and rescue

Usage

```
impac_recover()
```

impac_resume	<i>Function to resume an image packing where it left off</i>
--------------	--

Description

Function to resume an image packing where it left off

Usage

```
impac_resume(x = NULL, ...)
```

Arguments

x	An impac object created from a previous run of impac. Can also be left blank in which case this function attempts to recover the latest impac run from the cache (see impac_recover() for details).
...	Other arguments to be passed to impac() . By default, original arguments from the original call used to make x will be used. Passing an argument here will override the original arguments.

Value

An impac object

`impac_write`

Save packed image mosaic

Description

Save packed image mosaic

Usage

```
impac_write(x, file, quality = 0.7)
```

Arguments

quality

Index

`imager::cimg`, 2, 3
`imager::load.image()`, 2
`impac`, 2
`impac()`, 4
`impac_clear_cache`, 3
`impac_recover`, 4
`impac_recover()`, 4
`impac_resume`, 4
`impac_write`, 5

`rlang::as_function()`, 2