

# Package ‘imageviewer’

February 18, 2019

**Title** Simple 'htmlwidgets' Image Viewer with WebGL Brightness/Contrast

**Version** 0.1.0

**Author** Iakov Pustilnik [aut, cre],  
Denis Rastegaev [aut]

**Maintainer** Iakov Pustilnik <xyapus@gmail.com>

**URL** <https://github.com/yapus/imageviewer>

**BugReports** <https://github.com/yapus/imageviewer/issues>

**Description** Display a 2D-matrix data as a interactive zoomable gray-scale image viewer, providing tools for manual data inspection. The viewer window shows cursor guiding lines and a corresponding data slices for both axes at the current cursor position. A tool-bar allows adjusting image display brightness/contrast through WebGL filters and performing basic high-pass/low-pass filtering.

**Depends** R (>= 3.4), htmlwidgets

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.1.1

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2019-02-18 14:50:40 UTC

## R topics documented:

imageviewer . . . . .	2
imageviewer-shiny . . . . .	2

<b>Index</b>	<b>4</b>
--------------	----------

imageviewer

*imageviewer*

---

**Description**

Simple htmlwidgets matrix viewer with WebGL brightness/contrast

**Usage**

```
imageviewer(data, width = NULL, height = NULL, elementId = NULL,  
            options = list())
```

**Arguments**

data	A matrix
width, height	matrix dimensions
elementId	HTML element Id
options	list of other options (passed through to JS code)

**Value**

Plot matrix in html widget

**Examples**

```
# Create matrix  
m <- matrix(rnorm(512 * 512, mean = 100, sd = 10), 512, 512)  
  
# Plot  
imageviewer(m)
```

---

imageviewer-shiny*Shiny bindings for imageviewer*

---

**Description**

Output and render functions for using imageviewer within Shiny applications and interactive Rmd documents.

**Usage**

```
imageviewerOutput(outputId, width = "100%", height = "400px")  
  
renderImageviewer(expr, env = parent.frame(), quoted = FALSE)
```

**Arguments**

outputId	output variable to read from
width, height	Must be a valid CSS unit (like '100%', '400px', 'auto') or a number, which will be coerced to a string and have 'px' appended.
expr	An expression that generates a imageviewer
env	The environment in which to evaluate expr.
quoted	Is expr a quoted expression (with quote())? This is useful if you want to save an expression in a variable.

# Index

`imageviewer`, [2](#)  
`imageviewer-shiny`, [2](#)  
`imageviewerOutput (imageviewer-shiny)`, [2](#)  
`renderImageviewer (imageviewer-shiny)`, [2](#)