

# Package: asciify (via r-universe)

August 29, 2024

**Type** Package

**Title** Make ASCII Art From Images

**Version** 0.1.0

**Description** Takes an arbitrary image as input and constructs an  
text-based approximation to the image using the imager package.

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**URL** <https://github.com/djnavarro/asciify>

**BugReports** <https://github.com/djnavarro/asciify/issues>

**RoxygenNote** 6.1.1

**Imports** imager, dplyr, purrr, ggplot2, tibble, stringr

**Repository** <https://djnavarro.r-universe.dev>

**RemoteUrl** <https://github.com/djnavarro/asciify>

**RemoteRef** HEAD

**RemoteSha** 469b1fbe77aac4f4e4b157feef129a6de5975491

## Contents

ascii_data	2
ascii_grid	2
ascii_map	3
ascii_plot	4
ascii_rain	5
ascii_text	5

<b>Index</b>	<b>7</b>
--------------	----------

---

ascii_data	<i>Specifies path to one of the data files in the package</i>
------------	---

---

**Description**

Specifies path to one of the data files in the package

**Usage**

```
ascii_data(file)
```

**Arguments**

file	Name of file as a character string
------	------------------------------------

**Details**

This is a convenience function that returns the path to one of the external data files bundled in the asciify package. There are only four such files: "bayes.png", "unicorn.png", "heart.jpg", "rain.html"

**Value**

Path to file as a character string

**Examples**

```
ascii_data("bayes.png")
ascii_data("unicorn.png")
ascii_data("heart.jpg")
ascii_data("rain.html")
```

---

ascii_grid	<i>Converts an ASCII map to a matrix</i>
------------	--

---

**Description**

Converts an ASCII map to a matrix

**Usage**

```
ascii_grid(image_map)
```

**Arguments**

image_map	Map from ascii_map
-----------	--------------------

**Value**

A matrix

**Examples**

```
bayes_img <- ascii_data("bayes.png") # path to the bayes image
bayes_map <- ascii_map(bayes_img)     # construct ASCII map
bayes_grid <- ascii_grid(bayes_map)   # make grid
```

---

ascii_map	<i>Creates ASCII art from an image</i>
-----------	--

---

**Description**

Creates ASCII art from an image

**Usage**

```
ascii_map(file, alphabet = letters, rescale = NULL, threshold = 0.5)
```

**Arguments**

file	A character string specifying the path to the file
alphabet	A character vector that lists the set of characters to use
rescale	Scale to resize image to (if NULL, sets maximum size of 100x100)
threshold	Lightness value at which to truncate

**Details**

This is the workhorse function for the package. The user specifies the path to the image file as a character vector, and the function returns a tibble specifying the map. You can customise the set of characters used to render the image by changing the alphabet argument. The amount of whitespace in the ASCII image depends on the threshold parameter, which specifies the maximum brightness (in greyscale terms, from 0 to 1) at which the relevant cell in the output will be mapped to a character. Pixels in the image brighter than the threshold are automatically mapped to whitespace. Finally, you can "rescale" the input image. By default, images are rescaled so that the largest dimension is 100 pixels, but you can choose any rescaling factor you want. If your original image is 600x400 you could specify rescale = .1 which would result in a character map that is 60x40 characters in size.

**Value**

The function returns a tibble with three variables. The x and y variables specify co-ordinates on the grid, and the label variable specifies the character that should be shown at that point. Whitespace characters are not included in the output.

## Examples

```
bayes_img <- ascii_data("bayes.png")
bayes_map <- ascii_map(file = bayes_img)
bayes_map
```

---

ascii\_plot

*Plots an ASCII character map*

---

## Description

Plots an ASCII character map

## Usage

```
ascii_plot(image_map, charsize = 4)
```

## Arguments

image_map	A tibble specifying a character map
charsize	Size of the characters

## Details

A simple plotting function for a character map. It takes a tibble as input, in the form output by the `ascii_map` function, and plots it using `ggplot2`. The `charsize` argument allows you to customise the size of the characters in the plot

## Value

A `ggplot` object.

## Examples

```
bayes_img <- ascii_data("bayes.png")
bayes_map <- ascii_map(file = bayes_img)
ascii_plot(bayes_map)
```

---

ascii_rain	<i>Writes an ASCII grid to an HTML file with the rain animation</i>
------------	---

---

**Description**

Writes an ASCII grid to an HTML file with the rain animation

**Usage**

```
ascii_rain(text_grid, file, fontsize = "5px", lineheight = "4px",  
          turnon = 0.1, turnoff = 0.025)
```

**Arguments**

text_grid	Matrix from ascii_grid
file	Path to HTML file
fontsize	How big is the text
lineheight	How tall is a line
turnon	Animation parameter
turnoff	Animation parameter

**Value**

A matrix, invisibly

**Examples**

```
## Not run:  
bayes_img <- ascii_data("bayes.png") # path to the bayes image  
bayes_map <- ascii_map(bayes_img) # construct ASCII map  
bayes_grid <- ascii_grid(bayes_map) # make grid  
ascii_rain(bayes_grid, file = "bayes_rain.html")  
## End(Not run)
```

---

ascii_text	<i>Writes an ASCII grid to text file</i>
------------	--

---

**Description**

Writes an ASCII grid to text file

**Usage**

```
ascii_text(text_grid, file)
```

**Arguments**

text_grid	Matrix from ascii_grid
file	Path to text file

**Value**

A matrix, invisibly

**Examples**

```
## Not run:  
bayes_img <- ascii_data("bayes.png") # path to the bayes image  
bayes_map <- ascii_map(bayes_img) # construct ASCII map  
bayes_grid <- ascii_grid(bayes_map) # make grid  
ascii_text(bayes_grid, file = "bayes_grid.txt")  
## End(Not run)
```

# Index

ascii\_data, 2  
ascii\_grid, 2  
ascii\_map, 3  
ascii\_plot, 4  
ascii\_rain, 5  
ascii\_text, 5