

Package: Rpixpack (via r-universe)

May 19, 2026

Title R Bindings for PixPack - Encode Files into PNG Images

Version 0.1.1

Description R bindings for PixPack, a utility that encodes arbitrary files and strings into PNG images with strong integrity checks using BLAKE3 hashing. Files can be encoded into visually distinctive PNG images and decoded back to recover the exact original data byte-for-byte. The underlying PixPack implementation uses a macro-cell grid format with error detection capabilities.

License MIT

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

SystemRequirements Cargo (Rust's package manager), rustc

Imports png

Suggests tinytest, rmarkdown, knitr

VignetteBuilder knitr

Config/pak/sysreqs libpng-dev libclang-dev

Repository <https://sounkou-bioinfo.r-universe.dev>

Date/Publication 2025-08-18 07:52:55 UTC

RemoteUrl <https://github.com/sounkou-bioinfo/pixpack>

RemoteRef HEAD

RemoteSha d066d472be8b2c3608c27deec83aa61cd1df5853

Contents

fileConversion	2
pixpack_convert	2
pixpack_info	3
pixpack_plot	4
pixpack_text	5
StringConversion	5

Index[7](#)

fileConversion	<i>Encode a file to PixPack PNG or decode a PixPack PNG back to original file.</i>
----------------	--

Description

Encode a file to PixPack PNG or decode a PixPack PNG back to original file.

Usage

```
fileConversion(path)
```

Arguments

path	Input file path (character scalar).
------	-------------------------------------

Value

Output artifact path (character scalar).

pixpack_convert	<i>Encode or decode files with automatic format detection</i>
-----------------	---

Description

This is a user-friendly wrapper around the core fileConversion function. It automatically detects whether to encode (file -> PNG) or decode (PNG -> file) based on the input file type.

Usage

```
pixpack_convert(file_path, verbose = TRUE)
```

Arguments

file_path	Path to the input file. If it's a PNG created by PixPack, it will be decoded. Otherwise, it will be encoded into a PNG.
verbose	Logical; if TRUE, prints status messages

Value

Path to the output file (either the created PNG or decoded file)

Examples

```
## Not run:
# Create a test file
test_file <- tempfile(fileext = ".txt")
writeLines(c("Hello", "World"), test_file)

# Encode to PNG
png_file <- pixpack_convert(test_file)

# Decode back to original
decoded_file <- pixpack_convert(png_file)

# Clean up
unlink(c(png_file, decoded_file, test_file))

## End(Not run)
```

pixpack_info

Get information about a PixPack file

Description

Extracts basic information about a file or PixPack PNG without decoding.

Usage

```
pixpack_info(file_path)
```

Arguments

file_path Path to the file to inspect

Value

A list with file information

Examples

```
## Not run:
# Create test file
test_file <- tempfile(fileext = ".txt")
writeLines(c("Line 1", "Line 2"), test_file)

# Get info
info <- pixpack_info(test_file)
print(info)

# Clean up
unlink(test_file)
```

```
## End(Not run)
```

pixpack_plot	<i>Visualize PixPack PNG metadata</i>
--------------	---------------------------------------

Description

Creates a simple visualization showing the grid structure and metadata of a PixPack PNG file.

Usage

```
pixpack_plot(png_path, show_grid = TRUE)
```

Arguments

png_path	Path to a PixPack PNG file
show_grid	Logical; if TRUE, overlays a grid showing the data cells

Value

Invisibly returns a list with metadata (if extractable)

Examples

```
## Not run:  
# Create a test PNG  
test_text <- "Hello, PixPack!"  
png_file <- tempfile(fileext = ".png")  
pixpack_text(test_text, png_file)  
  
# Visualize it  
pixpack_plot(png_file)  
  
# Clean up  
unlink(png_file)  
  
## End(Not run)
```

pixpack_text	<i>Encode text to PNG or decode PNG to text</i>
--------------	---

Description

User-friendly wrapper for encoding strings to PNG images or decoding them back.

Usage

```
pixpack_text(text = "", png_path, verbose = TRUE)
```

Arguments

text	Character string to encode (ignored if png_path exists and is a PixPack PNG)
png_path	Path where the PNG should be saved or read from
verbose	Logical; if TRUE, prints status messages

Value

If encoding: path to created PNG. If decoding: the decoded text.

Examples

```
## Not run:  
# Encode text to PNG  
png_file <- tempfile(fileext = ".png")  
result <- pixpack_text(text = "Hello, PixPack!", png_path = png_file)  
  
# Decode PNG back to text  
decoded_text <- pixpack_text(png_path = png_file)  
print(decoded_text) # Should print "Hello, PixPack!"  
  
# Clean up  
unlink(png_file)  
  
## End(Not run)
```

StringConversion	<i>Encode a string into a PixPack PNG at given path or decode an existing PixPack PNG to string. If png_path already exists and is a PixPack PNG, it is decoded; otherwise input is encoded to that PNG path.</i>
------------------	---

Description

Encode a string into a PixPack PNG at given path or decode an existing PixPack PNG to string. If `png_path` already exists and is a PixPack PNG, it is decoded; otherwise `input` is encoded to that PNG path.

Usage

```
StringConversion(input, png_path)
```

Arguments

<code>input</code>	Character scalar (content when encoding; ignored on decode).
<code>png_path</code>	Target PNG path (must be provided).

Value

On encode: PNG path; On decode: decoded UTF-8 string (lossy).

Index

fileConversion, [2](#)

pixpack_convert, [2](#)
pixpack_info, [3](#)
pixpack_plot, [4](#)
pixpack_text, [5](#)

StringConversion, [5](#)