

Package ‘tinyimg’

March 6, 2026

Title Optimize and Compress Images

Version 0.3

Description Optimize and compress images using 'Rust' libraries to reduce file sizes while maintaining image quality. Supports PNG palette reduction and dithering via the 'exoquant' crate before lossless PNG optimization via the 'oxipng' crate. The package provides functions to optimize individual image files or entire directories, with configurable compression levels.

License MIT + file LICENSE

URL <https://github.com/yihui/tinyimg>

BugReports <https://github.com/yihui/tinyimg/issues>

SystemRequirements Cargo (Rust's package manager), rustc

Encoding UTF-8

RoxygenNote 7.3.3

Suggests testit

Config/rextendr/version 0.3.1

NeedsCompilation yes

Author Yihui Xie [aut, cre, cph] (ORCID:
<<https://orcid.org/0000-0003-0645-5666>>, URL: <https://yihui.org>),
Authors of the dependency Rust crates [ctb, cph] (see AUTHORS file)

Maintainer Yihui Xie <xie@yihui.name>

Repository CRAN

Date/Publication 2026-03-06 07:50:02 UTC

Contents

tinypng	2
Index	4

tinypng

*Optimize PNG images***Description**

Optimize PNG files or directories of PNG files using optional lossy palette reduction and dithering before lossless compression.

Usage

```
tinypng(
  input,
  output = identity,
  level = 2L,
  alpha = FALSE,
  preserve = TRUE,
  recursive = TRUE,
  verbose = TRUE,
  lossy = 0
)
```

Arguments

input	Path to the input PNG file or directory. If a directory is provided, all PNG files in the directory (and subdirectories if <code>recursive = TRUE</code>) will be optimized.
output	Path to the output PNG file or directory, or a function that takes an input file path and returns an output path. When optimizing a directory, output should be a directory path or a function.
level	Optimization level (0-6). Higher values result in better compression but take longer.
alpha	Optimize transparent pixels for better compression. This is technically lossy but visually lossless.
preserve	Preserve file permissions and timestamps. Ignored when lossy optimization is enabled (<code>lossy > 0</code>).
recursive	When input is a directory, recursively process subdirectories.
verbose	Print file size reduction info for each file.
lossy	A numeric threshold for the color difference in lossy processing. Values ≤ 0 disable lossy optimization.

Details

The lossy algorithm uses color difference in the International Commission on Illumination (CIE) 1976 $L^*a^*b^*$ (often written as CIELAB or Lab) color space.

For a candidate palette size n , the image is quantized with n colors using nearest-color mapping. Sampled pixels are then **grouped by their original color**, so each distinct color in the image gets

one equal vote regardless of how many pixels share it (preventing a large uniform background from masking errors in rarer content colors). The worst-case ΔE_{76} within each color group is recorded, and we take the 95th percentile of those per-color values. Bisection on n (1–256) finds the smallest palette size whose per-color p95 is \leq lossy.

$\Delta E_{76} \approx 2.3$ is often considered the just noticeable difference (JND) threshold. Larger values allow more color difference and thus smaller palette and file sizes, with more loss of color fidelity. In theory, ΔE_{76} can exceed 100.

Value

Character vector of output file paths (invisibly).

References

https://en.wikipedia.org/wiki/Color_difference

Examples

```
# Create a test PNG
tmp = tempfile()
png(tmp, width = 400, height = 400)
plot(1:10)
dev.off()

# Optimize with different levels
tinypng(tmp, paste0(tmp, "-o1.png"), level = 1)
tinypng(tmp, paste0(tmp, "-o6.png"), level = 6)
tinypng(tmp, paste0(tmp, "-lossy.png"), lossy = 2.3)
```

Index

tinypng, [2](#)