Package: avif (via r-universe)

October 17, 2026

Title Read and Write AVIF Images

Version 0.0.1

Description Functions to read and write AV1 Image File Format. It can read and write both files and in-memory raw vectors.

License BSD_3_clause + file LICENSE

URL https://github.com/r-zon/avif

BugReports https://github.com/r-zon/avif/issues

Depends R (>= 4.5.0)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

SystemRequirements libavif, zig (>= 0.16.0)

Config/pak/sysreqs libavif-dev

Repository https://r-zon.r-universe.dev

Date/Publication 2026-05-20 05:44:49 UTC

RemoteUrl https://github.com/r-zon/avif

RemoteRef HEAD

RemoteSha a48b6a1623908d53be730a08fa078c6a40bf4b95

Contents

read_avif 2
write_avif 3

Index **4**

read_avif *Read an AVIF Image*

Description

Read an AVIF image from a file or a raw vector into an RGB array.

Usage

```
read_avif(
  source,
  ...,
  ptype = raw(),
  normalize = FALSE,
  native_raster = FALSE,
  codec = NULL,
  jobs = NULL
)
```

Arguments

source	A file path or a raw vector.
...	Unused.
ptype	Prototype like raw(), 0L or 0.0 that defines the output type.
normalize	If TRUE, output a normalized (0-1) real array.
native_raster	If TRUE, output a nativeRaster integer matrix.
codec	Codec for decoding, automatic if NULL.
jobs	Number of decoder threads, must be greater than 0, or NULL for all cores.

Value

An RGB array.

Examples

```
if (file.exists("8bpc.avif")) {
  read_avif("8bpc.avif")
}
if (file.exists("10bpc.avif")) {
  read_avif("10bpc.avif", ptype = 0L, native_raster = TRUE)
}
if (file.exists("12bpc.avif")) {
  read_avif(
    readBin("12bpc.avif", "raw", file.size("12bpc.avif")),
    ptype = 0.,
    normalize = TRUE
  )
}
```

write_avif *Write an AVIF Image*

Description

Create an AVIF image from an RGB array into a file or return a raw vector.

Usage

```
write_avif(
  image,
  target = NULL,
  ...,
  speed = 6L,
  quality = 60L,
  alpha_quality = 60L,
  format = 444L,
  codec = NULL,
  jobs = NULL
)
```

Arguments

image	A raw or integer vector with dimensions (height, width, channel).
target	A file path, or NULL for a raw vector.
...	Unused.
speed	Encoder speed in [0, 10] where 0 is the slowest, 10 is the fastest.
quality	Color quality in [0, 100] where 100 is lossless.
alpha_quality	Alpha quality in [0, 100] where 100 is lossless.
format	YUV format, must be one of 444, 422, 420 or 400.
codec	Codec for encoding, automatic if NULL.
jobs	Number of encoder threads, must be greater than 0, or NULL for all cores.

Value

NULL if target is a file path, otherwise a raw vector.

Examples

```
rgb_array <- hcl.colors(700) |>
  col2rgb() |>
  as.raw() |>
  array(c(3, 700, 100)) |>
  aperm()
save_path <- file.path(tempdir(), "8bpc.avif")
write_avif(rgb_array, save_path, speed = 0L, quality = 100L)
writeBin(write_avif(rgb_array), save_path)
```

Index

read_avif, 2

write_avif, 3