

# Package: crop (via r-universe)

August 27, 2024

**Version** 0.0-3

**Encoding** UTF-8

**Title** Graphics Cropping Tool

**Description** A device closing function which is able to crop graphics (e.g., PDF, PNG files) on Unix-like operating systems with the required underlying command-line tools installed.

**Author** Marius Hofert [aut, cre]

**Maintainer** Marius Hofert <mhofert@hku.hk>

**Depends** R (>= 3.0.0)

**License** GPL-2 | GPL-3

**NeedsCompilation** no

**Date/Publication** 2024-03-04 16:20:14 UTC

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

**RemoteUrl** <https://github.com/cran/crop>

**RemoteRef** HEAD

**RemoteSha** e88f27533f3c56d7f7be5fe61618e7176858fd63

## Contents

dev_off_crop . . . . .	1
<b>Index</b>	<b>4</b>

---

dev_off_crop	<i>Cropping Device</i>
--------------	------------------------

---

## Description

`dev.off.crop()` calls `dev.off()` to close the plotting device and then crops the produced graphic. This works on Unix-like systems if the right underlying Unix tools are installed and found (see below).

## Usage

```
dev.off.crop(file = NULL, warn = TRUE, ...)
```

## Arguments

file	An output file name including extension.
warn	A <a href="#">logical</a> indicating whether warnings shall be given in case the wrong operating system is used or the underlying Unix tools are not available.
...	Additional arguments passed to <a href="#">dev.off()</a> .

## Details

Supported file extensions are .ps (epstool), .eps (epstool), .pdf (pdfcrop), .png (mogrify) if the underlying tools (given in parentheses) are installed and found. If they are not installed or found or if the wrong operating system is used, simply `dev.off()` is called (and thus no cropping is done).

## Value

[invisible\(\)](#).

## Author(s)

Marius Hofert

## See Also

[dev.off\(\)](#) for closing a device.

## Examples

```
## Example data
x <- 1:10
y <- rev(x)

## PS
postscript(file = (file <- "crop.ps"))
plot(x, y, xlab = "x", ylab = "y")
dev.off.crop(file = file)
file.remove("crop.ps") # clean-up

## EPS
setEPS() # or use onefile = FALSE, horizontal = FALSE, paper = "special" in postscript()
postscript(file = (file <- "crop.eps"))
plot(x, y, xlab = "x", ylab = "y")
dev.off.crop(file = file)
file.remove("crop.eps") # clean-up

## PDF
pdf(file = (file <- "crop.pdf"))
```

```
plot(x, y, xlab = "x", ylab = "y")
dev.off.crop(file = file)
file.remove("crop.pdf") # clean-up

## PNG with white background
png(filename = (file <- "crop.png"))
plot(x, y, xlab = "x", ylab = "y")
dev.off.crop(file = file)
file.remove("crop.png") # clean-up

## PNG with transparent background
## Not doable by the underlying 'mogrify' (only cropped vertically, not horizontally)
if(FALSE) {
  png(filename = (file <- "crop.png"), bg = "transparent")
  plot(x, y, xlab = "x", ylab = "y")
  dev.off.crop(file = file)
}
```

# Index

## \* utilities

dev\_off\_crop, 1

dev.off, 1, 2

dev.off.crop (dev\_off\_crop), 1

dev\_off\_crop, 1

invisible, 2

logical, 2