

# Package ‘tesseract’

August 10, 2018

**Type** Package

**Title** Open Source OCR Engine

**Version** 2.3

**Description** Bindings to 'Tesseract' <<https://opensource.google.com/projects/tesseract>>:  
a powerful optical character recognition (OCR) engine that supports over 100 languages.  
The engine is highly configurable in order to tune the detection algorithms and  
obtain the best possible results.

**License** Apache License 2.0

**URL** <https://github.com/ropensci/tesseract>

**BugReports** <https://github.com/ropensci/tesseract/issues>

**SystemRequirements** Tesseract >= 3.03 (libtesseract-dev /  
tesseract-devel) and Leptonica (libleptonica-dev /  
leptonica-devel). On Debian you need to install the English  
training data separately (tesseract-ocr-eng)

**Imports** Rcpp (>= 0.12.12), pdftools (>= 1.5), curl, rappdirs, tibble,  
digest

**LinkingTo** Rcpp

**RoxygenNote** 6.1.0

**Suggests** magick (>= 1.7), spelling, knitr, rmarkdown

**Encoding** UTF-8

**VignetteBuilder** knitr

**Language** en-US

**NeedsCompilation** yes

**Author** Jeroen Ooms [aut, cre] (<<https://orcid.org/0000-0002-4035-0289>>)

**Maintainer** Jeroen Ooms <[jeroen@berkeley.edu](mailto:jeroen@berkeley.edu)>

**Repository** CRAN

**Date/Publication** 2018-08-10 21:50:03 UTC

## R topics documented:

ocr . . . . .	2
tesseract . . . . .	3
tesseract_download . . . . .	4
<b>Index</b>	<b>6</b>

---

ocr	<i>Tesseract OCR</i>
-----	----------------------

---

### Description

Extract text from an image. Requires that you have training data for the language you are reading. Works best for images with high contrast, little noise and horizontal text. See [tesseract wiki](#) and our package vignette for image preprocessing tips.

### Usage

```
ocr(image, engine = tesseract("eng"), HOOCR = FALSE)
```

```
ocr_data(image, engine = tesseract("eng"))
```

### Arguments

image	file path, url, or raw vector to image (png, tiff, jpeg, etc)
engine	a tesseract engine created with <a href="#">tesseract()</a> . Alternatively a language string which will be passed to <a href="#">tesseract()</a> .
HOOCR	if TRUE return results as HOOCR xml instead of plain text

### Details

The `ocr()` function returns plain text by default, or hOCR text if hOCR is set to TRUE. The `ocr_data()` function returns a data frame with a confidence rate and bounding box for each word in the text.

### References

[Tesseract: Improving Quality](#)

### See Also

Other tesseract: [tesseract\\_download](#), [tesseract](#)

## Examples

```
# Simple example
text <- ocr("https://jeroen.github.io/images/testocr.png")
cat(text)

xml <- ocr("https://jeroen.github.io/images/testocr.png", HOOCR = TRUE)
cat(xml)

df <- ocr_data("https://jeroen.github.io/images/testocr.png")
print(df)

# Full roundtrip test: render PDF to image and OCR it back to text
curl::curl_download("https://cran.r-project.org/doc/manuals/r-release/R-intro.pdf", "R-intro.pdf")
orig <- pdftools::pdf_text("R-intro.pdf")[1]

# Render pdf to png image
img_file <- pdftools::pdf_convert("R-intro.pdf", format = 'tiff', pages = 1, dpi = 400)

# Extract text from png image
text <- ocr(img_file)
unlink(img_file)
cat(text)

engine <- tesseract(options = list(tessedit_char_whitelist = "0123456789"))
```

---

tesseract

*Tesseract Engine*

---

## Description

Create an OCR engine for a given language and control parameters. This can be used by the `ocr` and `ocr_data` functions to recognize text.

## Usage

```
tesseract(language = NULL, datapath = NULL, configs = NULL,
  options = NULL, cache = TRUE)

tesseract_params(filter = "")

tesseract_info()
```

## Arguments

`language` string with language for training data. Usually defaults to eng  
`datapath` path with the training data for this language. Default uses the system library.

configs	character vector with files, each containing one or more parameter values. These config files can exist in the current directory or one of the standard tesseract config files that live in the tessdata directory. See details.
options	a named list with tesseract parameters. See details.
cache	speed things up by caching engines
filter	only list parameters containing a particular string

### Details

Tesseract **control parameters** can be set either via a named list in the options parameter, or in a config file text file which contains the parameter name followed by a space and then the value, one per line. Use `tesseract_params()` to list or find parameters. Note that that some parameters are only supported in certain versions of libtesseract, and that invalid parameters can sometimes cause libtesseract to crash.

### References

[tesseract wiki: control parameters](#)

### See Also

Other tesseract: [ocr](#), [tesseract\\_download](#)

### Examples

```
tesseract_params('debug')
```

---

tesseract\_download     *Tesseract Training Data*

---

### Description

Helper function to download training data from the official **tessdata** repository. Only use this function on Windows and OS-X. On Linux, training data can be installed directly with **yum** or **apt-get**.

### Usage

```
tesseract_download(lang, datapath = NULL, progress = interactive())
```

### Arguments

lang	three letter code for language, see <b>tessdata</b> repository.
datapath	destination directory where to download store the file
progress	print progress while downloading

## Details

Tesseract uses training data to perform OCR. Most systems default to English training data. To improve OCR performance for other languages you can to install the training data from your distribution. For example to install the spanish training data:

- [tesseract-ocr-spa](#) (Debian, Ubuntu)
- [tesseract-langpack-spa](#) (Fedora, EPEL)

On Windows and MacOS you can install languages using the [tesseract\\_download](#) function which downloads training data directly from [github](#) and stores it in a the path on disk given by the TESSDATA\_PREFIX variable.

## References

[tesseract wiki: training data](#)

## See Also

Other tesseract: [ocr](#), [tesseract](#)

## Examples

```
if(is.na(match("fra", tesseract_info()$available)))
  tesseract_download("fra")
french <- tesseract("fra")
text <- ocr("https://jeroen.github.io/images/french_text.png", engine = french)
cat(text)
```

# Index

ocr, [2](#), [3-5](#)

ocr\_data, [3](#)

ocr\_data(ocr), [2](#)

tessdata(tesseract\_download), [4](#)

tesseract, [2](#), [3](#), [5](#)

tesseract(), [2](#)

tesseract\_download, [2](#), [4](#), [4](#), [5](#)

tesseract\_info(tesseract), [3](#)

tesseract\_params(tesseract), [3](#)

tesseract\_params(), [4](#)