Package: kwb.gocr (via r-universe)

October 27, 2024

Title Interface to gocr Program

Version 0.1.0

Description Wrapper functions to the gocr (Optical Character Recognition) program developed by Jens Schulenberg (https://www-e.ovgu.de/jschulen/ocr/).

License MIT + file LICENSE

URL https://github.com/KWB-R/kwb.gocr

BugReports https://github.com/KWB-R/kwb.gocr/issues

Imports bitops, kwb.utils

Remotes github::kwb-r/kwb.utils

Encoding UTF-8

RoxygenNote 7.1.2

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

RemoteUrl https://github.com/KWB-R/kwb.gocr

RemoteRef HEAD

RemoteSha 2888427de059c08fcef92df0cfbd79a69a199362

Contents

gocrConfig *Create a gocr Configuration*

Description

Create a gocr Configuration

Usage

```
gocrConfig(
  inputfile,
  showhelp = FALSE,
  outputfile = "",
  errorfile = "",
  progressfile = "",
  databasepath = "",
  outputformat = "",
  greylevel = 0,
  dustsize = -1,
  spacewidth = 0,
  verbosity = 0,
  limitVerbosityToChars = "",
  limitRecognitionToChars = "",
  certainty = 95,
  mode = 0,
  onlyRecogniseNumbers = FALSE
)
```

Arguments

`inputfile`	-i file: read input from file (or stdin if file is a single dash)
`showhelp`	-h: show usage information
`outputfile`	-o file: send output to file instead of stdout. If "" (default), a file "gocrOut_<*basename(outputfile)*>" in `tempdir()` is used as the output file.
`errorfile`	-e file: send errors to file instead of stderr or to stdout if file is a dash
`progressfile`	-x file: progress output to file (file can be a file name, a fifo name or a file descriptor 1...255), this is useful for GUI developpers to show the OCR progress, the file descriptor argument is only available, if compiled with __USE_POSIX defined
`databasepath`	-p path: database path, that will be populated with images of learned characters. If "" (default), and a database is needed, a directory within the folder of the installed package is used
`outputformat`	-f format: output format of the recognized text (ISO8859_1 TeX HTML XML UTF8 ASCII), XML will also output position and probability data

`greylevel` -l level set grey level to level (0<160<=255, default: 0 for autodetect), darker pixels belong to characters, brighter pixels are inter- preted as background of the input image

`dustsize` -d size: set dust size in pixels (clusters smaller than this are removed), 0 means no clusters are removed, the default is -1 for auto detection

`spacewidth` -s num: set spacewidth between words in units of dots (default: 0 for autodetect), wider widths are interpreted as word spaces, smaller as character spaces

`verbosity` -v verbosity: be verbose to stderr; verbosity is a bitfield. Use `optionValueVerbosity` to get a proper value

`limitVerbosityToChars`
-c string: only verbose output of characters from string to stderr, more output is generated for all characters within the string, the

`limitRecognitionToChars`
-C string: only recognise characters from string, this is a filter function in cases where the interest is only to a part of the character alphabet

`certainty` -a certainty: set value for certainty of recognition (0..100; default: 95), characters with a higher certainty are accepted, characters with a lower certainty are treated as unknown (not recognized); set higher values, if you want to have only more certain recognized characters

`mode` -m mode: set oprational mode; mode is a bitfield (default: 0). Use `optionValueMode` to get a proper value

`onlyRecogniseNumbers`
-n bool: if bool is non-zero, only recognise numbers (this is now obsolete, use -C "0123456789")

`gocrDownload` *Download gocr executable*

Description

Download gocr executable

Usage

```
gocrDownload(
  version_number = "048",
  overwrite = FALSE,
  target_dir = file.path(system.file(package = "kwb.gocr"), "extdata/gocr")
)
```

Arguments

`version_number` latest version number is "049". However, "048" was used for the development of this R package and is still available (default: "048")

`overwrite` if TRUE downloads and overwrites existing gocr executable in target_directory, otherwise not (default: FALSE)

`target_dir` target directory (default: file.path(system.file(package = "kwb.gocr"), "extdata/gocr2")

Value

downloads gocr executable to target directory and returns path

Examples

```
gocrDownload(version_number = "048")
## Not run:
gocrDownload(version_number = "049")

## End(Not run)
```

`gocrExePath` *Path to gocr Executable File*

Description

Path to gocr Executable File

Usage

```
gocrExePath()
```

`gocrOptionString` *Option String for gocr Call*

Description

Option String for gocr Call

Usage

```
gocrOptionString(config)
```

Arguments

`config`	gocr configuration as returned by `gocrConfig`

gocrRun *Run gocr on an Image File*

Description

Run gocr on an Image File

Usage

```
gocrRun(
  config,
  useBatch = TRUE,
  waitForBatch = TRUE,
  opendir = TRUE,
  dbg = TRUE
)
```

Arguments

`config`	gocr configuration as returned by `gocrConfig`
`useBatch`	if TRUE (default), a batch file is written so that the user can reproduce the call by double-clicking the batch file in the file explorer (opens when *opendir* is TRUE)
`waitForBatch`	passed to `kwb.gocr:::writeAndRunBatchFile`
`opendir`	if TRUE (default), and if *useBatch* is TRUE, the directory in which the batch file is written, is opened in the Windows Explorer
`dbg`	if `TRUE` debug messages are shown

Value

(only if *waitForBatch* = TRUE): result of OCR as a vector of character representing the recognised lines. The result vector has the attribute *config* containing the configuration used (original config, with default values set where needed)

optionValueMode *Value for Mode Option*

Description

Value for Mode Option

Usage

```
optionValueMode(
  useDatabase = FALSE,
  layoutAnalysis = FALSE,
  doNotCompare = FALSE,
  doNotDivide = FALSE,
  doNotCorrect = FALSE,
  characterPacking = FALSE,
  extendDatabase = FALSE,
  switchOffEngine = FALSE
)
```

Arguments

`useDatabase`	(2) use database to recognize characters which are not recognized by other algorithms, (early development)
`layoutAnalysis`	(4) switching on layout analysis or zoning (development)
`doNotCompare`	(8) don't compare unrecognized characters to recognized one
`doNotDivide`	(16) don't try to divide overlapping characters to two or three single characters
`doNotCorrect`	(32) don't do context correction
`characterPacking`	(64) character packing, before recognition starts, similar characters are searched and only one of this characters will be send to the recognition engine (development)
`extendDatabase`	(128) extend database, prompts user for unidentified characters and extends the database with users answer (128+2, early development)
`switchOffEngine`	(256) switch off the recognition engine (makes sense together with -m 2)

References

http://manpages.ubuntu.com/manpages/gutsy/man1/gocr.1.html

`optionValueVerbosity` *Value for Option Verbosity*

Description

Value for Option Verbosity

Usage

```
optionValueVerbosity(
  printMore = 1,
  listShapes = 1,
  listPattern = 1,
  printPattern = 1,
  printDebug = 1,
  createOutPng = 0
)
```

Arguments

`printMore` (1) print more info

`listShapes` (2) list shapes of boxes (see -c) to stderr

`listPattern` (4) list pattern of boxes (see -c) to stderr

`printPattern` (8) print pattern after recognition for debugging

`printDebug` (16) print debug information about recognition of lines to stderr

`createOutPng` (32) create outXX.png with boxes and lines marked on each general OCR-step

Index